To Sense or To Transmit: A Learning-based Spectrum Management Scheme for Cognitive Radio Mesh Networks

Marco Di Felice*, Kaushik Roy Chowdhury†, Waleed Meleis†, Luciano Bononi*
* Department of Computer Science, University of Bologna, Italy
Email: {difelice,bononi}@cs.unibo.it
† Department of Electrical and Computer Engineering, Northeastern University, Boston, USA
Email:{krc,meleis}@ece.neu.edu

***Abstract*—Wireless mesh networks, composed of interconnected clusters of mesh router (MR) and multiple associated mesh clients (MCs), may use cognitive radio equipped transceivers, allowing them to choose licensed frequencies for high bandwidth communication. However, the protection of the licensed users in these bands is a key constraint. In this paper, we propose a reinforcement learning based approach that allows each mesh cluster to independently decide the operative channel, the durations for spectrum sensing, the time of switching, and the duration for which the data transmission happens. The contributions made in this paper are threefold. First, based on accumulated rewards for a channel mapped to the link transmission delays, and the estimated licensed user activity, the MRs assign a weight to each of the channels, thereby selecting the channel with highest performance for MCs operations. Second, our algorithm allows dynamic selection of the sensing time interval that optimizes the link throughput. Third, by cooperative sharing, we allow the MRs to share their channel table information, thus allowing a more accurate learning model. Simulations results reveal significant improvement over classical schemes which have pre-set sensing and transmission durations in the absence of learning.**

I. Introduction

Wireless mesh networks (WMNs) have been widely deployed on a large-scale basis, extending the reach of the public Internet and providing high bandwidth connectivity to rural areas, corporate enterprises, as well as across densely populated cities. A typical WMN is composed of a number of interconnected mesh routers (MRs) that form the communication backbone, with one or more mesh clients (MCs) associated with each MR. Thus, the set of MCs and their controlling MR may be jointly considered as a *cluster*, with the individual MRs forwarding packets over multiple hops to the Internet gateway. Most commercial WMNs [9] operate in the unlicensed ISM frequency band that is used by other local WiFi networks, cordless phones, remote controls for appliances, computer peripherals, and also has significant energy injected by heavy machinery and conventional microwave ovens. Thus, the WMN is likely to perceive elevated levels of interference in the densely populated city or residential areas. One solution for this concern is to pursue a spectrum-aware design for WMNs, so that each cluster may intelligently evaluate the need for switching to less occupied frequencies, choose the best frequency among the available options, while maximizing the network performance at the same time.

Recently, cognitive radio (CR) has been proposed as an enabling technology that allows for opportunistic use of the licensed spectrum bands, under the constraint that the operation of the licensed users of the bands are not affected [1]. A recent ruling by the FCC has allowed the use of unlicensed devices from range $511\,\mathrm{MHz}$ to $608\,\mathrm{MHz}$, a total of $180\,\mathrm{MHz}$ [6]. On the frequency scale, these bands are considerably lower than the $2.4\,\mathrm{GHz}$ frequency ISM band, and hence, they offer better propagation characteristics. Specifically for the case of economical deployment of WMNs, this translates to greater coverage, which allows fewer MRs to be deployed, and separated over wider distances. However, in such a dynamic spectrum access environment, the operation of the licensed users in the non-ISM frequency bands must be protected [4] [8]. Thus, the key challenge lies in answering the following questions, on a per-cluster basis: (*i*) when and for low long to sense the spectrum, (*ii*) when to switch the spectrum, and (*iii*) when to transmit [3]. We propose a reinforcement learning (RL) based channel selection and access protocol to address the above concerns, which allows each MR to autonomously learn of the best spectrum sensing, switching, and data transmission policy. Moreover, we also propose a cooperation framework that allows the MRs to contribute spectrum usage information locally, thereby allowing the MR to make spectrum-aware decisions with greater accuracy.

Reinforcement learning [2] is a learning paradigm, which was inspired by psychological learning theory from biology. Each agent (here, the MR) in the system (here, the WMN cluster) can sense the environment and achieve its own local knowledge and experience. Within an environment, a learning agent attempts to perform optimal actions to maximize long-term rewards that is the expected accumulated reward gained by performing certain actions, such as transmission, sensing or spectrum switching [7]. As an example, in the default case, the sensing duration t_s is fixed and precedes the transmission duration t_{tx}. We express the rewards in terms of time units that are directly decided by the network operation. Our spectrum access scheme may determine, based on the past history of

978-1-4244-7977-1/10/$26.00 ©2010 IEEE

its transmissions, that the long-term reward is maximized by increasing the sensing time to $k \times t_s$, $k > 1$, at the cost of reducing the current throughput on the link. Our scheme also explores the different spectrum bands conservatively, balancing the costs and benefits of *exploiting* the current best channel, as opposed to *exploring* a new channel.

The rest of this paper is organized as follows. Section II describes the related work in the use of learning for wireless networks, and in particular cognitive WMNs. Section III presents the network architecture. Section IV describes our proposed learning-based spectrum selection and access scheme. We undertake a thorough performance evaluation in Section V, and finally, Section VI concludes our work.

II. Related Works

In this paper we address the problem of spectrum management over cognitive WMNs, jointly considering spectrum sensing and spectrum selection tasks. Due to the importance of avoiding interference with licensed users, spectrum sensing constitutes a key and well investigated topic in CR systems [1], and several different techniques have been proposed in the literature, e.g. energy detection [4] and cooperative sensing schemes [8]. In many cases, these techniques necessitate a periodic sensing structure, where sensing and transmission operations are performed in a periodic manner with separate observation period (e.g. t_s) and transmission period (e.g. t_{tx}). In [7], the authors propose an optimization framework which derives the optimal value of t_s and t_{tx}, subject to interference avoidance constraints. Also, the impact of sensing-induced delay on the upper layer of the protocol stack has been investigated in [5]. Analogously, several distributed and centralized spectrum selection schemes have been proposed for both classical multi-channel and cognitive WMNs [10] [12]. However, few works provide a joint solution for spectrum sensing and spectrum selection tasks, by considering the mutual dependance between them. In some cases, the difficulties rely on the number of parameters, e.g. spectrum, power level, sensing frequency, which should be controlled by the optimization framework. Reinforcement learning (RL) constitutes a promising approach for distributed parameter optimization in cognitive radio networks, in e.g. routing [11], spectrum sensing [3] and spectrum decision [13] tasks. Instead of addressing a single factor at a time, a RL agent can observe all the factors as a state, receive an aggregate feedback (e.g. the cost of each transmission) and optimize a general goal as a whole, e.g. throughput [11]. In [14], the authors propose a cognitive MAC protocol based on Partially Observable Markov Decision Process (POMDP). Similar to our paper, in [3] the authors use a RL approach to solve the problem of spectral resources in OFDM-based CR networks. However, the scheme proposed in [3] does not balance the trade-off between sensing and transmitting actions. Compared to other existing proposal, in this paper we jointly address: (*i*) PU-aware and channel-aware spectrum selection, (*ii*) optimization of sensing/transmitting time and (*iii*) cooperation among mesh nodes.

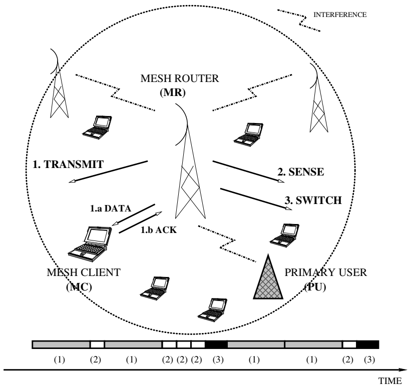


Fig. 1. Network architecture and functionalities.

III. Network Model and Architecture

We assume that all the WMN nodes are equipped with two radio interfaces: a control radio interface (R_{cont}), which operates in the unlicensed band and is used to transmit control messages, and a spectrum-agile radio interface (R_{CR}) for MR-MC data traffic. All the nodes of a mesh cluster operate in the same band of the licensed spectrum, but different clusters may operate on different bands to avoid inter-cluster interference problems. We assume M licensed or primary bands are present, where different bands have their own distinct center frequency, and possibly varying bandwidth resulting in different data-rates. Moreover, each licensed band can be occupied by a primary user (PU), whose activity is defined through an alternative exponential ON-OFF scheme, called as a birth-death Markov process: Let α_i be the death rate (or departure rate) for a PU on channel i (e.g. PU_i), then the duration of ON state follows an exponential distribution with mean $\frac{1}{\alpha_i}$. Similarly, let β_i be the birth rate (or arrival rate) for PU_i, then the duration of OFF state follows an exponential distribution with mean $1/\beta_i$.

The MR issues the `sense`, `switch`, and `transmit` commands to the MCs over the R_{cont} interface. Previous works such as [7] allow determining the sensing time t_s and transmission time t_{tx} to minimize the probability of false alarm (PU identified when channel is free), and missed detection (PU not detected, causing collisions). However, in our approach the MR may extend the sensing time $T_s \geq t_s$ through the `sense` command. If no PU is detected, then $T_s = t_s$. If PU is detected, then the channel is continued to be sensed for a duration $T_s = k \times t_s, k > 1, k \in I$ till it becomes available. The `switch` command issued on the R_{cont} contains the channel information which the cluster will use at the end of the switching delay, and the MR and MCs collectively switch to the new channel in a synchronous fashion. Note that no data transfer occurs during the sensing and switching durations. The `transmit` command allows both the MCs and MRs to engage in bi-directional communication over R_{CR} for the duration t_{tx}.

The network architecture of the WMN with the licensed users is shown in Figure 1. The MR is associated with several

MCs, forming a cluster. Along the time scale shown at the bottom, the cluster may undertake the `transmit` operation (shaded slot 1) or the `sense` (unfilled slot 2). However, the MR extends the sensing time to $3 \times t_s$ for slots $4 - -6$, which is determined by our algorithm based on the learning process. The current channel is exploited till slot 6, after which the `switch` command is issued by the MR, and a different channel is explored.

IV. Reinforcement Learning-based Channel Selection and Access

In our RL model, an agent interacts with its environment over a potentially infinite sequence of discrete time steps, say $t = 1, 2, 3, \ldots$. At each step, the agent finds itself in a *state*, chooses a permissible *action*, and observes the ensuing *reward*. As a result of the cumulative reward earned up to this time, the agent can choose to perform another action, or change its current state. We map this general concept of state-action pairs to the CR enabled WMN by formulating the learning framework as a Markov Decision Process composed of:

- A discrete set of *states* S, where each state represents one licensed frequency band.
- A discrete *set of actions* A, which may be either a *transmit* (a_{tx}), *sense* (a_{se}), or *switching* (a_{sw}) action of the node.
- A *reward function* R, which has a finite value for each state-action pair depending upon the current observed network behavior. Thus, there are a total of $S \times A \longrightarrow \Re$ entries for the reward function.

The goal of the RL agent is to find the optimal *policy* which maximizes the long-term expected reward. Formally, the policy can be given as $\pi^i(s, a)$ that states the decision making rule for selecting a particular action $a \in A$ (i.e., `sense`, `switch` or `transmit`) when the agent i is in state $s \in S$ (i.e., a licensed user channel) at a given time step t. Each action has a certain probability of being chosen, which is updated each time by the $Q^i_t(s, a)$ value calculated from the rewards earned as a result of performing the action. The future reward estimation is given by the state-value function $V^i_t(s)$. In summary, $V^i_t(s)$ states how good is a channel, and $Q^i_t(s, a)$ specifies the benefit of choosing one of above three network actions. Moreover, our learning model uses *eligibility trace* that gives an estimate of how reliable the state values $V^i_t(s)$ are [2]. We describe next the state actions, the rewards, and the state values in details:

A. State Actions

We recall that the MR in each cluster is the learning agent, which decides the spectrum and the sensing intervals of the cluster. Let the number of licensed bands be M. Thus, the number of states $|S| = M$. At a given time step t, a given MR, say MR i performs one action from the set A:

1) a_{tx}: MR i issues a broadcast message through its control radio R_{cont}, informing that the cluster members (MCs) may freely transmit on the channel. The permit for opportunistic transmission is limited to time t_{tx}. This allows the cluster to perform the `transmit` action.
2) a_{se}: The MR i issues a directive to the cluster for spectrum `sense` action, also through its control interface R_{cont}. This duration for the resulting sensing process is given by time T_s, whose value is at least t_s, or can be an integral multiple of t_s. During this time, there is no transmission by the cluster members, and thus the data throughput is 0.
3) a_{sw}: MR i broadcasts the command to switch the spectrum, thus entering into the `switch` state. The cluster then performs a handoff to the newly chosen spectrum band, with an handoff delay equal to t_h.

As for our formulation, the licensed spectrum bands are represented by states. Hence, only the switching action a_{sw} results in a state transition. Moreover, the rewards from the different possible actions are a function of the time delay to complete that action, i.e., the more the time delay, lesser is the reward. We describe next how the rewards are calculated for the each of the three actions.

1) Rewards: Each time MR i performs a a_{tx} action, it receives the corresponding reward $r(a_{tx})$ defined as follows:

$$r(a_{tx}) = \begin{cases} D_{max} - d & \text{if tx is successful} \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

Here, D_{max} defines the maximum delay that can be tolerated for a data transmission on the wireless link, a limit set by the Quality of Service (QoS) required by the MC's applications, or offered by the WMN to end-users. d is the sample link-layer delay, defined as the time taken to successfully deliver a packet from the MR to a MC. We compute d as $T_{init} - T_{ack}$, where T_{init} is the time when the packet is received on MAC layer from the MR, to the instant T_{ack}, which is the time when the corresponding ACK is received. Thus, the link layer delay d accounts for MAC transmission backoff delay, channel dependent propagation and transmission delay, delays due to packet retransmission, and the sensing/switching operation, which may be performed before the transmission step. We set the reward equal to 0 in case of packet drop because the MR exceeds the maximum number of retransmissions (which is equal to 7 in the MAC 802.11 DCF standard).

2) State Values: The value of the state $V^i_t(s)$ gives the long term reward for the MR i to use spectrum s at the current time t. The value of $V^i_t(s)$ is updated after each a_{tx} action in the following way:

$$V^i_{t+1}(s) = V^i_t(s) + \gamma \cdot \big(r(a_{tx}) - V^i_t(s)\big) \tag{2}$$

where γ is a parameter regulating the speed of learning by scaling down the earned reward $r(a_{tx})$. Hence, $V^i(s)$ is also a function of the average link-layer delay to transmit a packet successfully on the channel s by the MR i.

3) Q-values used by the RL Algorithm: For each MR i and state s, the policy $\pi^i(s, a)$ gives the how the next action is chosen, given an existing probability distribution over the three possible actions in A. The higher the rate of the PU activity (implying frequent on-off switching) on a channel s, the more is the variation in the state value for MR i. The *eligibility trace* ($e^i(s)$) is a measure of the availability of the licensed channel for opportunistic transmission, and increasing activity results in greater value of $e^i(s)$. Note that $e^i(s)$ is updated every time a sensing operation (`sense` phase) is undertaken, and may vary with time. This trace is updated as follows:

$$e^i(s) = e^i(s) + \phi \cdot (O_s - e^i(s)) \tag{3}$$

Here, O_s is the outcome of the sensing action, and is equal to 1 if the PU was detected in the sensing interval T_s. We observe that $0 \leq e^i(s) \leq 1$ and that $e^i(s)$ is discounted of a factor ϕ each time the sensing action finds the channel free from PU activity. Based on $V^i(s)$ and on the eligibility trace $e^i(s)$, we define the Q-update rules for each action.

For the sensing action a_{se}, the Q-values reflects the amount of time in which the CR-cluster is blocked for "effective sensing", meaning that no node in the cluster is allowed to transmit. After a_{se} occurs, the MR i updates the Q-values as:

$$Q^i_{t+1}(s, a_{se}) = Q^i_t(s, a_{se}) + \gamma \cdot e^i(s) \cdot \big(T_s - Q^i_t(s, a_{se})\big) \tag{4}$$

The term $\big(T_s - Q^i_t(s, a_{se})\big)$ constitutes the Temporal Difference (TD) error of the learning process [2]. We weight the TD-error through the eligibility trace $e^i(s)$, in order to gradually increase or reduce the amount of sensing based on effective PU detection.

Similarly, we define the update rule for the a_{tx} action as follows:

$$Q^i_{t+1}(s, a_{tx}) = Q^i_t(s, a_{tx}) + \gamma \cdot \big(1 - e^i(s)\big) \cdot \big(T_u - Q^i_t(s, a_{tx})\big) \tag{5}$$

Here, $0 \leq T_u \leq t_{tx}$ represents the amount of time during which MR i was involved in *successful* transmissions or was idle (i.e. no packets to transmit), in the last t_{tx} interval. The intuition is that MR i can increase the transmission activity if it does not experience interference problems, and if the PU activity is low on the current channel s.

Finally, action a_{sw} forces the current cluster to switch to another channel $\tilde{s}$ that has lower access delay, over a given convenience threshold θ. The convenience threshold θ allows us to (*i*) account for the cost of handoff operation required for cluster re-configuration, e.g. t_h, and to (*ii*) reduce the frequency of cluster handoff only to cases when convenient. We define the update rule for the a_{sw} action as follows:

$$Q^i_{t+1}(s, a_{sw}) = max_{1 \leq \tilde{s} \leq M} \left(V(\tilde{s}) - V(s) - \theta\right) \tag{6}$$

In case of channel switching, the next channel is decided by ϵ-greedy probability, i.e. with probability ϵ MR i chooses the channel $\tilde{s}$ which maximizes equation 6. The ϵ factor decides the trade-off between exploration and exploitation action, and it can be estimated as described in section IV-B.

Based on the Q-value for action $a \in A = \{a_{se}, a_{tx}, a_{sw}\}$ defined by Equations 4, 5 and 6, the current policy for an agent i and state s is derived by using the *soft-max* action selection method:

$$\pi(s, a) = \frac{e^{Q(s,a)}}{Q^i_t(s, a_{se}) + Q^i_t(s, a_{tx}) + Q^i_t(s, a_{sw})} \tag{7}$$

The complete learning scheme is described by Algorithm 1. The MR i chooses one the three actions in A based on the current value of the policy π. In case of sensing, it updates the eligibility trace using Equation 3. In case of transmission, it updates the current value of the state $V(s)$ by Equation 2 based on the actual delay. Then, it updates the Q-values using Equations 4, 5 and 6, and starts a new round. From Algorithm 1 we observe how the learning scheme dynamically adjusts the actual policy π. If the PU is low or absent on a given channel, then the MR gradually reduces the frequency of sensing, and increases the probability of transmission. Also, the eligibility trace allows to differentiate between interference caused by PUs and other mesh nodes. As soon as PU is detected, then the probability of sensing is increased, based on the eligibility trace. The sensing-induced delay decreases the state-value of a channel ($V(s)$), and thus increases the probability to choose a different channel.

Algorithm 1 Learning-based algorithm

```
for each time step $t$ do
  decide next action $a$ through the policy $\pi$
  if $a$ == $a_{se}$ then
    update eligibility trace $e^i(s)$ through Equation 3
    update $Q(s, a_{se})$ through Equation 4
  end if
  if $a$ == $a_{tx}$ then
    update state value $V(s)$ through Equation 2
    update $Q(s, a_{tx})$ through Equation 5
  end if
  if $a$ == $a_{sw}$ then
    decide next spectrum $\tilde{s}$ through $\epsilon$-greedy selection
    perform state transition to $\tilde{s}$
    return
  end if
  update $Q(s, a_{sw})$ through Equation 6
  update the policy $\pi$ through Equation 7
end for
```

B. Cooperative Learning

The sharing of information between nodes allows a device to assimilate a larger set of readings for state-value for the different channels in a short time duration. In this case, the factor ϵ that favors exploration may be reduced (Section IV-A3), and the information from the neighboring nodes may be used to populate the state-value entries of $V^i(s)$ for a given node i. This cooperative method also improves the probability of detection of the PU, as more readings are now available for estimating the presence of the PU. To this aim, each MR i periodically broadcasts an HELLO message on the R_{cont} interface, containing the $V^i(s)$ values, for each spectrum s.

When a MR i receives an HELLO message from MR j, it updates its own $V^i(s)$ for each channel $s \in S$:

$$V^i(s) = V^i(s) + \kappa(s, j) \cdot \big(V^j(s) - V^i(s)\big) \tag{8}$$

Here, the parameter $\kappa(s, j)$ defines the weight assigned to the cooperation, i.e. how much a neighbor node j can contribute to correct the current estimation of $V^i(s)$. It is reasonable that comparatively close MRs will experience similar spectrum conditions, because several parameters affecting the estimation of V-values, e.g. MR interference and PU activity are intrinsically position-dependant. For this reason, we propose a simple though effective formulation of $\kappa(s,j)$, based on the actual distance d between MR i and MR j, and on the maximum transmitting range on channel s (e.g. $TX_r(s)$):

$$\kappa = 1 - \frac{d(i,j)}{TX_r(s)} \tag{9}$$

We believe that the individual exploration performed by MR i, i.e. ϵ^i, can be bounded by the level of cooperation experienced by each node, i.e. the more the node relies on spectrum information from other nodes, the less it needs to select sub-optimal operations during switching. We define the average cooperation value $\bar{\kappa}^i$ as the average of $\kappa(s,j)$, for all the neighbors j and the channel s. Then, we set ϵ^i as $min\left\{(1-\bar{\kappa}^i), \epsilon_{min}\right\}$, where $\epsilon_{min} = 0.2$ in our experiments.

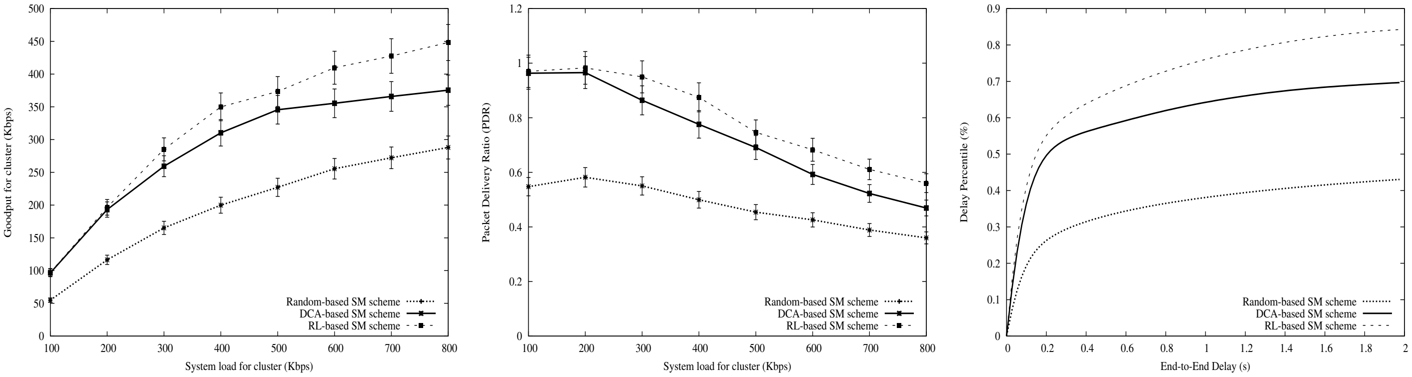
V. Performance Evaluation

In this section, we evaluate the performance of the proposed scheme using the ns-2 simulator, with the extension for CR networks [5]. Our WMN is composed by placing 16 MRs in the center locations of a square 4×4 node grid. There are 4 MCs associated with each MR, taking the total number of the nodes of the WMN to 80. Each MR-MC flow delivers UDP packets, produced by a Constant Bit Rate (CBR) application. We assume M=5 different primary bands, where primary band i supports data rate ϕ_i (expressed in Mb/s). Moreover, each band i can be occupied by a PU$_i$, whose activity is described by $< \alpha_i, \beta_i >$ parameters. Unless stated otherwise, we set $t_{tx} = 0.6\,\text{s}$, $t_s = 0.1\,\text{s}$ and $t_h = 0.001\,\text{s}$. Other parameters are: k=5, $D_{MAX} = 0.5s$ and $\gamma = 0.8$.

We compare the performance of our proposed scheme with the (*i*) random-based scheme, and the (*ii*) dynamic channel assigment (DCA) scheme. In the first case, each MR chooses randomly the next action, i.e. `transmit`, `sense` or `switch`, at the end of each transmission phase. Basically, the random-based scheme constitutes the configuration in which no learning is performed. In the DCA, each MR performs a sensing cycle, by alternating a `transmit` phase and a `sense` phase on the current channel. As soon as a PU is detected, then the MR switches its cluster to the less-interfered channel in its neighborhood, considering the number of mesh node interferers as selection metric. The DCA scheme constitutes a classical approach for spectrum management over cognitive WMNs. Figures 2(a), 2(b), 2(c) and 3(a) show the performance results of the three schemes, in a WMN with: $\phi = \{2,2,5,2,5\}$, $\alpha = \{0.01, 1.0, 1.0, 0.01, 1.0\}$ and $\beta = \{1.0, 0.01, 1.0, 1.0, 0.01\}$. Figures 2(a) and 2(b) show the goodput and packet delivery ratio (PDR) for cluster, as a function of the total load generated by MR-MCs connections in each cluster. The random-based scheme suffers of high packet losses, due to interference resulting from random `switch` actions. The DCA scheme avoids PU interference, but experiences suboptimal spectrum selection because it does not take into account the channel quality and the interference created by other mesh nodes, unlike in case where channel selection is based on the channel value (Equation 2). Figure 2(c) shows the delay percentile, in the configuration where the cluster load is equal to 400 Kbps. The values of the distribution can interpreted as the asymptotic probability of message arrival. Moreover, the distribution of delays reveals the ability of the RL-based approach in balancing the frequency of `transmit` and `sense` actions. The performance gain is also confirmed by Figure 3(a), which shows the average end-to-end delay as a function of the cluster load. Figure 3(b) analyzes the impact of sensing-induced delay on MCs performance for a single cluster. We consider a configuration with $\phi = \{2,2,2,2,2\}$ and no active PU in the WMN. Figure 3(b) shows the end-to-end delay as a function of the sensing time t_s on the x-axis. Both the random and the DCA schemes perform `sense` actions regardless of the PU activity on that channel. In the DCA scheme, each MR alternates between `sense` and `transmit` actions. In case of low PU activity, this traduces an additional overhead, which may have an impact on e.g. TCP performance [5]. In the proposed RL-based scheme, each MR adjusts the sensing frequency (Equation 4) based on the amount of PU activity detected on the current channel (Equation 3). In case of low PU activity, the sensing frequency is reduced to exploit the channel opportunities, and this explains the performance gain shown by Figure 3(b). Conversely, in case of intensive PU activity, the RL-based scheme increases the probability of sensing, in order to protect PU receivers. This is demonstrated by Figure 3(c), which shows the average interference experienced by a PU receiver, as a function of the PU OFF time on the x-axis. We consider the multi-cluster WMN topology described above. For all the schemes, the average interference decreases with the reduced PU activity. Intuitively, the random-based scheme incurs in the highest probability to interfere with PU activities, while the DCA scheme provides the highest PU protection, because each MR immediately switches to a new channel each time a PU is detected on the current channel. In the RL-based approach, the sensing frequency is dynamically adjusted based on the amount of PU activity through the eligibility trace function (Equation 3). The difference with the DCA approach can be explained by the fact that in our approach each MR uses a probabilistic policy π over the set of actions. Though the randomization given by Equation 7 allows choosing suboptimal actions, Figure 3(c) shows that the impact is quite limited. Thus, our RL-based approach provides a similar impact on PU receivers than the DCA approach, but with a consistent performance gains for CR users.

VI. Conclusion

In this paper, we have addressed the problem of spectrum management in cognitive radio wireless mesh networks (WMNs), by using a multi-agent reinforcement learning scheme. Our approach allows each MR to determine the spectrum for its operation, and decide the optimal frequency

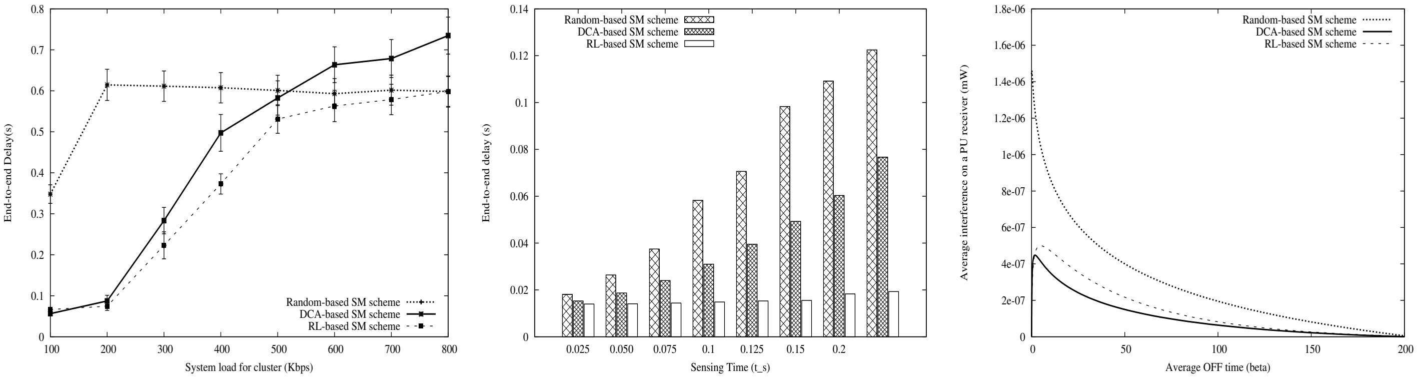


(a)

(b)

(c)

Fig. 2. Figures 2(a) and 2(b) show the per-cluster goodput and packet delivery ratio (PDR), as a function of the cluster load. Figure 2(c) shows the percentile on end-to-end delay in a configuration with cluster load equal to 400Kbps.



(a)

(b)

(c)

Fig. 3. Figure 3(a) shows the average end-to-end delay, as a function of the cluster load. Figure 3(b) shows the average end-to-end delay, as a function of the sensing time t_s, when no PUs operate in the licensed bands. Figure 3(c) shows the average interference (in mW) experienced by a PU receiver.

for sensing the channel or transmitting data for the mesh cluster. An important contribution also lies in the cooperation framework that allows the MRs to gather information faster from other MRs, and thereby improve the accuracy of the decisions. The simulation results reveal significant end-to-end performance gain for WMNs, but does not come at the expense of PU protection. We plan to investigate the relationship between cooperation and individual exploration as future work.

VII. Acknowledgment

This work is partially supported by University of Bologna and Italian MIUR funds.

References

[1] I. F. Akyildiz, W. Y. Lee, M. C. Vuran, and S. Mohanty. NeXt Generation/Dynamic Spectrum Access/Cognitive Radio Wireless Networks: A Survey. *Computer Networks Journal*, 50(1), pp. 2127- 2159, 2006.

[2] A.G. Barto and R. Sutton. In *Reinforcement Learning: An Introduction*, MIT Press, Cambridge 1998.

[3] U. Berhold, F. Fu, M. Van Der Schaar and F. K. Jondral. Detection of Spectral Resources in Cognitive Radios Using Reinforcement Learning. In *Proc. of IEEE Dyspan*, pp. 1-5, Chicago, 2008.

[4] D. Cabric, S. M. Mishra, and R. W. Brodersen. Implementation issues in spectrum sensing for cognitive radios in *Proc. of IEEE ACSSC*, pp. 772-776, Pacific Grove, 2004.

[5] M. Di Felice, K. R. Chowdhury and L. Bononi. Modeling and performance evaluation of transmission control protocol over cognitive radio ad hoc networks. In *Proc. of ACM MSWIM*, pp. 4-12, Tenerife, 2009.

[6] FCC, Spectrum policy task force report - In the Matter of Unlicensed Operation in the TV Broadcast Bands: Second Report and Order and Memorandum Opinion and Order. *ET Docket No. 08-260*, 2008.

[7] W. Y. Lee and I. Akyildiz. Optimal Spetrum Sensing Framework for Cognitive Radio Networks. in *IEEE Transactions on Wireless Communication*, 7(10), pp. 3845-3857, 2008.

[8] S. M. Mishra, A. Sahai, and R. W. Brodersen. Cooperative sensing among cognitive radios. in *Proc. of IEEE ICC*, 4(1), pp. 1658-1663, Istanbul, 2006.

[9] Meraki Project: Wireless Networks that simply work. http://meraki.com/

[10] H. Skalli, S. Ghosh, S.K. Das, L. Lenzini and M. Conti. Channel assignment strategies for multiradio wireless mesh networks issues and solutions. in *IEEE Communications Magazine*, 45(11), pp. 86-95, 2007.

[11] B. Wahab, Y. Yang, Z. Fan and M. Sooriyabandara Reinforcement Learning Based Spectrum-aware Routing in Multi-hop Cognitive Radio Networks. In *Proc. of IEEE CROWNCOM*, Hannover, 2009.

[12] L. Yang, L. Cao and H. Zheng. Physical Interference Driven Dynamic Spectrum Management. in Proc. of *IEEE Dyspan*, Chicago, 2008.

[13] K.-L. A. Yau, P. Komisarczuk and P. D. Teal. A Context-aware and Intelligent Dynamic Channel Selection Scheme for Cognitive Radio Networks. In *Proc. of IEEE CROWNCOM*, Hannover, 2009.

[14] Q. Zhao, L. Tong, A. Swami, and Y. Chen. Decentralized cognitive MAC for opportunistic spectrum access in ad hoc networks: A POMDP framework. in *IEEE JSAC*, 25(3), pp. 589- 600, 2007.