Design and Implementation of Distributed Dynamic Spectrum Allocation Protocol

Rami Hamdi, Mehdi Ben Ghorbel, Bechir Hamdaoui[†] and Mohsen Guizani
Qatar University | {rami.hamdi,mahdi.benghorbel,mguizani}@qu.edu.qa
[†] Oregon State University | hamdaoub@onid.orst.edu

***Abstract*—In this paper, we design, implement, and evaluate a resource allocation protocol for distributed dynamic spectrum access (DSA). This protocol, based on learning techniques, enables users to locate and exploit unused spectrum opportunities effectively. It relies on private objective functions in order to allow users to maximize their achieved reward/throughput from accessing the DSA system. This protocol, implemented and tested using ns3, considers that users ending up selecting the same spectrum band share the band equally among themselves by means of a carrier sense multiple access (CSMA) scheme. We use ns3 to implement our proposed protocol, thus allowing us to evaluate its performance while taking into account various practical implementation aspects and constraints, such as packet collision due to medium access contention, traffic overhead due to information sharing among users, and errors due to estimation models. Using simulations, we show the impact of several practical aspects on the performance of proposed protocol.**

Index Terms: Distributed dynamic spectrum access, protocol design, learning and adaptive techniques, private objective functions, carrier sense multiple access.

I. Introduction

In recent years, the success and emergence of wireless technology witnessed a rapid increase in the number of wireless devices, networks and applications, resulting thus in an increased demand for bandwidth resources. This increased demand in bandwidth led to a shortage in the wireless spectrum. On the other hand, reports from FCC [1] show that parts of the spectrum are still under-utilized. As a result, dynamic spectrum access (DSA) emerges as a potential solution for overcoming the spectrum shortage problem. During the past few years, significant research work has focused on DSA to prevent unbalanced spectrum utilization [2], [3], [4] and [5]. For instance, Akyildiz *et al.* [2] propose techniques that aim to help spectrum users to find good spectrum opportunities quickly, thereby improving spectrum utilization efficiency. Developing decentralized algorithms to promote DSA has been a challenge for researchers. Distributed algorithms are of a great paramount importance to DSA networks because they scale well and require less coordination among spectrum users.

Learning-based techniques [6] are shown to be good candidates for enabling distributed DSA, as they can easily be implemented in a decentralized manner. These techniques do not require users to have prior knowledge of the dynamics and characteristics of the environment, yet can still achieve good performances. There have been some efforts that aim to use learning techniques to promote effective DSA [7], [8], [9] and [10]. The focus of some of these works is on the derivation of objective functions that enable DSA users to assess, locate, and exploit unused spectrum opportunities effectively. By doing so, they maximize the total average rewards that the spectrum users can achieve in the long run.

NoroozOliaee *et al.* have, in [7] and [8], proposed and analyzed a complete framework for distributed DSA based on learning. In this work, they have considered an elastic traffic model where users' level of satisfaction is essentially proportional to the amount of reward (e.g., throughput) they receive from accessing the DSA system. A learning algorithm (e.g. Q-learner [6]) can be applied by each user independently to identify the best spectrum opportunities. The focus of this work was to propose and analyze the performance of efficient objective functions that can be used to maximize spectrum users' received rewards on the long run. The authors show that the proposed objective function achieves near-optimal performance while keeping a scalable complexity.

In this paper, we design and simulate a resource allocation protocol for DSA networks based on the distributed techniques proposed in [7] and [8]. We implement the proposed protocol in ns3 [11] and evaluate its performance while varying and studying the impact of various practical aspects that arise from the nature and characteristics of the DSA environment. The design and implementation tasks are very challenging because the protocol must be implementable by considering and accounting for the practical aspects of the DSA system while also incorporating the key theoretical concepts of the proposed techniques. Examples of practical aspects that our protocol considers are: traffic overhead due to control message exchanges, data collision arising from the contention nature of the wireless medium, and erroneous received reward/throughput estimation due to the unequal sharing of spectrum resources among users.

The rest of this paper is organized as follows. In Section 2, we revisit the theoretical system model and briefly describe the main results. In Section 3, we describe the protocol design and implementation method while paying special attention to the implementation and practical aspects. In Section 4, we evaluate the performance of the proposed protocol using ns3. Finally, concluding remarks are given in Section 5.

II. System Model

Consider a cognitive network with n secondary users (SUs) sharing m data channels (DCs) obtained from an equal division of the total available spectrum. Each DC is associated with a number of primary users (PUs) that have exclusive right to access it. So, upon the detection of any PU activities, an SU must immediately vacate the DC. We refer to any group of two or more SUs who want to communicate together as an agent.

We assume a static and fully connected network. That is, the SUs are assumed to enter and leave the system all at the same time, and all each interferes with one another. Each DC j offers an amount of service V_j (e.g., channel bandwidth in Mbps). The SUs that select the same DC will share it according to a carrier sense multiple access (CSMA) scheme [12], and the amount of service/reward (e.g., received throughput) is assumed to be shared equally among interfering users. An elastic traffic model is considered where the SU reward increases proportionally to the amount of service received from using the DC. But, it decreases exponentially when the received service drops below a threshold R. The instantaneous received reward for a user i that selected a DC j is written as

$$r_i(t) = \begin{cases} \frac{V_j}{|S_i(t)|+1} & \text{if} |S_i(t)| + 1 \le V_j/R \\ R\exp\left(-\beta\frac{(|S_i(t)|+1)R-V_j}{V_j}\right) & \text{otherwise,} \end{cases} \tag{1}$$

where $S_i(t)$ is the set of users interfering with user i and β is a reward decaying factor. Each SU implements a Q-learning algorithm [6] to find the best spectrum band for its transmission. The objective is to design an efficient objective function $g_i(t)$ that can be used by the learning algorithm to maximize the users' received reward.

Starting from observations made on the performance of the two most obvious objective functions which are the intrinsic reward ($g_i(t) = r_i(t)$) and the global reward ($g_i(t) = \sum_{i=1}^{n} r_i(t)$), NoroozOliaee *et al.* in [7] proposed an objective function representing the effect of only the user itself on the global reward by taking out the effect of other users. Explicitly, this objective function is written as

$$D_i(t) = \sum_{k:i\in S_k(t)} r_k(S_k(t)) - \Big(\sum_{k:i\in S_k(t)} r_k(S_k(t)-\{i\})\Big) + r_i(S_i(t)) \tag{2}$$

In our context, this expression can be simplified profiting from the fully connected network characteristics and the equal share assumption as

$$D_i(t) = (|S_i(t)|+1)r_i(S_i(t)) - |S_i(t)|r_i(S_i(t)-\{i\}) \tag{3}$$

In Fig. 1, it has been shown that this function achieves better performance than the other two obvious reward functions. In addition, this function can easily be implemented in a decentralized manner as it depends only on the number of interfering users.

III. Protocol Design and Implementation

Our aim in this work is to design a DSA protocol, based on the techniques described in Section 2 above, that enable SUs to locate and use DSA opportunistically.

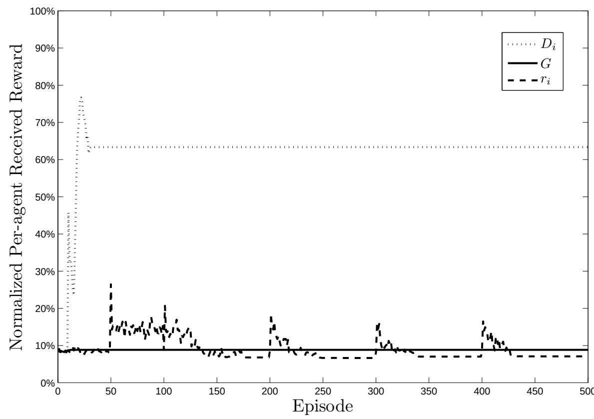

Fig. 1. Per-agent average achieved reward under three objective functions: $R = 2$, $\beta = 2$, $V_j = 20$ for $j = 1, 2, ..., 10$ and $n = 300$.

A. Protocol Description

The proposed protocol divides time into episodes. Each time episode consists of three window durations: select window (SelWin), data communication window (DCWin), and update window (UpWin). Events occurring during each of these phases are briefly described as follows:

- **Select Phase:**
  Each SU stores a table Q (initialized to zero) containing m elements which correspond to the available DCs. The SUs use the epsilon-greedy strategy to pick their action at each time step. It selects a random DC with probability ϵ and chooses the best DC which corresponds to the index of the highest value in the table Q with probability $1-\epsilon$. The SU uses the selected DC until the end of the episode.
- **Data Communication Phase:**
  All SUs turned to the same DC use CSMA as the access method to share the DC for data communication. Each SU has a random access time on the shared data channel DC. It verifies the absence of other traffic (by other users) before transmitting its packets using a feedback from the receiver. If a carrier is sensed, the SU waits for the transmission in progress to finish before initiating its own transmission.
- **Update Phase:**
  At the end of the DCwin, each SU i updates its Q-table using the chosen objective function $g_i(t)$ as

  $$Q(j) = (1-\alpha)Q(j) + \alpha g_i(t), \tag{4}$$

  where j is the selected DC for user i. The challenge consists of evaluating the objective function which, as shown above, depends only on the number of interfering users. Using the hypothesis of "equal share", the number of interfering users can be estimated by dividing the total amount of service offered by the band over the amount of service received by the user ($V_j/\hat{r_i}(t)$).

B. Implementation Challenge

In the protocol described above, it is clear that the most challenging task lies in the evaluation of the objective function. In fact, the reward function can be written as a function of the number of interfering users only under the assumption of an exactly equal share of the band service among all interfering users. But, in practice, a CSMA scheme results in a random access of the interfering users, enabling users to receive equal share on average, but users' instantaneous (at each time episode) shares may deviate from one another. Thus, the received reward of each user can not be estimated correctly, impacting the objective function as well.

For the intrinsic objective function, each user needs to evaluate only its reward function. Thus, the estimation problem can be solved by using directly the measured throughput as the received reward for that user. For the other objective functions, rewards of the other users are required to be able to compute the objective functions. While the global objective function can be evaluated using a centralized system where all users are assumed to exchange information about their received throughput, the difference reward cannot be exactly evaluated even when using a centralized approach, as the second term of the expression involves a virtual expression (by assuming the actual user is absent) which needs to be estimated.

IV. Performance Evaluation

The performance of the proposed protocol is evaluated using ns3. We consider the per-agent average received reward at each time episode as the performance metric to evaluate the effectiveness of the proposed protocol by measuring the throughput for each node as:

$$\overline{r}(t) = \frac{\sum_{i=1}^{n} r_i(t)}{n} \tag{5}$$

A. Estimation of the Reward Function Parameters

In the elastic traffic model, the SUs' received rewards are expressed following Equation 7 as a function of the amount of service (throughput), threshold R, and decaying factor β. In order to estimate R and β, we set a CSMA scheme with n nodes exchanging packets with an UDP (user datagram protocol) echo server (the clients will receive from the server what they sent to it) via one DC. We show for the sake of illustration the reward r_i as a function of the number of nodes n. We repeat this data communication for multiple time episodes and vary the number of nodes. We fix the global capacity (Uplink and Downlink) to $V = 20$ Mbps. At the end of the episode, each node measures the throughput it receives for using the DC. In Fig. 2, we plot first the instantaneous rewards received by the first node over different number of interfering nodes and the average rewards over time. From this figure, we deduce that each node does not receive a throughput exactly equal to V/n but on average it is. The difference of the received rewards between users sharing the CSMA DC is due to the random access of the channel using CSMA as explained earlier. Also, it can be shown that only 80% of the total capacity V of the DC ($\hat{V} = 80\% V$) is exploited due to the overhead and the back-off algorithm (i.e., the nodes exchange Address Resolution Protocol (ARP) and Internet Control Message Protocol (ICMP) packets).

The curve showing the average received reward over the SUs is to be used to estimate the parameters R and β. Thus, we get the analytical function of the average received reward function of the number of interfering users. An iterative algorithm can be implemented at the beginning of the learning algorithm with sample data to estimate these parameters.

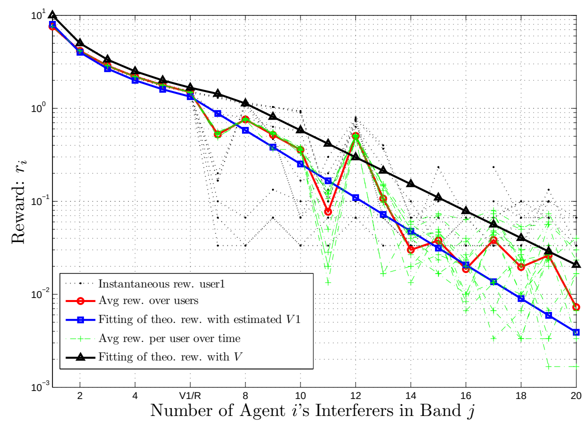

Fig. 2. Elastic reward function: analytic and simulation, $\hat{V}$/R=6.

B. Result Analysis

We simulate an UDP echo client server protocol. Each SU (client) generates random sessions, each of size Z bytes selected from a uniform distribution with mean $\overline{Z}$ and coefficient of variation δ_Z. The server sends what it receives from the clients back to the clients. The packet length is set to L bytes. We assume that PUs may return anytime to the DC, and have priority to use their own DC upon their return. We let m_p denote the average PU load. Finally, we have to choose the duration of each phase "Selwin", "Upwin", and "DCwin". The parameters used in the simulation are shown in Table I.

Now, we evaluate and compare the performances using the difference objective functions $g_i = D_i$ in terms of the per-agent average achievable rewards against those achievable using the intrinsic reward functions $g_i = r_i$.

In Fig. 3, we compare the performance using ns3 simulations to the theoretical performance given in [7]. Theoretically, even though the performance achievable with the proposed objective function D_i reaches as low as 60% of the optimal achievable reward when $n = 100$, it is still much higher than the performance obtained with the intrinsic function, r_i.

TABLE I
SIMULATION PARAMETERS.

Symbol	Description	Value
n	number of SUs	100
m	number of DCs	5
V	Capacity of each DC	20 Mbps
L	Length of packet	1250 Bytes
m_p	average load due to PUs	0%
ϵ	Coeff. probability	0.05
R	threshold	$\hat{V}/6$
β	reward decaying factor	2.5
α	learning rate	0.5

Using the ns3 implementation, we observe different behaviors of the performance obtained when using the proposed objective function D_i. Note that using the intrinsic reward function r_i, the performance is even better than the theoretical performance, profiting from the update of the Q-table with the exact received throughput. The lack of synchronization among users results in a time diversity which enhances the performance obtained when using this objective function.

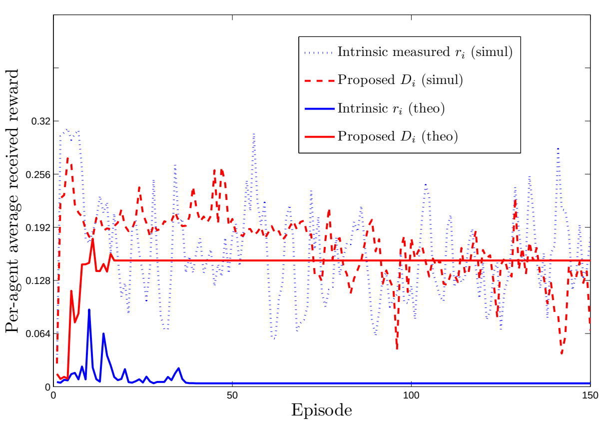

Fig. 3. Per-agent average achieved reward as a function of time episodes t under the two objective functions r_i and D_i: analytic and simulation results.

In Fig. 4, we compare performance achieved under the proposed distributed protocol, as described in Section 3, to that achievable under a centralized protocol where the correct number of interferers is assumed to be known to each user through server feedback. Thus, each node computes the objective functions r_i and D_i of this received value using Equations 3 and 7, and uses it to update its Q-table. We observe that the performance achievable under r_i is worse than that under the measured throughput which confirms our expectations on the effect of the estimated error; the computed r_i returns the average reward, which is different from the instantaneous one. On the other hand, the exchange of information helps improve the performance when using the proposed objective D_i, due to having the correct value of the number of interferers.

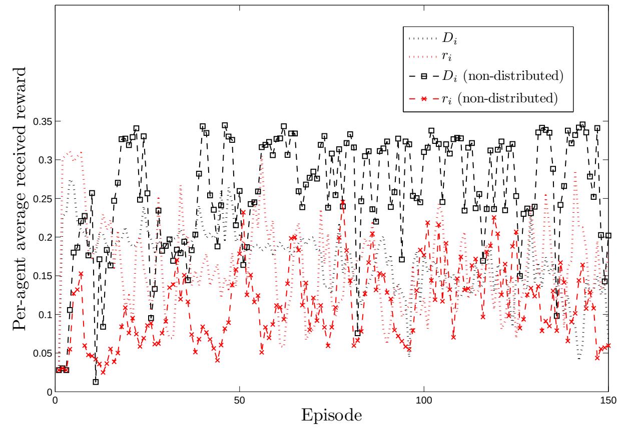

Fig. 4. Per-agent average achieved reward as a function of time episode t computed using fully distributed and non-distributed algorithms.

C. Impact of PUs' Activities

We now study the impact of the presence of PUs on the performance of the protocol under the proposed D_i function. We run simulations for different values of the average PU load, m_p. We consider the performance of the protocol for three network scenarios:

- $m_p = 0\%$ on each DC. PUs are not present.
- $m_p = 30\%$ on each DC. PUs are present, provided that they generate a total traffic load of 30%.
- $m_p = 60\%$ on each DC. PUs are present, provided that they generate a total traffic load of 60%.

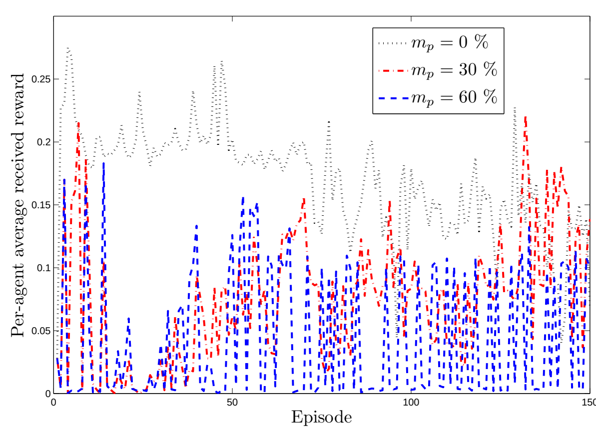

Fig. 5. Per-agent average achieved reward with and without the presence of PUs.

In Fig. 5, we show that the achievable rewards reach zero when primary users are present, but quickly go up to higher values as soon as the PUs leave their bands. Also, as expected, when the PU traffic load increases, the total achievable average reward decreases, since rewards will also be taken away by the PUs themselves.

D. Adaptive Service Model

Now, we study an adaptive service model [10] that can be used by SUs to compute the rewards they receive from using the DSA system. This proposed service model complements the objective functions by enhancing the amount of service that each SU receives in the long run. We consider that each agent i has a total required Level of Service (LoS), Q_{total}, that should be received by a given target time period T. We assume that the required LoS at time step t, $Qi(t)$, changes adaptively based on the required LoS that has not been received yet and the target period T. Formally,

$$Q_i(t) = \frac{Q_{total} - \sum_{t'=1}^{t} A_i(t')}{T - t} \tag{6}$$

The reward of agent i, $r_i(t)$, at time t can be written as

$$r_i(t) = \begin{cases} A_i(t) & \text{if} A_i(t) \geq Q_i(t) \\ Q_i(t) \exp\left(-\beta \frac{Q_i(t) - A_i(t)}{A_i(t)}\right) & \text{otherwise,} \end{cases} \tag{7}$$

where $A_i(t)$ denotes agent i 's received LoS at time t.

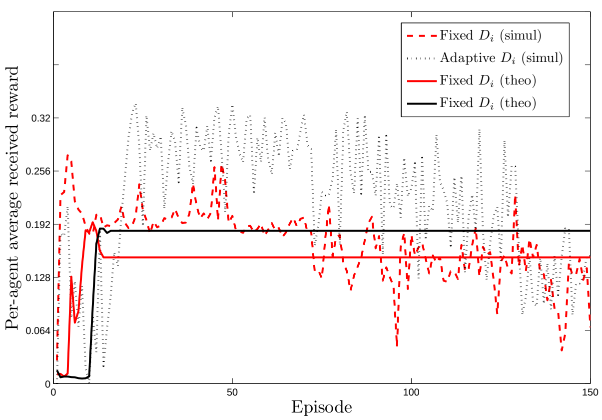

Fig. 6. Per-agent average achieved reward: adaptive and fixed model.

In Fig. 6, we depict the performance of the proposed protocol in terms of the per-agent/SU average achieved reward when using the adaptive and fixed models. The figure shows, for each model, the theoretical and simulated results. We observe that the simulated adaptive model outperforms the fixed model under the objective function D_i.

V. Conclusion

In this paper, we designed a protocol for distributed DSA systems. The efficiency of the proposed protocol is evaluated using ns3. The simulation results show that the theoretical model does not consider some practical aspects of real networks since it does not assume the overhead of information sharing among users, the duration of selection and update phases, the collision between packets and especially the inequality of the received throughput among interfering users. All these factors impact the performance of the proposed protocol and thus should be taken into consideration.

Acknowledgment

This work was made possible by NPRP grant # NPRP $5-319-2-121$ from the Qatar National Research Fund (a member of Qatar Foundation). The statements made herein are solely the responsibility of the authors.

References

[1] Federal Communications Commission, *"Spectrum policy task force,"*, Report ET Docket, Nov. 2002.

[2] Ian F. Akyildiz, Won-Yeol Lee, Mehmet C. Vuran and Shantidev Mohanty, *"Next generation/dynamic spectrum access/cognitive radio wireless networks: A survey"*, Computer Networks Journal Elsver, vol. 50, pp. 2127Ű2159, 2006.

[3] B. Hamdaoui and K.G. Shin, *"OS-MAC: an efficient MAC protocol for spectrum-agile wireless networks,"*, IEEE Transactions on Mobile Computing 7 (8) (2008) 915Ű930.

[4] Q. Zhao and Sadler B.M., *"A Survey of Dynamic Spectrum Access,"*, Signal Processing Magazine, IEEE (Volume:24 , Issue: 3), p.79 - 89, May 2007.

[5] Q. Zhao, L. Tong and Swami A., *"Decentralized cognitive mac for dynamic spectrum access,"*, New Frontiers in Dynamic Spectrum Access Networks, 2005. DySPAN 2005. 2005 First IEEE International Symposium.

[6] C. Watkins and P. Dayan, *"Technical note: Q-learning,"*, Machine Learning, vol. 8, no. 3-4, pp. 279Ű292, 1992.

[7] M. NoroozOliaee, B. Hamdaoui and K. Tumer, *"efficient objective functions for coordinated learning in large-scale distributed osa systems,"*, IEEE Transactions on Mobile Computing, May 2013.

[8] M. NoroozOliaee, B. Hamdaoui and K. Tumer, *"Achieving optimal elastic traffic rewards in dynamic multichannel access,"*, In Proc. of IEEE Conference on High Performance Computing and Simulation, July 2011.

[9] M. NoroozOliaee, B. Hamdaoui and M. Guizani, *"Maximizing Secondary-User Satisfaction in Large-Scale DSA System Through Distributed Team Cooperation,"*, IEEE Transactions on Wireless Communications, vol. 11, no. 10, October 2012.

[10] M. NoroozOliaee, B. Hamdaoui and M. Guizani, *"Adaptive Service Function for System Reward Maximization Under Elastic Traffic Model,"*, In Proc. of IEEE Global Communication Conference, December 2013.

[11] ns-developers, *"ns-3 Reference Manual,"*, ns-3-dev 20 August 2010.

[12] H. Kwon, H. Seo, S. Kim and B. Gi Lee, *"Generalized CSMA/CA for OFDMA Systems: Protocol Design, Throughput Analysis, and Implementation Issues,"*, IEEE Transactions on Wireless Communications, vol. 8, no. 8, August 2009.