Distributed Spectrum Management based on Reinforcement Learning

Francisco Bernardo, Ramon Agustí, Jordi Pérez-Romero and Oriol Sallent
Signal Theory and Communications Department
Universitat Politècnica de Catalunya (UPC)
08034 Barcelona, Spain
Email: [fbernardo, ramon, jorperez, sallent]@tsc.upc.edu

***Abstract*—This paper presents a novel distributed framework to decide the spectrum assignment in a primary cellular radio access network. The distributed nature of the framework allows each cell to autonomously decide (by means of machine learning procedures) the best frequencies to use in order to maximize spectral efficiency, preserve quality-of-service, and generate spectrum gaps, so that secondary cognitive radio networks can improve overall spectrum usage. The proposed distributed framework has been validated over a downlink multicell OFDMA radio access network, showing comparable performance results with respect to its centralized counterpart and superior performance with respect to fixed frequency planning schemes.**

***Index Terms*—Spectrum Management, Reinforcement Learning, Cognitive Radio, Self-organization, Autonomic Systems, OFDMA.**

I. Introduction

Current primary cellular networks are difficult to manage and require a lot of human interaction. For example, tasks such as assigning spectrum resources to cells (i.e., network planning to avoid intercell interference) are carried out off-line during network deployment and remain unaltered until new infrastructure is added to the system. Moreover, spectrum is allocated regularly among cells expecting to cover the maximum demand at any place of the service area. This spectrum assignment strategy is proved to be clearly inefficient with variable traffic demands. It could also become intractable with the advent of new technologies like *femtocells* [1] (small range base stations introduced at a considerable amount of random locations to increase coverage and capacity) or in the framework of new regulation scenarios like private commons [2], which may require a primary spectrum re-assignment in a dynamic and unpredictable manner to allow both the QoS guarantee for primary users and the release of spectrum chunks for secondary usage. Hence, it becomes necessary to include cognitive and autonomic capabilities in primary network elements to automatically reconfigure spectrum assignment and minimize human interaction. In this context, self-organization arises as a promising solution.

Self-organization is the ability of a system composed of several entities to adopt a particular structure and perform certain functions to fulfill a global purpose without any external supervisor or central dedicated control entity [3]. Intuitive examples of self-organization are swarms of ants looking for food, or schools of fish protecting against predators. In the field of radio access networks (RAN), self-organization can be applied to network planning, deployment, optimization and maintenance bringing operational and capital expenditures reductions [4]. Therefore, certain activities of several projects and standardization bodies (e.g., 3GPP, IEEE) are steered to study the automation of network procedures.

The main characteristics of a self-organized system are its *distributed* nature and the *localized* interactivity between system elements. That is, each entity performs its operation based only on the information retrieved from other entities in its vicinity. Hence, overall system's organization and performance is achieved from an *autonomous* behavior of each entity that, from the *experience* acquired from a variable environment, decides the proper *actions* to adapt to it. In this context, Reinforcement Learning (RL) arises as a potential approach to implement autonomic self-organizing procedures in each of the system entities [5]. RL shows inherent cognitive capabilities since it consists in learning the suitable set of actions to choose in order to maximize a numerical *reward* given that there is a continuous interaction with an environment. Hence, RL has been successfully applied to spectrum sensing [6] or spectrum sharing [7] procedures in Cognitive Radio (CR). Also, in our previous work, we showed that RL can be used to implement centralized dynamic spectrum assignment (DSA) strategies for primary cellular networks [8].

This paper presents a novel distributed framework for the spectrum assignment in a cellular RAN. Each cell behaves as an autonomous entity that executes a RL Dynamic Spectrum Assignment strategy (RL-DSA). The objective of the RL-DSA strategy is to maximize spectral efficiency per cell (in bits/s/Hz) while quality of service (QoS) of primary communications is preserved. In addition, if primary traffic demands are low enough, RL-DSA generates spectrum gaps in the cell to enable that secondary spectrum markets can make a cognitive opportunistic access and hence enhance overall

This work has been performed in the framework of the project E^3, which has received research funding from the Community's Seventh Framework programme. Also, the Spanish Research Council and FEDER funds under COGNOS grant (ref. TEC2007-60985) have supported this work. This paper reflects only the authors' views and the Community is not liable for any use that may be made of the information contained therein. The contributions of colleagues from E^3 consortium and the support of the Spanish Ministry of Science and Innovation via FPU grant AP20051165 are hereby acknowledged.

978-1-4244-3424-4/09/$25.00 ©2009 IEEE

spectrum usage. Thus, the proposed approach is a distributed self-organized framework aimed to ease the deployment of future RANs where it is expected to have lots of base stations, relay nodes, and femtocells. Moreover, it encompasses future CR applications, making the primary network aware that not used spectrum could be used by secondary networks.

The framework is validated over a primary downlink RAN based on Orthogonal Frequency Division Multiple Access (OFDMA), which is in the main stream of future RANs such as 3GPP LTE or WiMax. The proposed framework exhibits a performance in terms of spectral efficiency comparable with that offered by its centralized version and superior performance with respect to classical network planning strategies. Certainly, the distributed approach allows for much lower signaling load and implementation complexity than its centralized version.

The paper is organized as follows. Section II presents our distributed framework for spectrum management, including system model, cell functional scheme, and functionalities descriptions. Proposed framework is based on the RL-DSA algorithm, which each cell executes to decide the best spectrum assignment. Thus, section III presents the RL-DSA functional scheme and its detailed description and section IV details procedures of each component of the framework necessary for the RL-DSA execution. Section V is devoted to present the simulation model and results for two case studies: one compares the performance of the proposed framework with a centralized version and fixed network planning schemes, and the other shows the autonomous and adaptable behavior of the self-organized system by adding new cells that automatically decide their spectrum assignment regarding nearby environment. Finally, section VI states final conclusions.

II. Distributed Framework Description

The proposed distributed framework is depicted in Fig. 1. Fig. 1(a) shows the system model where an autonomous cell surrounded of other cells is depicted. Each cell performs autonomous spectrum assignment decisions with the objective of improving spectral efficiency while guaranteeing cell users' QoS. A generalized OFDMA radio interface is supposed in downlink, where a common system bandwidth W for the service area is divided into N *chunks* (i.e., groups of contiguous OFDM subcarriers). Moreover, time is divided into *frames* and then, the minimum radio resource block assignable to users is a specific chunk into a frame. There is an uplink control channel where users report their measurements in terms of signal-to-interference plus noise ratio (SINR) in the different chunks. As it will be explained in section IV, this is the way a cell obtains information about neighboring cells spectrum usage useful to estimate chunks capacities of the own cell. Hence no explicit coordination between cells is needed.

Fig. 1(b) depicts the cell functional scheme. Operation of the cell is divided into two timescales. In the short-term (i.e., at the frame time scale), the cell schedules users' transmissions into available frequency resources following standard scheduling strategies implemented in the *Short-Term Scheduler* (STS). On the other hand, in the medium-term (i.e., from tens of seconds to tens of minutes), the cell determines which frequencies should use and which not. To this end, a *Cell DSA Controller* is included in each cell. Other distributed [9] or semi-distributed [10] approaches have been found in literature for spectrum assignment. However, they perform dynamic spectrum assignment to users in the short-term, what may lead to high computational requirements, and in some cases may need a centralized coordinator to perform the spectrum assignment among cells. On the other hand, the proposed framework here is entirely distributed among cells and spectrum assignment is provided in medium-term considerably reducing execution time requirements and signaling overhead.

The core of the Cell DSA controller is the *RL-DSA algorithm*, which provides the final spectrum assignment for the cell. Once in execution, the RL-DSA continuously interacts with a *Cell Characterization Entity* (CCE) by exchanging

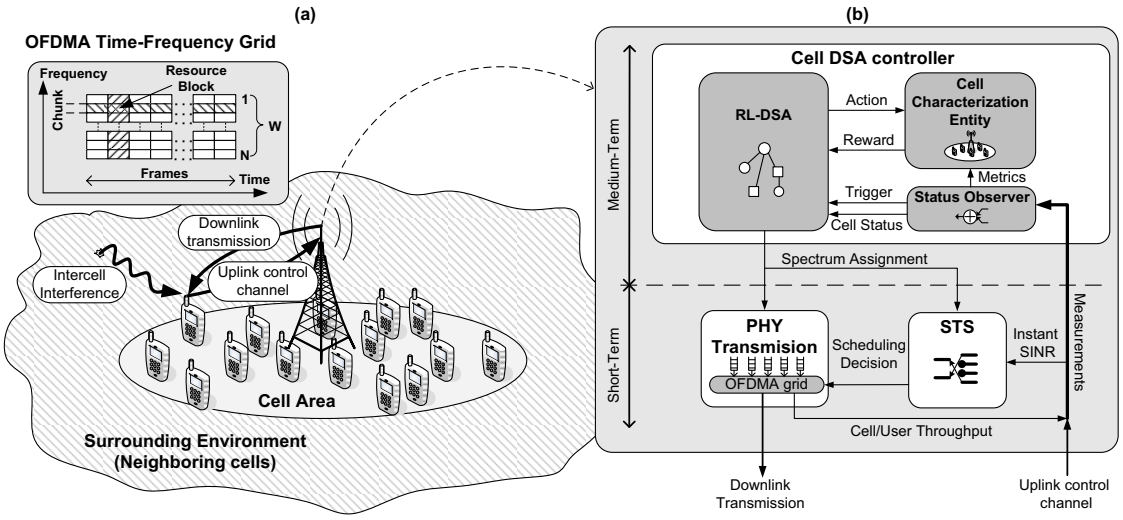

Fig. 1. Self-organization framework for spectrum assignment. (a) System model. (b) Cell functional scheme.

actions and rewards. Each action of the RL-DSA represents a candidate spectrum assignment for the cell. The CCE, which implements a model of the cell's behavior, returns a reward representing the suitability of a given action allowing the RL-DSA to *learn* the most appropriate spectrum assignment for the cell. The *Status Observer* is in charge of triggering the execution of the RL-DSA algorithm and providing current status of the cell (e.g., cell load). It also collects and builds the necessary metrics to cope with the iterative learning procedures.

III. RL-DSA ALGORITHM

Consider N available chunks (numbered from 1 to N) in a downlink OFDMA cellular system. Each cell should select a spectrum assignment defined as a binary $1\times N$ vector $\boldsymbol{\Upsilon}=(y_1,\cdots,y_N)$, where $y_n\in\{0,1\}$ denotes that $n-$th chunk is assigned to the cell if $y_n=1$ (and not assigned if $y_n=0$). The RL-DSA algorithm is based on the RL REINFORCE methods that assure the maximization of an average reward in the long-term [11]. Hence, we propose a feed-forward network composed of N RL REINFORCE agents (Fig. 2) to implement the RL-DSA algorithm in each cell. This network interacts with the CCE on a step by step basis maximizing average reward obtained from CCE. The RL-DSA is periodically triggered by Status Observer that provides current status of the cell as a constant input for RL-DSA execution. *Decision Maker* stops RL-DSA when it has converged by continuously examining RL-DSA status, and obtains the new spectrum assignment learnt.

The $n-$th RL agent in the feed-forward network is devoted to learn whether the $n-$th chunk should be assigned to the cell or not. To this end, the $n-$th RL agent's output $y_n(t)\in\{0,1\}$ in a step t is a Bernoulli random variable, so that each action taken by the set of agents in the network represents a candidate spectrum assignment $\boldsymbol{\Upsilon}(t)$. Knowledge of each RL agent is contained in parameter $p_n(t)$, which represents the probability that the output $y_n(t)$ is 1. Probability $p_n(t)$ depends on the current status of the cell x and a corresponding weight $w_n(t)$.

The first time that RL-DSA is run a random assignment is set, and in next executions RL-DSA algorithm begins from the assignment learnt in the previous execution so that the know-ledge acquired until that moment is retained. With these definitions RL-DSA procedure is given in the following:

1) For each RL step t, assignment $\boldsymbol{\Upsilon}(t)$ is obtained considering that the $n-$th chunk is assigned to the cell if the output $y_n(t)$ is 1.
2) For each assignment $\boldsymbol{\Upsilon}(t)$ the CCE returns a reward $r(t)$ that is used by the respective RL agents to update its internal weights for the next step as [11]

$$w_n(t+1) = w_n(t)+\Delta w_n(t), \tag{1}$$

$$\Delta w_n(t)=\alpha(t)\,(r(t)-\bar{r}(t-1))\,(y_n(t)-p_n(t))\,x. \tag{2}$$

Notice that the learning from reward is enforced in the weighting value. Parameter $\alpha(t)$ is called the learning rate.

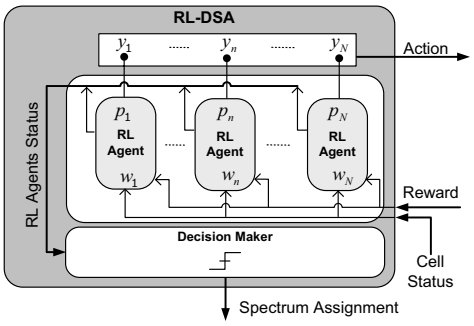

Fig. 2. RL-DSA algorithm functional scheme.

$\bar{r}(t)$ is the reinforcement baseline or average reward calculated using a exponential moving average with parameter β. Finally, in this paper, we use the current status x to modify the weight update $\Delta w_n(t)$ in accordance with the current load of the cell as follows

$$x = \max\left\{\overline{W}/W, \omega\right\} \tag{3}$$

where $\overline{W}$ is the amount of unused bandwidth in the cell and $0<\omega\ll 1$ is a parameter that simply assures a non-zero minimum value of x if $\overline{W}=0$.

3) Probabilities for the next step $p_n(t+1)$ are updated with the new weights $w_n(t+1)$ as

$$p_n(t)=\max\left\{\min\{1/(1+e^{-w_n(t)x}), 1-p_{\exp}\}, p_{\exp}\right\} \tag{4}$$

where probability $p_{\exp}$ is introduced as a small bias in order to enforce some exploratory behavior in the agent even if its internal probability $p_n(t)$ is very near to 1 or 0 (when approaching to algorithm's convergence).
4) Outputs $y_n(t+1)$ are obtained for next assignment from the random Bernoulli generators. Decision Maker keeps track of RL-DSA status (internal probabilities evolution and current step t). If it detects that the variation of all p_n between two successive steps is below ε during S steps, or if $t >$ MAX_STEPS, then phase 4) is executed. Otherwise, next assignment is tested from 1).
5) Decision Maker stops RL-DSA. It decides the spectrum assignment for the cell from the knowledge acquired by RL-DSA. Hence, it assigns (de-assigns) a chunk to the cell if p_n is greater (lower) than 0.5.

Speed of convergence of an RL algorithm can be increased by increasing the value of the learning rate. However, this compromises the algorithm's accuracy to converge to a correct action in a finite number of steps [12]. Thus, to provide some tradeoff between speed of convergence and accuracy, $\alpha(t)$ in (2) is linearly decreased as $\alpha(t)=\alpha(t-1)-\Delta$, where Δ is a factor that should be small enough to assure a smooth transition between steps. On the other hand, for high traffic load situations the number of suitable spectrum assignments is considerably reduced. Note that, under such conditions, the inclusion in (2) of the cell status x given in (3), improves the convergence to suitable solutions, since the learning rate $\alpha(t)$ is weighted by $x\ll 1$.

IV. RL-DSA Supporting Procedures

The RL-DSA algorithm in the Cell DSA controller implements the adaptable behavior of the proposed framework. The following procedures support RL-DSA execution and complete the functionalities description of the framework.

1. Short-Term Scheduler (STS). Short-term exploitation of multiuser diversity at the cell is carried out by the STS that dynamically assigns available chunks (assigned by Cell DSA controller) to users. The well-known Round-Robin (RR) strategy [13] has been retained as illustrative and implemented in this paper. Transmitted chunk power is supposed to be constant and users' transmission bit rate is variable by means of Adaptive Coding and Modulation (ACM). The detailed SINR thresholds for each modulation and coding rate considered are given in TABLE I. The instantaneous SINR ($\gamma_{m,n}$) for the $m-\text{th}$ user is computed for each active chunk n in the cell considering distance dependant pathloss and shadowing (both not frequency dependant), and frequency selective fast fading for both serving cell and interfering cells. $\gamma_{m,n}$ is reported in uplink to perform scheduling. Then, the $m-\text{th}$ user achievable bit rate for each chunk n is computed as $R_{m,n} = Bq(\gamma_{m,n})$, where B is the chunk bandwidth in Hz and $q(\gamma_{m,n})$ stands for the achievable spectral efficiency in bits/s/Hz for a given SINR threshold.

2. Status Observer. Inputs for the Cell DSA controller in each cell come only from local information and measurements reported by users from neighboring cells. The Status Observer entity collects those inputs, builds necessary metrics, and averages them over a period of l seconds. The local metrics used are the average number of users in the cell (U), average throughput per user (th), average user throughput in the cell (TH), and spectral efficiency defined as the aggregate user throughput in the cell per hertz (η). Additionally, the so-called average user dissatisfaction probability in the cell ($P^{T_{th}}$) is defined as the percentage of seconds in which th is below a target throughput T_{th}. On the other side, Status Observer computes for each chunk the Probability Density Function (PDF) $f_n(\gamma)$ of the average SINR $\overline{\gamma}_{m,n}$ estimated and reported by each user in the same chunk during a period of l seconds. $\overline{\gamma}_{m,n}$ can be computed as:

$$\overline{\gamma}_{m,n}=\frac{P_m}{I_{m,n}}, \tag{5}$$

$$I_{m,n}=\begin{cases} P_{Total,n}-P_m & \text{if chunk } n \text{ is used in the cell;} \\ P_{Total,n} & \text{otherwise.} \end{cases} \tag{6}$$

where P_m is the average received power of $m-\text{th}$ user from serving cell. $I_{m,n}$ is the average intercell interference plus thermal noise power per chunk n, and $P_{Total,n}$ the total measured power in the chunk. Control signaling to inform users about the chunks that are used in the cell is assumed. Notice also that P_m does not depend of the chunk in (5) and (6). This is because transmitted power per chunk is constant and average path loss and shadowing is assumed to be non-frequency dependent into the service bandwidth. Users estimate and report $\overline{\gamma}_{m,n}$ for all chunks $n = 1..N$.

Finally, the Status Observer entity in each cell is also responsible of triggering the RL-DSA algorithm in periods of L seconds.These periods are not aligned between cells, that is, RL-DSA is not executed simultaneously in the different cells.

3. CCE. The CCE constitutes the environment for the RL-DSA and tries to mimic the response of the cell for a given spectrum assignment. It returns the reward value reflecting the suitability of each candidate spectrum assignment $\boldsymbol{\Upsilon}(t)$ given by RL-DSA in step t. Since RL-DSA maximizes reward value in the long-run, a reward function that captures the performance of the cell for an assignment $\boldsymbol{\Upsilon}(t)=(y_1(t),\cdots,y_N(t))$ has to be defined. Reward value $r(t)$ is obtained as

$$r(t)=\begin{cases} 0 & \text{if } \widehat{TH}\left(\boldsymbol{\Upsilon}(t)\right)<T_{th}; \\ \lambda\widehat{\eta}\left(\boldsymbol{\Upsilon}(t)\right)+\mu\frac{\overline{W}(\boldsymbol{\Upsilon}(t))}{B} & \text{otherwise,} \end{cases} \tag{7}$$

where $\widehat{TH}\left(\boldsymbol{\Upsilon}(t)\right)$ and $\widehat{\eta}\left(\boldsymbol{\Upsilon}(t)\right)$ are estimations of average user throughput and spectral efficiency in the cell for a given spectrum assignment, respectively. $\overline{W}\left(\Upsilon(t)\right)$ is the bandwidth released in the cell for e.g., secondary usage and then $\overline{W}\left(\boldsymbol{\Upsilon}(t)\right)/B$ is the number of free chunks. λ and μ are positive weighting constants. Then the reward signal $r(t)$ reflects the suitability of the spectrum assignment for the cell in terms of spectral efficiency, QoS and released bandwidth. Note that reward is zero for $\widehat{TH}\left(\boldsymbol{\Upsilon}(t)\right)$ lower than QoS throughput T_{th}.

Released bandwidth $\overline{W}\left(\boldsymbol{\Upsilon}(\mathbf{t})\right)$ can be written as $\overline{W}\left(\Upsilon(t)\right)=W-B\left|\boldsymbol{\Omega}_{\Upsilon(t)}\right|$, where $\boldsymbol{\Omega}_{\Upsilon(t)}$ is the set of chunks assigned to the cell by the RL-DSA for a given action (i.e., $n\in\boldsymbol{\Omega}_{\Upsilon(t)}$ if $y_n(t)=1$) and $|\mathbf{X}|$ denotes cardinality of set $\mathbf{X}$. Hence, $B\left|\boldsymbol{\Omega}_{\Upsilon(t)}\right|$ is the assigned bandwidth. CCE estimates $\widehat{TH}\left(\boldsymbol{\Upsilon}(t)\right)$ and $\widehat{\eta}\left(\boldsymbol{\Upsilon}(t)\right)$ as follows

$$\widehat{\eta}\left(\Upsilon(t)\right)=\frac{1}{\left|\boldsymbol{\Omega}_{\Upsilon(t)}\right|}\sum_{n\in\boldsymbol{\Omega}_{\Upsilon(t)}}\int_{-\infty}^{\infty}q(\gamma)f_n(\gamma)d\gamma, \tag{8}$$

$$\widehat{TH}\left(\Upsilon(t)\right)=\frac{B\left|\boldsymbol{\Omega}_{\Upsilon(t)}\right|\widehat{\eta}\left(\boldsymbol{\Upsilon}(\mathbf{t})\right)}{U}, \tag{9}$$

where $q(\gamma)$ is the achievable spectral efficiency for a given SINR γ (TABLE I).

V. Simulation Model and Results

Results were obtained by means of dynamic simulations over a 7 hexagonal cells scenario representing a simulated time of 1 hour. A total of 6 chunks are available for the entire system. At the beginning 105 users are equally distributed among cells (i.e., 15 users per cell). Users move at 3Km/h with a random walk model [14] and always remain within their cell. A full-buffer traffic model is assumed. During the 10 minutes period between the minutes 25 and 35, 4 new sessions per minute are started in the central cell and one session per minute is stopped in the rest of cells. In this way, simulations consider both spatial and temporal variations of the traffic. Satisfaction throughput is set to T_{th}=128 kbps. More simulation parameters values including RL parameters are given in TABLE II.

TABLE I
MODULATION AND CODING SCHEMES

Modulation m (bits/s/Hz)	Coding Rate r (bits/s/Hz)	Spectral efficiency q (bits/s/Hz)	SINR threshold (dB)
2 (QPSK)	1/3	0.66	≥ 0.9
2 (QPSK)	1/2	1	≥ 2.1
2 (QPSK)	2/3	1.33	≥ 3.8
4 (16QAM)	1/2	2	≥ 7.7
4 (16QAM)	2/3	2.66	≥ 9.8
4 (16QAM)	5/6	3.33	≥ 12.6
6 (64QAM)	2/3	4	≥ 15.0
6 (64QAM)	5/6	5	≥ 18.2

TABLE II
SIMULATION PARAMETERS

Number of cells	$K = 7$
Cell Radius	$R = 500$ meters
Antenna patterns	Omnidirectional
Frame time	2 ms
Number of chunks	$N = 6$
Chunk bandwidth	$B = 375$ KHz
Power per chunk	$P = 33$ dBm
Path loss in dB at d km	$128.1 + 37.6log_{10}(d)$ [14]
Shadowing standard deviation	8 dB [14]
Shadowing decorrelation distance	5 m [14]
Small Scale Fading model	ITU Ped. A [14]
UE thermal noise	-174 dBm/Hz
UE noise factor	9 dB
RL parameters $[\alpha, \beta, \Delta, \omega]$	$[10, 0.01, 10^{-5}, 0.02]$
Exploratory probability p_{exp}	1%
Reward constants $[\lambda, \mu]$	$[100, 10]$
RL convergence criterion $[\varepsilon, S]$	$[10^{-4}, 5000]$
MAX_STEPS	100000
Measurements averaging window	$l = 10$ s
RL-Trigger period	$L = 60$ s

A. Case Study 1. Performance comparison

Performance of distributed framework is compared with classical frequency planning schemes (Frequency Reuse Factors (FRF)), such as FRF1 (6 chunks per cell) and FRF3 (2 chunks per cell). Also, a centralized version of the RL-DSA algorithm is simulated [8]. It is expected that this centralized strategy outperforms distributed RL-DSA thanks to its global vision of the spectrum assignment, but self-organized systems, and in particular the distributed RL-DSA presented here, aim at approximate centralized performance while scalability and autonomous capabilities are given to the system.

Fig. 3 depicts the average dissatisfaction probability, average spectral efficiency, and average user throughput fairness for the considered schemes. User throughput fairness reflects the balance between the throughput obtained by users in the system. To this end, we consider as fairness metric the 5-th percentile user throughput normalized to average throughput to allow fair comparison between spectrum assignment schemes.

Fig. 3(a) shows the dissatisfaction probability performance. It can be seen that both centralized and distributed RL-DSA schemes achieve the lowest dissatisfaction along simulation. On the other hand, both FRF schemes fail to guarantee dissatisfaction especially when the distribution of the traffic load is heterogeneous (from minute 35). These poor results

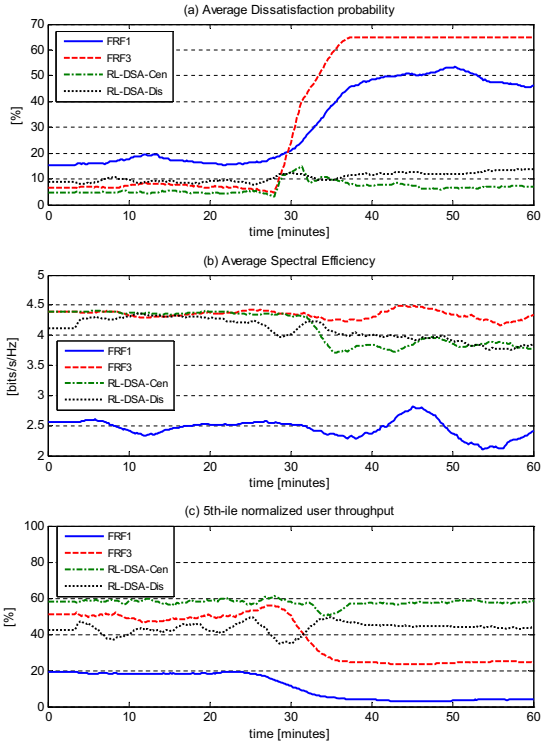

Fig. 3. Performance comparison. (a) Average user throughput fairness. (b) Average dissatisfaction probability. (c) Average espectral efficiency.

are also reported in terms of spectral efficiency (Fig. 3(b)) and fairness (Fig 3(c)), demonstrating that fixed frequency planning schemes are inefficient. For example, FRF1 shows the worst average spectral efficiency results because of intercell interference. Moreover, the high spectral efficiency of FRF3 from minute 35 is useless because dissatisfaction probability and fairness are unacceptable (60% of the users are dissatisfied and fairness dramatically decays).

In contrast, RL-DSA schemes demonstrate the best tradeoff between spectral efficiency, dissatisfaction probability and fairness, by dynamically selecting the proper chunks to cope with variable traffic demands. Comparing the distributed RL-DSA with the centralized version, it can be observed that distributed RL-DSA achieves a similar performance. Finally, regarding spectrum used by the RL-DSA strategies, both centralized and distributed approaches allocate 2 chunks per cell from minutes 0 to 25, and 5 chunks for the central cell and 1 chunk for the rest of the cells from minutes 35 to 60. Notice that these values suppose a reduction of used bandwidth compared with fixed frequency planning schemes, especially with respect to FRF1 that allocates 6 chunks in all the cells.

B. Case Study 2. System adaptability

This case study pretends to demonstrate the adaptability of the proposed distributed self-organized spectrum assignment scheme through an illustrative example. Suppose that a traffic hot-spot emerges on the macrocell scenario described at the beginning of this section (Fig. 4). Traffic hot-spot has 30 uniformly distributed static users that connect to nearest macrocell

and has a radius of 100 meters. In addition to that, there are 15 uniformly distributed users already operating in each cell area. Then, the performance in most affected macrocells (average dissatisfaction probability and spectral efficiency) is negatively impacted by the hot-spot, whose users experience low signal strength from macrocells, and hence are more sensitive to intercell interference. To cope with this loss of performance, a microcell is activated in the hot-spot area at a certain point of the time T. Then, users in the hot-spot perform handover to the microcell after microcell's activation.

Fig. 5 shows how the microcell and macrocells rearrange their spectrum, as well as the performance evolution of the distributed spectrum management for the microcell and macrocells 1, 5, and 6. Notice in Fig. 5 that, before microcell activation, macrocells use between 3 and 5 chunks and that average dissatisfaction probability and spectral efficiency are poor. After microcell activation, spectrum is dynamically managed in micro- and macrocells, activating 4 chunks in the microcell and only 2 chunks per macrocell, which are enough to cope with users' requirements. It can be seen that dissatisfaction improves very significantly by falling below 5% and spectral efficiency increases accordingly after microcell activation.

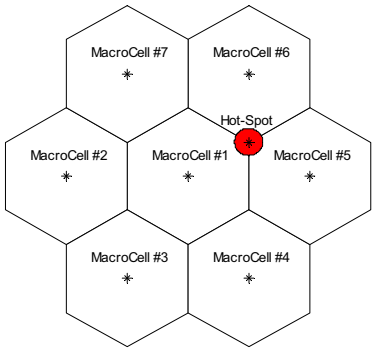

Fig. 4. Scenario layout with hot-spot.

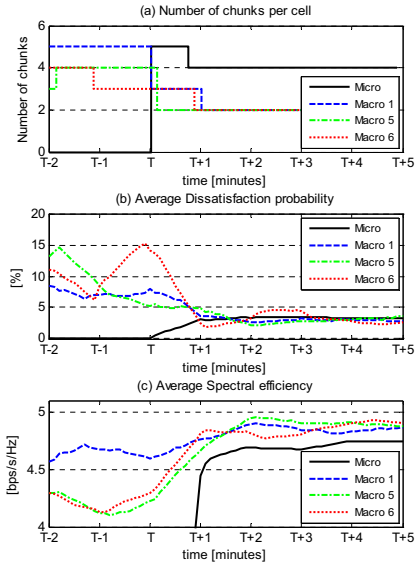

Fig. 5. Number of assigned chunks, average dissatisfaction probability and spectral efficiency evolution for the scenario with hot-spot.

VI. Conclusion

In this paper a distributed framework for spectrum assignment in the context of cellular primary networks has been presented. System model has been designed following self-organization paradigms: distributed and autonomous nature, implicit coordination, reduced state system modeling and adaptive procedures. A dynamic spectrum assignment algorithm based of Reinforcement Learning (RL-DSA) has been included in each autonomous cell. Compared with other fixed spectrum planning and dynamic centralized strategies, the proposed algorithm demonstrates the best tradeoff between spectral efficiency and QoS fulfillment thanks to an adequate adaptability to temporal and spatial variations of the spectrum demands. Additionally, the proposed distributed framework thanks to its autonomous, self-organized nature, appropriately manages the spectrum configuration of the system when new infrastructure is added reducing thus operational expenditures.

References

[1] V. Chandrasekhar, J. Andrews, and A. Gatherer, "Femtocell networks: a survey," *Communications Magazine, IEEE*, vol. 46, no. 9, pp. 59–67, September 2008.

[2] M. M. Buddhikot, "Understanding dynamic spectrum access: Models, taxonomy and challenges," in *IEEE New Frontiers in Dynamic Spectrum Access Networks (DySPAN)*, 2007, pp. 649–663.

[3] C. Prehofer and C. Bettstetter, "Self-organization in communication networks: principles and design paradigms," *IEEE Commun. Mag.*, vol. 43, no. 7, pp. 78–85, July 2005.

[4] E. Bogenfeld and I. e. Gaspard, "Self-x in radio access networks," Whitepaper, Tech. Rep., Dec. 2008. [Online]. Available: https://www.ict-e3.eu/project/dissemination/whitepapers/ whitepapers.html

[5] G. Tesauro, "Reinforcement learning in autonomic computing: A manifesto and case studies," *IEEE Internet Computing*, vol. 11, no. 1, pp. 22–30, 2007.

[6] U. Berthold, F. Fu, M. van der Schaar, and F. Jondral, "Detection of spectral resources in cognitive radios using reinforcement learning," in *3rd IEEE Symposium on New Frontiers in Dynamic Spectrum Access Networks, 2008. DySPAN 2008.*, Oct. 2008, pp. 1–5.

[7] R. Farha, N. Abji, O. Sheikh, and A. Leon-Garcia, "Market-based resource management for cognitive radios using machine learning," in *IEEE Global Telecommunications Conference, 2007. GLOBECOM '07.*, Nov. 2007, pp. 4630–4635.

[8] F. Bernardo, R. Agustí, J. Perez-Romero, and O. Sallent, "A novel framework for dynamic spectrum assignment in multicell OFDMA networks based on reinforcement learning," in *IEEE Wireless Communications and Networking Conference (WCNC)*, April 2009.

[9] A. Stolyar and H. Viswanathan, "Self-organizing dynamic fractional frequency reuse in ofdma systems," in *IEEE Conference on Computer Communications. INFOCOM 2008.*, April 2008, pp. 691–699.

[10] S. Xinghua, H. Zhiqiang, N. Kai, and W. Weiling, "A hierarchical resource allocation for ofdma distributed wireless communication systems," in *IEEE Global Telecommunications Conference, 2007. GLOBECOM '07.*, Nov. 2007, pp. 5195–5199.

[11] R. J. Williams, "Simple statistical gradient-following algorithms for connectionist reinforcement learning," *Machine Learning*, vol. 8, no. 3, pp. 229–256, May 1992.

[12] M. Thathachar and P. Sastry, "Varieties of learning automata: an overview," *IEEE Trans. on Systems, Man, and Cybernetics, Part B: Cybernetics,*, vol. 32, no. 6, pp. 711– 722, December 2002.

[13] C. Wengerter, J. Ohlhorst, and A. v. Elbwart, "Fairness and throughput analysis for generalized proportional fair frequency scheduling in OFDMA," in *IEEE 61st Vehicular Technology Conference 2005-Spring*, vol. 3, 2005, pp. 1903–1907.

[14] 3GPP, "Physical layer aspects for evolved universal terrestrial radio access (UTRA)," 3GPP, Tech. Rep. TR 25.814 v7.1.0, September 2006, release 7.