# Implications of Decentralized Q-learning Resource Allocation in Wireless Networks

Francesc Wilhelmi, Boris Bellalta
Wireless Networking (WN-UPF)
Universitat Pompeu Fabra
Barcelona, Spain

Cristina Cano
WINE Group
Universitat Oberta de Catalunya
Castelldefels, Spain

Anders Jonsson
Art. Int. and Mach. Learn. (AIML-UPF)
Universitat Pompeu Fabra
Barcelona, Spain

***Abstract*—Reinforcement Learning is gaining attention by the wireless networking community due to its potential to learn good-performing configurations only from the observed results. In this work we propose a stateless variation of Q-learning, which we apply to exploit spatial reuse in a wireless network. In particular, we allow networks to modify both their transmission power and the channel used solely based on the experienced throughput. We concentrate in a completely decentralized scenario in which no information about neighbouring nodes is available to the learners. Our results show that although the algorithm is able to find the best-performing actions to enhance aggregate throughput, there is high variability in the throughput experienced by the individual networks. We identify the cause of this variability as the adversarial setting of our setup, in which the most played actions provide intermittent good/poor performance depending on the neighbouring decisions. We also evaluate the effect of the intrinsic learning parameters of the algorithm on this variability.**

## I. Introduction

Reinforcement Learning (RL) has recently spread use in the wireless communications field to solve many kinds of problems such as Access Point (AP) association [1], channel selection [2] or transmit power adjustment [3], as it allows learning good-performing configurations only from the observed results. Among these, Q-learning has been applied to dynamic channel assignment in mobile networks in [4] and to automatic channel selection in Femto Cell networks in [5]. However, to the best of our knowledge, the case of a fully decentralized scenario where nodes do not have knowledge from each other, has not yet been considered.

In this work we propose a stateless variation of Q-learning in which nodes select the transmission power and channel to use solely based on their resulting throughput. We concentrate on a fully decentralized scenario where no information about the actions and resulting performance of the other nodes is available to the learners. Note that inferring the throughput of neighbouring nodes allocated to different channels is costly as periodic sensing in the other channels would then be needed. We aim to characterize the performance of Q-learning in such scenarios, obtaining insight on the most played actions (i.e., channel and transmit power selected) and the resulting performance. We observe that when no information about the neighbours is available to the learners, these will tend to apply selfish strategies that result in alternating good/poor performance depending on the actions of the others. In such scenarios, we show that the use of Q-learning allows each network to find the best-performing actions, though without reaching a steady solution. Note that achieving a steady solution in a decentralized environment relies in finding a Nash Equilibrium, a concept used in Game Theory to define a set of individual strategies that maximize the profits of each player in a non-cooperative game, regardless of the others' strategy. Formally, a set of best player actions $a^* = (a_1^*, ..., a_n^*) \in A$ leads to a Nash Equilibrium if $a_i^* \in B_i(a_{-i}^*), \forall i \in N$, where $B_i(a_{-i})$ is the best response to the others actions $(a_{-i})$. Thus, the consequences of not reaching a Nash Equilibrium can have an impact on performance variability.

In addition, we look at the resulting performance in terms of throughput when varying several parameters intrinsic to the learning algorithm, which helps in understanding the interactions between the degree of exploration and learning rate, and the variability of the resulting performance.

The remaining of this document is structured as follows: Section II introduces the simulation scenario and considerations. Then, Section III presents our Stateless variation of Q-learning and its practical implementation for the resource allocation problem in Wireless Networks (WNs). Simulation results are later discussed in Section IV. Finally, some final remarks are provided in Section V.

## II. System model

For the remainder of this work, we consider a scenario in which several WNs are placed in a 3D-map (with parameters described later in Section IV-A), each one formed by an Access Point (AP) transmitting to a single Station (STA) in downlink manner.

### A. Channel modelling

Path-loss and shadowing effects are modelled using the log-distance model for indoor communications. The path-loss between WN $i$ and $j$ is given by

$$\begin{aligned} \mathrm{PL}_{i,j} &= \mathrm{P}_{\mathrm{tx},i} - \mathrm{P}_{\mathrm{rx},j} = \\ &= \mathrm{PL}_0 + 10\alpha_{\mathrm{PL}}\log_{10}(d_{i,j}) + \mathrm{G_s} + \frac{d_{i,j}}{d_{\mathrm{obs}}}\mathrm{G_o}, \end{aligned}$$

where $\mathrm{P}_{\mathrm{tx},i}$ is the transmitted power in dBm by WN $i$, $\mathrm{P}_{\mathrm{rx},j}$ is the power in dBm received in WN $j$, $\mathrm{PL}_0$ is the path-loss at one meter in dB, $\alpha_{\mathrm{PL}}$ is the path-loss exponent, $d_{i,j}$ is the distance between the transmitter and the receiver in meters, $\mathrm{G_s}$

is the shadowing loss in dB, and $\mathrm{G_o}$ is the obstacles loss in dB. Note that we include the factor $d_{\mathrm{obs}}$, which is the distance between two obstacles in meters.

### B. Throughput calculation

By using the power received and the interference, we calculate the maximum theoretical throughput of each WN $i$ at time $t \in \{1, 2...\}$ by using the Shannon Capacity.

$$\Gamma_{i,t} = B \log_2(1 + \mathrm{SINR}_{i,t}),$$

where $B$ is the channel bandwidth and the experienced Signal to Interference plus Noise Ratio (SINR) is given by:

$$\mathrm{SINR}_{i,t} = \frac{\mathrm{P}_{i,t}}{\mathrm{I}_{i,t} + \mathrm{N}},$$

where $\mathrm{P}_{i,t}$ and $\mathrm{I}_{i,t}$ are the received power and the sum of the interference at WN $i$ at time $t$, respectively, and N is the floor noise power. For each STA in a WN, the interference is considered to be the total power received from all the APs of the other coexisting WNs as if they were continuously transmitting. Adjacent channel interference is also considered in $\mathrm{I}_{i,t}, i \in \{1, .., W\}$, where $W$ is the number of neighbouring WNs. We consider that the transmitted power leaked to adjacent channels is 20 dBm lower for each channel separation.

## III. Decentralized Stateless Q-learning for enhancing Spatial Reuse in WNs

Q-learning [6, 7] is an RL technique that enables an agent to learn the optimal policy to follow in a given environment. A set of possible states describing the environment and actions are defined in this model. In particular, an agent maintains an estimate of the expected long-term discounted reward for each state-action pair, and selects actions with the aim of maximizing it. The expected cumulative reward $\mathrm{V}^{\pi}(s)$ is given by:

$$\mathrm{V}^{\pi}(s) = \lim_{N \to \infty} \mathbb{E}\left(\sum_{t=1}^{N} r_t^{\pi}(s)\right),$$

where $r_t^{\pi}(s)$ is the reward obtained at iteration $t$ after starting from state $s$ and by following policy $\pi$. Since the reward may easily get unbounded, a discount factor parameter ($\gamma < 1$) is used. The optimal policy $\pi^*$ that maximizes the total expected reward is given by the Bellman's Optimality Equation [6]:

$$Q^*(s,a) = \mathbb{E}\Big\{ r_{t+1} + \gamma \mathrm{max}_{a'} Q^*(s_{t+1}, a') | s_t = s, a_t = a \Big\}.$$

Henceforth, Q-learning receives information about the current state-action tuple $(s_t, a_t)$, the generated reward $r_t$ and the next state $s_{t+1}$, in order to update the Q-table:

$$\hat{Q}(s_t, a_t) \leftarrow (1 - \alpha_t)\hat{Q}(s_t, a_t) + \alpha_t \Big( r_t + \gamma \big( \max_{a'} \hat{Q}(s_{t+1}, a') \big) \Big),$$

where $\alpha_t$ is the learning rate at time $t$, and $\max_{a'} \hat{Q}(s_{t+1}, a')$ is the best estimated value for the next state $s_{t+1}$. The optimal solution is theoretically achieved with probability 1 if $\sum_{t=0}^{\infty} \alpha_t = \infty$, and $\sum_{t=0}^{\infty} \alpha_t^2 < \infty$, which satisfies that $\lim_{t \to \infty} \hat{Q}(s,a) = Q^*(s,a)$. Since we focus on a completely decentralized scenario where no information about the other nodes is available, the system can then be fully described by the set of actions and rewards.[1] Thus, we propose using a stateless variation of the original Q-learning algorithm. To implement decentralized learning to the resource allocation problem, we consider each WN to be an agent running Stateless Q-learning through an $\varepsilon$-greedy action-selection strategy, so that actions $a \in \mathcal{A}$ correspond to all the possible configurations that can be chosen with respect to the channel and transmit power. During the learning process we assume that WNs select actions sequentially, so that at each learning iteration, every agent takes an action in an ordered way. The order at which WNs choose an action at each iteration is randomly selected at the beginning of it. The reward after choosing an action is set as:

$$r_{i,t} = \frac{\Gamma_{i,t}}{\Gamma_i^*},$$

where $\Gamma_{i,t}$ is the experienced throughput at time $t$ by WN $i \in \{1, ..., n\}$, being $n$ the number of WNs in the scenario, and $\Gamma_i^* = B \log_2(1 + \mathrm{SNR}_i)$ is WN $i$ maximum achievable throughput (i.e., when it uses the maximum transmission power and there is no interference). Each WN applies the Stateless Q-learning as follows:

- Initially, it sets the estimates of its actions $k \in \{1, ..., K\}$ to 0: $\hat{Q}(a_k) = 0$.
- At each iteration, it applies an action by following the $\varepsilon$-greedy strategy, i.e., it selects the best-rewarding action with probability $1 - \varepsilon_t$, and a random one (uniformly distributed) the rest of the times.
- After choosing action $a_k$, it observes the generated reward (the relative experienced throughput), and updates the estimated value $\hat{Q}(a_k)$.
- Finally, $\varepsilon_t$ is updated to follow a decreasing sequence: $\varepsilon_t = \frac{\varepsilon_0}{\sqrt{t}}$.

Note, as well, that the optimal policy cannot be derived for the presented scenario, but it can be approximated to enhance spatial reuse. This is due to the nature of the presented environment, as well as WNs decisions affect the others performance. Formally, the implementation details of Stateless Q-learning are described in Algorithm 1. The presented learning approach is intended to operate at the PHY level, allowing the operation of the current MAC-layer communication standards (e.g., in IEEE 802.11 WLANs, the channel access is governed by the CSMA/CA operation, so that Stateless Q-learning may contribute to improve spatial reuse at the PHY level).

## IV. Performance Evaluation

In this section we introduce the simulation parameters and describe the experiments.[2] Then, we show the main results.

[1] We note that local information such as the observed instantaneous channel quality could be incorporated in the state definition. However, such a description of the system entails increased complexity.

[2] The code used for simulations can be found at https://github.com/wn-upf/Decentralized_Qlearning_Resource_Allocation_in_WNs.git (Commit: eb4042a1830c8ea30b7eae3d72a51afe765a8d86).

**Algorithm 1:** Stateless Q-learning

Function Stateless Q-learning (SINR, $\mathcal{A}$);
**Input :** SINR: Signal-to-Interference-plus-Noise Ratio sensed at the STA
$\mathcal{A}$: set of possible actions in {1, ..., K}
**Output:** $\overline{\Gamma}$: Mean throughput experienced in the WN
initialize: $t = 0$, $\hat{Q}(a_k) = 0, \forall a_k \in \mathcal{A}$
**while** *active* **do**
  Select $a_k$ $\begin{cases} \underset{k=1,\dots,K}{\text{argmax}}\ \hat{Q}(a_k), & \text{with prob } 1-\varepsilon \\ i \sim \mathcal{U}(1,K), & \text{otherwise} \end{cases}$
  Observe reward $r_{a_k} = \frac{\Gamma_{a_k,t}}{\Gamma^*}$
  $\hat{Q}(a_k) \leftarrow \hat{Q}(a_k) + \alpha \cdot \left(r_{a_k} + \gamma \cdot \max \hat{Q} - \hat{Q}(a_k)\right)$
  $\varepsilon_t \leftarrow \varepsilon_0/\sqrt{t}$
  $t \leftarrow t + 1$
**end**

### A. Simulation Parameters

According to [8], a typical high-density scenario for residential buildings contains $0.0033\text{APs/m}^3$. We then consider a map scenario with dimensions $10 \times 5 \times 10$ m containing 4 WNs that form a grid topology in which STAs are placed at the maximum possible distance from the other networks. This toy scenario allows us to study the performance of Stateless Q-learning in a controlled environment , which is useful to check the applicability of RL in WNs by only using local information [3]. We consider that the number of channels is equal to half the number of coexisting WNs, so that we can study a challenging situation regarding the spatial reuse. Table I details the parameters used.

| Parameter | Value |
|---|---|
| Map size (m) | $10 \times 5 \times 10$ |
| Number of coexistent WNs | 4 |
| APs/STAs per WN | 1 / 1 |
| Distance AP-STA (m) | $\sqrt{2}$ |
| Number of Channels | 2 |
| Channel Bandwidth (MHz) | 20 |
| Initial channel selection model | Uniformly distributed |
| Transmit power values (dBm) | {5, 10, 15, 20} |
| $PL_0$ (dB) | 5 |
| $\alpha_{PL}$ | 4.4 |
| $G_s$ (dB) | Normally distributed with mean 9.5 |
| $G_o$ (dB) | Uniformly distributed with mean 30 |
| $d_{obs}$ (meters between two obstacles) | 5 |
| Noise level (dBm) | -100 |
| Traffic model | Full buffer (downlink) |

TABLE I: Simulation parameters

### B. Optimal solution

We first identify the optimal solutions that maximize: $i$) the aggregate throughput, and $ii$) the proportional fairness, which is computed as the logarithmic sum of the throughput experienced by each WN, i.e., $\text{PF} = \max_{k \in \mathcal{A}} \sum_i \log(\Gamma_{i,k})$. The optimal solutions are listed in Table II. Note that, since the considered scenario is symmetric, there are two equivalent solutions. Note, as well, that in order to maximize the aggregate network throughput two of the WNs sacrifice themselves by choosing a lower transmit power. This result is then not likely to occur in an adversarial selfish setting.

[3]The analysis of the presented learning mechanisms in more congested scenarios is left as future work.

| WN id | Action that maximizes the Aggregate Throughput | Action that maximizes the Proportional Fairness |
|---|---|---|
| 1 | 1 (2) | 7 (8) |
| 2 | 1 (2) | 8 (7) |
| 3 | 7 (8) | 7 (8) |
| 4 | 8 (7) | 8 (7) |

TABLE II: Optimal configurations (action indexes) to achieve the maximum network throughput and prop. fairness, resulting in 1124 Mbps and 891 Mbps, respectively. In parenthesis the analogous solution is shown. Actions indexes range from 1 to 8 are mapped to {channel number, transmit power (dBm)}: {1,5}, {2,5}, {1,10}, {2,10}, {1,15}, {2,15},{1,20} and {2,20}, respectively.

### C. Input Parameters Analysis

We first analyse the effects of modifying $\alpha$ (the learning rate), $\gamma$ (the discount factor) and $\varepsilon_0$ (the initial exploration coefficient of the $\varepsilon$-greedy update rule) with respect to the achieved network throughput. We run simulations of 10000 iterations and capture the results of the last 5000 iterations to ensure that the initial transitory phase has ended. Each simulation is repeated 100 times for averaging purposes.

Figure 1 shows the average aggregate throughput achieved for each of the proposed combinations. It can be observed that the best results with respect to the aggregate throughput, regarding both average and variance, are achieved when $\alpha = 1$, $\gamma = 0.95$ and $\varepsilon_0 = 1$. This means that for achieving the best results (i.e., high average aggregate throughput and low variance), the immediate reward of a given action must be considered rather than any previous information ($\alpha = 1$). We see that the difference between the pay-off offered by the best action and the current one must also be high ($\gamma = 0.95$). In addition, exploration must be highly boosted at the beginning ($\varepsilon_0 = 1$). For this setting, the resulting throughput (902.739 Mbps) represents 80.29% of the one provided by the optimal configuration that maximizes the aggregate throughput (shown in Table II). Regarding proportional fairness, the algorithm's resulting throughput is only 1.32% higher than the optimal.

We also evaluate the relationship between different values of $\alpha$ and $\gamma$ in the average aggregate throughput and standard deviation (shown in Figure 2). We observe a remarkably higher aggregate throughput when $\alpha > \gamma$. We also see that the variability between different simulation runs is much lower when the average throughput is higher. Additionally, we note a peak in the standard deviation when $\gamma \approx \alpha$ and $\gamma > \alpha$.

To further understand the effects of modifying each of the aforementioned parameters, we show for different $\varepsilon_0$, $\alpha$ and $\gamma$: $i$) the individual throughput experienced by each WN during the total 10000 iterations of a single simulation run (Figure 3),

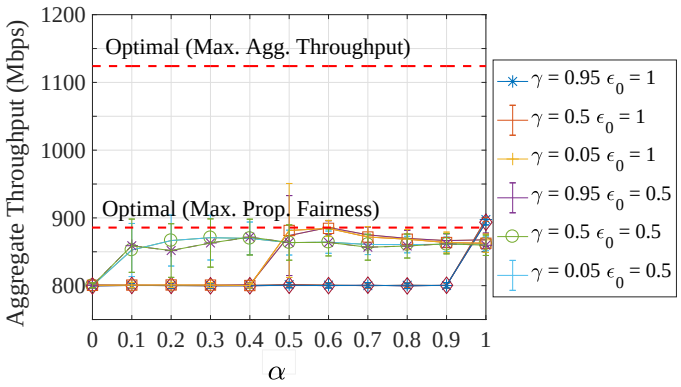
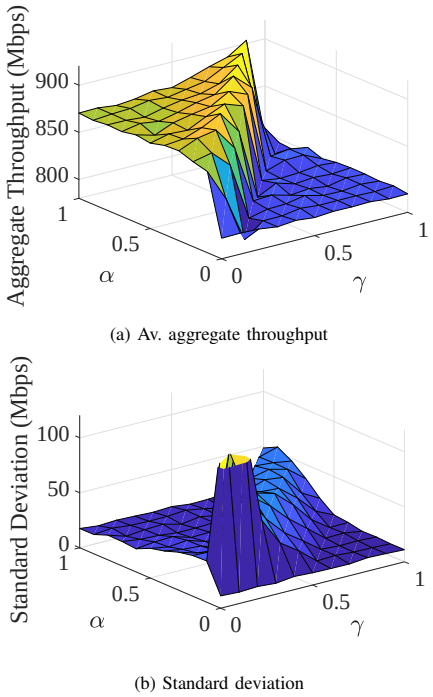


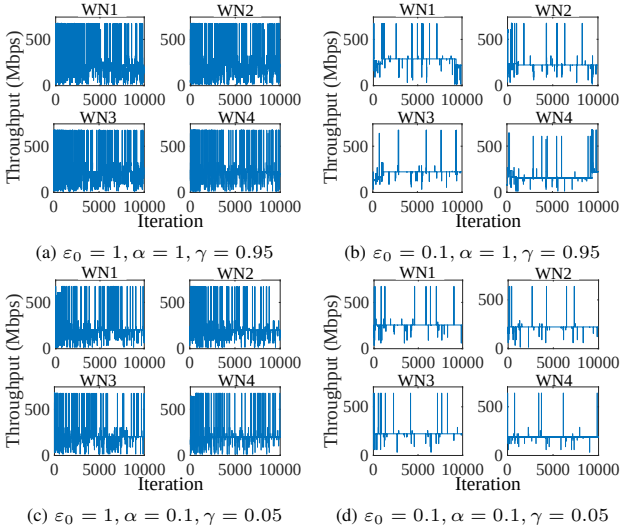

Fig. 1: Effect of $\alpha, \gamma$ and $\epsilon_0$ in the average aggregate throughput (100 simulation runs per sample).




(a) Av. aggregate throughput

(b) Standard deviation

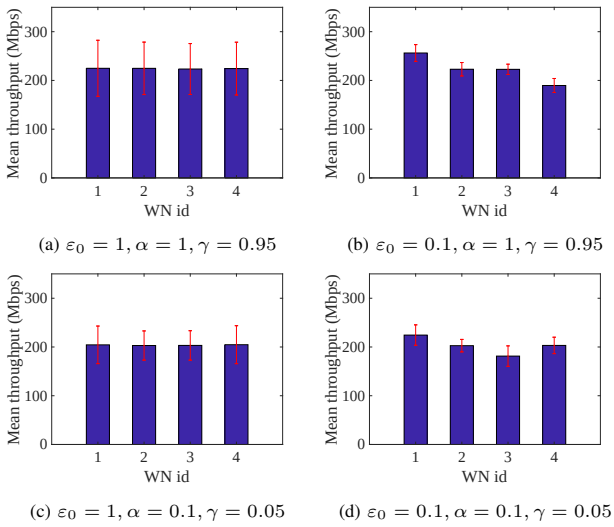
Fig. 2: Evaluation of $\alpha$ and $\gamma$.




(a) $\varepsilon_0 = 1, \alpha = 1, \gamma = 0.95$ (b) $\varepsilon_0 = 0.1, \alpha = 1, \gamma = 0.95$

(c) $\varepsilon_0 = 1, \alpha = 0.1, \gamma = 0.05$ (d) $\varepsilon_0 = 0.1, \alpha = 0.1, \gamma = 0.05$

Fig. 3: Individual throughput experienced by each WN during a single simulation run for different $\varepsilon_0$, $\alpha$ and $\gamma$.



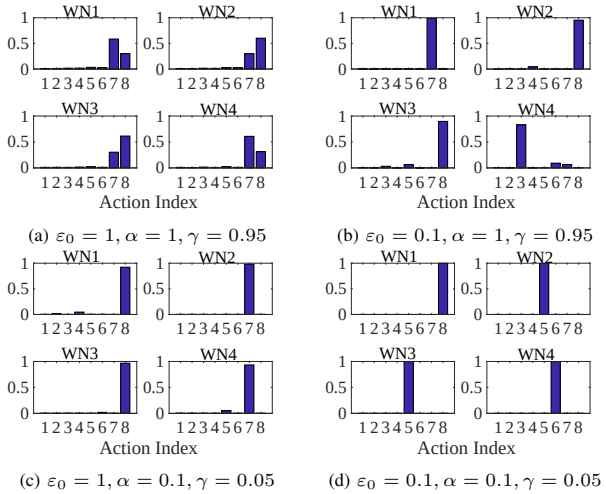

(a) $\varepsilon_0 = 1, \alpha = 1, \gamma = 0.95$ (b) $\varepsilon_0 = 0.1, \alpha = 1, \gamma = 0.95$

(c) $\varepsilon_0 = 1, \alpha = 0.1, \gamma = 0.05$ (d) $\varepsilon_0 = 0.1, \alpha = 0.1, \gamma = 0.05$

Fig. 4: Average throughput experienced by each WN during the last 5000 iterations of a total of 10000 iterations (in a single simulation run) and for different $\varepsilon_0$, $\alpha$ and $\gamma$.

$ii$) the average throughput experienced by each WN for the last 5000 iterations, also for a single simulation run (Figure 4), and $iii$) the probability of choosing each action at each WN (Figure 5). We observe the following aspects:

- In Figure 3 a high variability of the throughput experienced by each WN can be observed, specially if $\epsilon_0$ is high (as in Figures 3(a), 3(c)). A high degree of exploration allows WNs to discover changes in the resulting performance of their actions due to the activity of the other nodes, which at the same time generates more variability (WN adapt to changes in the environment).
- Despite the variability generated, we obtain fairer results for high $\epsilon_0$ (Figure 4). Henceforth, there is a relationship between the variability generated and the average throughput fairness.
- Finally, in Figures 5(a) and 5(c) we observe that for the former, there are two favourite actions that are being played the most, but for the latter there is only one preferred action. The lower the learning rate ($\alpha$), and consequently the discount factor ($\gamma$), the higher the probability of choosing a unique action, which results to be the one that provided the best performance in the past. The opposite occurs for higher $\alpha$ and $\gamma$ values, since giving more importance to the immediate reward allows for a reaction only to the recently-played actions of the neighbouring nodes: the algorithm is short-sighted.




(a) $\varepsilon_0 = 1, \alpha = 1, \gamma = 0.95$ (b) $\varepsilon_0 = 0.1, \alpha = 1, \gamma = 0.95$

(c) $\varepsilon_0 = 1, \alpha = 0.1, \gamma = 0.05$ (d) $\varepsilon_0 = 0.1, \alpha = 0.1, \gamma = 0.05$

Fig. 5: Probability of choosing the different actions at each WN for a single (10000 iterations) simulation run and different $\varepsilon_0$, $\alpha$ and $\gamma$ values

## V. Conclusions

Decentralized Q-learning can be used to improve spatial reuse in dense wireless networks, enhancing performance as a result of exploiting the most rewarding actions. We have shown in this article, by means of a toy scenario, that Stateless Q-learning in particular allows finding good-performing configurations that achieve close-to-optimal (in terms of throughput maximization and proportional fairness) solutions.

However, the competitiveness of the presented fully-decentralized environment involves the non-existence of a Nash Equilibrium. Thus, we have also identified high variability in the experienced individual throughput due to the constant changes of the played actions, motivated by the fact that the reward generated by each action changes according to the opponents' ones. We have evaluated the impact of the parameters intrinsic to the learning algorithm on this variability showing that it can be reduced by decreasing the exploration degree and learning rate. The individual reduction on the throughput variability occurs at the expense of losing aggregate performance.

This variability can potentially result in negative effects on the overall WN's performance. The effects of such a fluctuation in higher layers of the protocol stack can have severe consequences depending on the time scale at which they occur. For example, noticing high throughput fluctuations may trigger congestion recovery procedures in TCP (Transmission Control Protocol), which would harm the experienced performance.

We left for future work to further extend the decentralized approach in order to find collaborative algorithms that allow the neighbouring WNs to reach an equilibrium that grants acceptable individual performance. Acquiring any kind of knowledge about the neighbouring WNs is assumed to solve the variability issues arisen from decentralization. This information may be directly exchanged or inferred from observations. Furthermore, other learning approaches are intended to be analysed in the future for performance comparison in the resource allocation problem.

## Acknowledgment

This work has been partially supported by the Spanish Ministry of Economy and Competitiveness under the Maria de Maeztu Units of Excellence Programme (MDM-2015-0502), and by the European Regional Development Fund under grant TEC2015-71303-R (MINECO/FEDER).

## References

[1] Chen, L. (2010, May). A distributed access point selection algorithm based on no-regret learning for wireless access networks. In Vehicular Technology Conference (VTC 2010-Spring), 2010 IEEE 71st (pp. 1-5). IEEE.

[2] Maghsudi, S., & Staczak, S. (2015). Channel selection for network-assisted D2D communication via no-regret bandit learning with calibrated forecasting. IEEE Transactions on Wireless Communications, 14(3), 1309-1322.

[3] Maghsudi, S., & Staczak, S. (2015). Joint channel selection and power control in infrastructureless wireless networks: A multiplayer multi-armed bandit framework. IEEE Transactions on Vehicular Technology, 64(10), 4565-4578.

[4] Nie, J., & Haykin, S. (1999). A Q-learning-based dynamic channel assignment technique for mobile communication systems. IEEE Transactions on Vehicular Technology, 48(5), 1676-1687.

[5] Bennis, M., & Niyato, D. (2010, December). A Q-learning based approach to interference avoidance in self-organized femtocell networks. In GLOBECOM Workshops (GC Wkshps), 2010 IEEE (pp. 706-710). IEEE.

[6] Sutton, R. S., & Barto, A. G. (1998). Reinforcement learning: An introduction (Vol. 1, No. 1). Cambridge: MIT press.

[7] Watkins, C. J., & Dayan, P. (1992). Q-learning. Machine learning, 8 (3-4), 279-292.

[8] Bellalta, B. "IEEE 802.11 ax: High-efficiency WLANs." IEEE Wireless Communications 23.1 (2016): 38-46.