Distributed Cooperative Q-learning for Power Allocation in Cognitive Femtocell Networks

Hussein Saad*, Amr Mohamed† and Tamer ElBatt*
*Wireless Intelligence Network Center (WINC),
Nile University, Cairo, Egypt.
hussein.saad@nileu.edu.eg
telbatt@nileuniversity.edu.eg
†Computer Science and Engineering Department
Qatar University, P.O. Box 2713, Doha, Qatar.
amrmg@qu.edu.qa

***Abstract*—In this paper, we propose a distributed reinforcement learning (RL) technique called distributed power control using Q-learning (DPC-Q) to manage the interference caused by the femtocells on macro-users in the downlink. The DPC-Q leverages Q-Learning to identify the sub-optimal pattern of power allocation, which strives to maximize femtocell capacity, while guaranteeing macrocell capacity level in an underlay cognitive setting. We propose two different approaches for the DPC-Q algorithm: namely, independent, and cooperative. In the former, femtocells learn independently from each other, while in the latter, femtocells share some information during learning in order to enhance their performance. Simulation results show that the independent approach is capable of mitigating the interference generated by the femtocells on macro-users. Moreover, the results show that cooperation enhances the performance of the femtocells in terms fairness and aggregate femtocell capacity.[1]**

I. Introduction

Femtocells are considered to be a highly promising solution for the enhancement of the indoor coverage problem. However, femtocells are deployed unpredictably in the macrocell area. Thus, their interference on macro-users and other femtocells is considered to be a daunting problem [1], [2].

Since femtocells are installed by the end user, their number and positions are random and unknown to the network operator. This makes the centralized approach for solving the interference problem very hard due to the huge overhead needed which in turn calls for a distributed interference management strategy. In the distributed scheme, each femtocell needs to ***learn*** how to interact with the dynamic environment created by the coexistence of the femto and macro cells in order to adjust its parameters (carrier frequency and transmission power) to satisfy the QoS of its own users while guaranteeing certain QoS for the macrocell users.

Based on these observations, in this paper we focus on closed access femtocells [3] working in the same bandwidth with macrocells (cognitive femtocells). We will use a distributed machine learning technique called reinforcement learning (RL) [4] to handle the interference problem generated by the femtocells on the macrocells' users. One of the most popular RL techniques is Q-learning [5]. The reason we chose Q-learning is because it finds optimal decision policies without any prior model of the environment (in our settings, a prior model can not be achieved due to the unplanned placement of the femtocells and the dynamics of the wireless environment). Moreover, Q-learning allows the agents (i.e the femtocells) to take actions while they are learning (i.e no need for a centralized approach). These features make Q-learning very suitable to be applied to the distributed femtocell setting in the form of the so called multi-agent Q-learning (**MAQL**) [6]. In this paper, MAQL is applied in two different paradigms: independent learning (**IL**) and cooperative learning (**CL**). The former assumes that agents are unaware of the other agents' actions while the latter allows the agents to share some knowledge while they are learning to enhance their performance.

In literature, RL has been used to perform power allocation in femtocell networks. In [7], authors addressed the problem of interference control in the context of OFDMA-based femtocells. In [8], authors used IL Q-learning in the context of cognitive femtocells and introduced a new concept called docitive femtocells. However, all the papers discussed above were interested in maintaining the QoS of the primary users and ignored the QoS of the femtocells (e.g: fairness, maximizing the femtocell capacity). Moreover, they all used the IL paradigm and did not take into consideration any cooperation between the agents (femtocells) during the learning process.

Motivated by this, in this paper we apply Q-learning for power control in closed access cognitive femtocells network. The contributions of this paper can be summed up as follows:

- A distributed algorithm based on IL paradigm is used to handle the interference problem. A new reward function is introduced and compared to the reward function used in literature [7]. The comparison is applied in two different scenarios:
  1) Maintaining the QoS (i.e. the capacity) of the macrocell without taking into consideration the QoS of the femtocells.
  2) Enhancing the capacity of the femtocells while maintaining the QoS of the macrocell.
- Cooperation between the femtocells is introduced to enhance the aggregate capacity and fairness amongst all the femtocells, while maintaining the macrocell QoS.

The remaining part of this paper is organized as follows. Section II gives a brief background for the original single agent Q-learning. In section III, the system model is described.

[1]Tamer ElBatt is also affiliated with Faculty of Eng., Cairo University

978-1-4673-1881-5/12/$31.00 ©2012 IEEE

Section IV introduces the proposed distributed Q-learning algorithm and the Q-learning formulation for the cognitive femtocells problem. The simulation scenario and the results are discussed in section V. Finally the conclusion is given in section VI.

II. Background: Single Agent Q-learning (SAQL)

In this section, the idea of Q-learning is presented by introducing the single agent case [5]. The Q-learning model can be defined by the tuple $\{S, A, P_{s,s'}, R(s,a)\}$ where $S = \{s_1, s_2, \cdots, s_m\}$ is the set of possible states the agent can occupy, $A = \{a_1, a_2, \cdots, a_l\}$ is the set of possible actions the agent can perform, $P_{s,s'}$ is the probabilistic transition function that defines the probability that the agent transits from state s to state $s^{'}$, given a certain action a is performed, and $R(s,a)$ is the reward function that determines the reward fed back to the agent by the environment when performing action a in state s. The interaction between the agent and the environment at time t can be described as follows:

- The agent senses the environment and observes its current state $s_t \in S$.
- Based on s_t, the agent selects action $a_t \in A$.
- Based on a_t and $P_{s,s'}$, the environment makes a transition to a new state $s_{t+1} \in S$ and as a result achieves a reward $r_t = R(s_t, a_t)$ due to this transition.
- The reward is fed back to the agent and the process is repeated.

The end goal of the agent is to find an optimal policy $\pi^*(s)$, which defines the action to be selected for each state $s \in S$ in order to maximize the expected discounted reward over an infinite time:

$$V^{\pi}(s) = \mathbb{E}\{\sum_{t=0}^{\infty} \gamma^t r(s_t, \pi(s)) | s_o = s\} \quad (1)$$

where $V^{\pi}(s)$ is the value function of state s which represents the expected discounted infinite reward when the initial state is s_o and $0 \leq \gamma \leq 1$ is the discount factor that determines how much effect future rewards have on the decisions at each moment. From equation (1), the optimal value function $V^*(s)$ can be written as [7]:

$$V^*(s) = \max_{a \in A} (\mathbb{E}\{r(s,a)\} + \gamma \sum_{s' \in S} P_{s,s'}(a) V^*(s^{'})) \quad (2)$$

Q-learning aims at finding the optimal policy $\pi^*(s)$ that corresponds to $V^*(s)$ without having any prior knowledge about the transition probabilities $P_{s,s'}$. In order to do this, a new value called Q-value is defined for each state-action pair, where the optimal Q-value is defined as:

$$Q^*(s,a) = \mathbb{E}\{r(s,a)\} + \gamma \sum_{s' \in S} P_{s,s'}(a) \max_{b \in A} Q^*(s^{'}, b) \quad (3)$$

Equation (3) states that the optimal value function can be expressed by $V^*(s) = \max_{a \in A} Q^*(s,a)$. Thus, if the optimal Q-value is known for each state-action pair, the optimal policy can be determined by $\pi^*(s) = arg\max_{a \in A} Q^*(s,a)$. The Q-learning algorithm finds $Q^*(s,a)$ in a recursive manner using a simple update rule:

$$Q(s,a) := (1-\alpha) Q(s,a) + \alpha (r(s,a) + \gamma \max_{b \in A} Q(s^{'}, b)) \quad (4)$$

Where α is the learning rate. It was proved in [5], [9] that this update rule converges to the optimal Q-value under certain conditions. One of these conditions is that each state-action pair must be visited infinitely often [5]. To address this notion, a random number ϵ is introduced where at each step of the learning process the action is chosen according to $a = arg\max_{a \in A} Q(s,a)$ with probability $(1-\epsilon)$ or randomly with probability ϵ. Moreover, in the convergence proof, the reward function is assumed to be bounded and deterministic for each state-action pair [9]. However, in the multi-agent case, this condition is violated since the reward for each state will depend on the joint action of all agents, hence the reward function will not be deterministic from the agent point of view.

III. System Model

In this paper, a wireless network consisting of one macro cell with one single transmit and receive antenna denoted by Macro Base Station (MBS) underlaid with N_{femto} femtocells each with one Femto Base Station (FBS) is considered. U_m and U_f macro and femto users are located randomly inside the macro and femto cells respectively. Both MBS and FBS's transmit over the same N_{sub} sub-carriers where orthogonal downlink transmission is assumed in each time slot.

The transmission powers of the MBS and FBS i on subcarrier n are denoted by $P_o^{(n)}$ and $P_i^{(n)}$ respectively. Moreover, the maximum transmission powers for the MBS and FBS i are P_{max}^m and P_{max}^f respectively, where $\sum_{n=1}^{N_{sub}} P_o^{(n)} \leq P_{max}^m$ and $\sum_{n=1}^{N_{sub}} P_i^{(n)} \leq P_{max}^f$.

The system performance is analyzed in terms of the capacity measured in (bits/sec/Hz). The capacity achieved by the MBS at its associated user on subcarrier n is:

$$C_o^{(n)} = \log_2(1 + \frac{h_{oo}^{(n)} P_o^{(n)}}{\sum_{i=1}^{N_{femto}} h_{io}^{(n)} P_i^{(n)} + \sigma^2}) \quad (5)$$

where $h_{oo}^{(n)}$ denotes the channel gain between the MBS and its associated user on subcarrier n; $h_{io}^{(n)}$ denotes the channel gain between FBS i and the macro user on subcarrier n and σ^2 is the noise power. The capacity achieved by FBS i at its associated user on subcarrier n is:

$$C_i^{(n)} = \log_2(1 + \frac{h_{ii}^{(n)} P_i^{(n)}}{\sum_{j=1, j \neq i}^{N_{femto}} h_{ji}^{(n)} P_j^{(n)} + h_{oi}^{(n)} P_o^{(n)} + \sigma^2}) \quad (6)$$

where $h_{ii}^{(n)}$ denotes the channel gain between FBS i and its associated user on subcarrier n; $h_{ji}^{(n)}$ denotes the channel gain between FBS j and the femto user associated with FBS i on subcarrier n.

IV. Distributed Power Control using Q-learning (DPC-Q)

DPC-Q is a distributed MAQL algorithm in which multiple agents (i.e: femtocells) aim at learning a sub-optimal decision policy (i.e: power allocation) by repeatedly interacting with the environment. DPC-Q is applied in two different paradigms:

- **Independent learning (IL)**: In this paradigm, each agent learns independently from other agents (i.e: ignores other agents' actions and considers other agents as part of the environment). Although, this may lead to oscillations and convergence problems, the IL paradigm showed good results in many applications [10]. The only difference here compared to the SAQL case is that the reward function is now dependent on the joint action of all agents $\vec{a}$. Thus, the update rule can be rewritten as:

$$Q_i(s_i,a_i) := (1-\alpha)Q_i(s_i,a_i) + \alpha(r_i(s_i,\vec{a}) + \gamma \max_{b\in A_i} Q_i(s_i^{'},b)) \quad (7)$$

  Since the environment is no longer stationary, the dynamics of learning may be long and complex in terms of required time and memory. A possible solution to mitigate this problem is to exchange knowledge between the agents during the learning process, aiming at speeding up the learning process and enhancing the agents' performance. Motivated by this, we propose the following paradigm in which each agent shares a portion of its Q-table with all other agents [2].
- **Cooperative learning (CL)**: CL is performed as follows: Agent i shares the row of its Q-table that corresponds to its current state with all other cooperating agents j (i.e. femtocells in the same range). Then agent i selects its action according to the following equation:

$$a_i = arg\max_a \Big(\sum_{1\le j\le N} Q_j(s_j,a)\Big) \quad (8)$$

  The main idea behind this strategy depends on what is called: the global Q-value **Q(s,a)**, which represents the Q-value of the whole system (i.e. if the multi-agent scenario is transformed into a single agent one using a centralized controller with global state **s** and global joint action **a**). This global Q-value can be decomposed into a linear combination of local agent-dependent Q-values: $\mathbf{Q(s,a)} = \sum_{1\le j\le N} Q_j(s_j,a_j)$ [6]. Thus, if each agent j maximized its own Q-value, the global Q-value will be maximized. Based on this observation, choosing the action based on equation 8 would maximize the global Q-value. However, the solution is still not global optimum because based on equation 8, all agents will choose the same action. For example, if there are two agents (femtocells) 1 and 2, each agent has one state s and three actions $a1$, $a2$ and $a3$, the reward for each agent is its capacity and the Q-values for both agents are as follows: $Q_1(s,a1)=1$, $Q_1(s,a2)=2$, $Q_1(s,a3)=3$, $Q_2(s,a1)=4$, $Q_2(s,a2)=6$ and $Q_2(s,a3)=4.5$, then in the IL paradigm, agent 1 will choose action $a3$, thus maximizing its capacity, while agent 2 will choose action $a2$, thus maximizing its capacity. However, in the CL paradigm, both agents will choose action $a2$ (the maximum of the summation of the Q-values is $2+6$), thus maximizing the aggregate capacity.

  In terms of overhead, according to our proposed cooperation algorithm each femtocell should only share a row of its Q-table with all its neighbors. This row has a size of $1x|A|$. So if the number of femtocells is N_{femto}, then the total overhead needed is $N_{femto}.(N_{femto}-1)$ messages, each of size $|A|$, per unit time (i.e. the overhead is quadratic in the number of cooperating femtocells). The different paradigms of the DPC-Q algorithm are summarized in algorithm 1.

Algorithm 1 The proposed DPC-Q algorithm

```
Let t = 0, Q_i^0(s_i, a_i) = 0 for all s_i ∈ S and a_i ∈ A
Initialize the starting state s_i^t
loop
send Q_i^t(s_i^t, :) to all other cooperating agents j
receive Q_j^t(s_j^t, :) from all other cooperating agents j
if rand < ε then
   select action randomly
else
   if leaning paradigm == IL then
      choose action: a_i^t = arg max_a Q_i(s_i^t, a)
   else
      choose action: a_i^t = arg max_a (Σ_{1≤j≤N} Q_j^t(s_j^t, a))
   end if
end if
receive reward r_i^t
observe next state s_i^{t+1}
update Q-table as in equation 7
s_i^t = s_i^{t+1}
end loop
```

The agents, states, actions and reward function are defined as follows:

- **Agent:** $FBS_i, \forall 1 \le i \le N_{femto}$
- **State:** At time instant t for femtocell i in subcarrier n, the state is defined as: $s_t^{i,n} = \{I_t^n, P_t^i\}$ where $I_t^n \in \{0,1\}$ indicates the level of interference measured at the macro-user in subcarrier n at time t:

$$I_t^n = \begin{cases} 1, & C_o^{(n)} < \Gamma^o \\ 0, & C_o^{(n)} \ge \Gamma^o \end{cases} \quad (9)$$

  where Γ^o is the target capacity determining the QoS performance of the macrocell. We assume that the macrocell reports the value of C_o^n to all FBS through the backhaul connection.

[2]We assume that the shared portion of the Q-table is put in the control bits of the packets transmitted between the femtocells. The details of the exact protocol lie out of the scope of this paper.

P_t^i determines the total power FBS i is transmitting with at time t:

$$P_t^i = \begin{cases} 0, & \sum_{n=0}^{N_{sub}} p_t^{i,n} < (P_{max}^f - A1) \\ 1, & (P_{max}^f - A2) \leq \sum_{n=0}^{N_{sub}} p_t^{i,n} \leq P_{max}^f \\ 2, & \sum_{n=0}^{N_{sub}} p_t^{i,n} > P_{max}^f \end{cases} \quad (10)$$

where P_{max}^f, $A1$ and $A2$ are set to 15, 5 and 5 dBm respectively in the simulations and $p_t^{i,n}$ is the power femtocell i transmitting with on subcarrier n at time t. *It should be noticed that other values for $A1$ and $A2$ as well as more power levels were tried through the simulations and the performance gain was marginal.*

- **Action:** The set of actions for each agent is the set of possible powers that the FBS can use. In the simulations a range from -20 to 15 dBm with step of 2 dBm is used.
- **Reward:** Two different reward functions were considered in the simulations. The first one is:

$$r_t^{i,n} = \begin{cases} e^{-(C_o^{(n)} - \Gamma^o)^2}, & \sum_{n=0}^{N_{sub}} p_t^{i,n} \leq P_{max}^f \\ -1, & \sum_{n=0}^{N_{sub}} p_t^{i,n} > P_{max}^f \end{cases} \quad (11)$$

The rationale behind this reward function is to maintain the capacity of the macrocell at the target capacity Γ^o while not exceeding the allowed P_{max}^f. The reason for the small difference between the positive (when P_{max}^f is not exceeded) and negative (when P_{max}^f is exceeded) rewards is due to the way the states are defined. Since the state $s_t^{i,n}$ is defined as $\{I_t^n, P_t^i\}$ and P_t^i is defined for certain ranges of powers not for discrete power levels, therefore, large negative numbers can not be assigned as a reward when P_{max}^f is exceeded. For example, if $I_t^n = 1$ and $P_t^i = 6$ dBm, then FBS i is in state $\{1, 0\}$ in subcarrier n. If FBS i took the action $a_t^{i,n} = 8$ dBm, then the next state would be $\{1, 1\}$ and FBS i is rewarded positively according to equation 11. Now consider the case when $I_t^n = 1$ and $P_t^i = 9$ dBm, then FBS i is again in state $\{1, 0\}$ in subcarrier n. If FBS i took the same action $a_t^{i,n} = 8$ dBm, then the next state would $\{1, 2\}$ and FBS i is rewarded -1. So from this example, it can be shown that different rewards could be assigned for the same state-action pair. Thus, the difference between these different rewards must not be large. Based on this observation, in the next section we compare our reward function to the reward function used in [7]:

$$r_t^{i,n} = \begin{cases} K - (C_o^{(n)} - \Gamma^o)^2, & \sum_{n=0}^{N_{sub}} p_t^{i,n} \leq P_{max}^f \\ 0, & \sum_{n=0}^{N_{sub}} p_t^{i,n} > P_{max}^f \end{cases} \quad (12)$$

where K is a constant value. We will show that our reward function improves the convergence compared to above reward function. Note that the authors in [7] defined the state for discrete power levels and this proves our point.

The second reward function used is:

$$r_t^{i,n} = \begin{cases} e^{-(C_o^{(n)} - \Gamma^o)^2} - e^{-C_i^{(n)}}, & \sum_{n=0}^{N_{sub}} p_t^{i,n} \leq P_{max}^f \\ -3, & \sum_{n=0}^{N_{sub}} p_t^{i,n} > P_{max}^f \end{cases} \quad (13)$$

The reward function defined by equation (11) does not take into consideration the femtocell capacity. Thus, we define the above reward function with the rationale of maximizing the femtocell capacity while maintaining the macrocell capacity at Γ^o.

V. Performance Evaluation

A. Simulation Scenario

We consider a wireless network consisting of one macrocell underlaid with N_{femto} femtocells. Each femtocell serves $U_f = 1$ femto-user which is randomly located in the femtocell coverage area. Both the macro and femto cells share the same frequency band composed of $N_{sub} = 6$ subcarriers where orthogonal downlink transmission is assumed. The channel gain between transmitter i and receiver j on subcarrier n is assumed to be path-loss dominated and is given by: $h_{ij}^{(n)} = d_{ij}^{(-k)}$, where d_{ij} is the physical distance between transmitter i and receiver j, and k is the path loss exponent. In the simulation $k = 2$ is used. The distances are calculated according to the following assumptions: 1) The maximum distance between the MBS and its associated user is set to 1000 meters, 2) The maximum distance between the MBS/FBS and a femto/macro-user is set to 800 meters, 3) The maximum distance between a FBS and its associated user is set to 80 meters, 4)The maximum distance between a FBS and another femtocell's user is set to 300 meters.

We used MatLab on a cluster computing facility with 300 cores to simulate such scenario, where in the simulations we set the noise power σ^2 to 10^{-7}, the maximum transmission power of the macrocell P_{max}^m to 43 dBm, the learning rate α to 0.5, the discounted rate γ to 0.9 and the random number ϵ to 0.1 during the first 80% of the Q-iterations [7].

B. Numerical Results

We will refer to the reward functions defined by equations (11), (12) and (13) as $RF1$, $RF2$ and $RF3$ respectively in all the simulations. Figure 1 shows the convergence of the macrocell capacity on a certain subcarrier ($C_o^{(n)}$) using $RF1$ and $RF2$ with $K = 80$, $K = 1000$ and $K = 10000$. It can be observed that $RF1$ shows better convergence behavior than $RF2$ with all values of K (i.e: $RF1$ converges to the target capacity ($\Gamma_o = 6$) more accurately). Moreover, the figure shows that the value of K affects the convergence where $K = 80$ is better than $K = 1000$ and $K = 1000$ is better than $K = 10000$, which proves our point that as the difference between the positive and negative rewards decreases, the convergence is enhanced. Note that in the simulations, the number of Q-iterations was 3000 while in the figure only 300 iterations are shown (i.e: The figure is drawn with step $= 10$) in order to achieve better resolution.

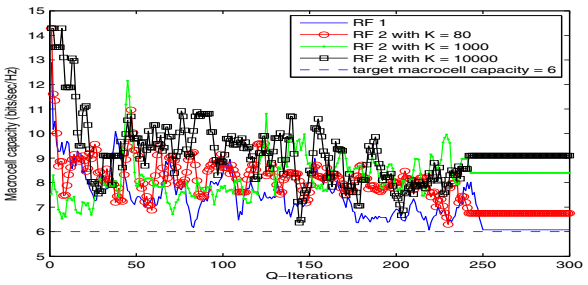

Fig. 1. Convergence of the macrocell capacity using different reward functions with $N_{femto} = 4$ with target capacity $= 6$.

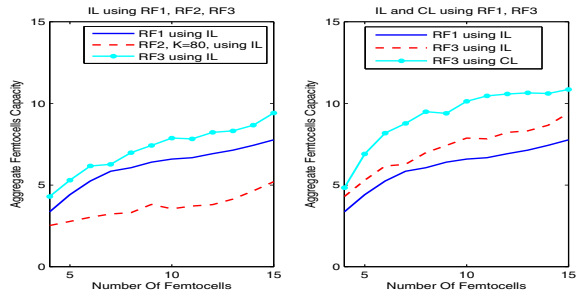

Fig. 2. Aggregate femtocell capacity as a function of the number of femtocells.

The left most figure of figure 2 shows the aggregate femtocell capacity using $RF1$, $RF2$ with $K = 80$ and $RF3$ in the IL paradigm. It can be observed that introducing $C_i^{(n)}$ in $RF3$ increases the aggregate femtocell capacity compared to $RF1$. However, since the IL paradigm is used here, the femtocells act in a selfish way, which may reduce the fairness (in terms of capacity) between the femtocells compared to $RF1$. This is shown in the left most figure of figure 3. Note that the fairness is evaluated using Jain's fairness index [11]: $f(x_1, x_2, \cdots, x_n) = \frac{(\sum_{i=1}^{n} x_i)^2}{n \sum_{i=1}^{n} x_i^2}$ where $0 \leq f(x_1, x_2, \cdots, x_n) \leq 1$ and the equality to 1 occurs when all the femtocells achieve the same capacity.

As for the cooperation effect, the right most figure of figure 2 shows the aggregate femtocell capacity using $RF1$ in the IL paradigm and $RF3$ in both IL and CL paradigms. From the figure, it can be noticed that introducing cooperation increases the total femtocell capacity. Actually, it can be observed that at $N_{femto} = 11$ cooperation increased the capacity by around 2.6 bits/sec/Hz. The right most figure of figure 3 shows that cooperation not only increases the capacity but also enhances the fairness compared ti the IL paradigm.

VI. Conclusion

In this paper, a distributed Q-learning algorithm based on the multi-agent systems theory called DPC-Q is presented to perform power allocation in cognitive femtocells network. The DPC-Q algorithm is applied in two different paradigms: independent and cooperative. In the independent paradigm, two scenarios were considered. The first scenario is to control

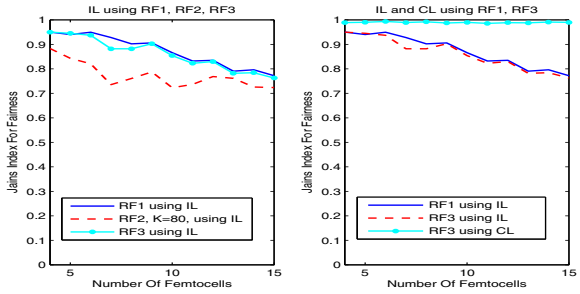

Fig. 3. Jain's fairness index (in terms of capacity) as a function of the number of femtocells.

the interference generated by the femtocells on the macro-user where the results showed that the proposed algorithm is capable of maintaining the capacity of the macro-user at a certain threshold. The second scenario is to enhance the aggregate capacity of femtocells while maintaining the QoS of the macro-user. Through simulations, we showed that the independent learning paradigm can be used to increase the aggregate femtocell capacity. However, due to the selfishness of the femtocells, fairness is reduced compared to the first scenario. Thus, we proposed a cooperative paradigm, in which, femtocells share a portion of their Q-tables with each other. Simulation results showed that cooperation is capable of increasing the aggregate femtocell capacity and enhancing the fairness compared to the independent paradigm, with a relatively small overhead.

Acknowledgment

This work is supported by the Qatar Telecom (Qtel) Grant No.QUEX-Qtel-09/10-10.

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] A. G. S. Saunders, S. Carlaw *et al.*, *Femtocells: Opportunities and Challenges for Business and Technology*. Great Britain: John Wiley and Sons Ltd, 2009.

[3] P. Xia, V. Chandrasekhar, and J. G. Andrews, "Open vs closed access femtocells in the uplink," *CoRR*, vol. abs/1002.2964, 2010.

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

[5] C. J. C. H. Watkins and P. Dayan, "Technical note Q-learning," *Journal of Machine Learning*, vol. 8, pp. 279–292, 1992.

[6] J. R. Kok, "Coordination and learning in cooperative multiagent systems," *Communication*, 2006.

[7] A. Galindo-Serrano and L. Giupponi, "Distributed Q-learning for interference control in OFDMA-based femtocell networks," in *Vehicular Technology Conference (VTC 2010-Spring), 2010 IEEE 71st*, May 2010, pp. 1 –5.

[8] A. Galindo-Serrano, L. Giupponi, and M. Dohler, "Cognition and docition in OFDMA-based femtocell networks," in *proceeding of GLOBECOM 2010, 2010 IEEE Global Telecommunications Conference*, Dec. 2010, pp. 1 –6.

[9] F. S. Melo, "Convergence of Q-learning: A simple proof," Institute Of Systems and Robotics, Tech. Rep.

[10] L. Panait and S. Luke, "Cooperative multi-agent learning: The state of the art," *Autonomous Agents and Multi-Agent Systems*, vol. 11, 2005.

[11] R. Jain, D.-M. Chiu, and W. Hawe, "A quantitative measure of fairness and discrimination for resource allocation in shared computer systems," *CoRR*, 1998.