Case-Based Reinforcement Learning for Cognitive Spectrum Assignment in Cellular Networks with Dynamic Topologies

Nils Morozs, David Grace and Tim Clarke
Department of Electronics, University of York
Heslington, York YO10 5DD, United Kingdom
E-mail: {nm553, david.grace, tim.clarke}@york.ac.uk

***Abstract*—Case-based reinforcement learning is a combination of reinforcement learning (RL) and case-based reasoning which has been successfully applied to a variety of artificial intelligence problems concerned with dynamic environments. This paper demonstrates how case-based RL can be applied to distributed dynamic spectrum assignment in cellular networks with dynamic topologies, and what performance improvements can be expected from using this approach in favour of a standard RL algorithm. Simulation results have shown that augmenting a stateless Q-learning algorithm with case-based reasoning functionality has significantly improved the temporal performance of a 9 base station network with dynamic topology. It has mitigated the performance degradation in terms of the probabilities of call blocking and dropping after transitions between different phases of the network topology, thus substantially increasing the usable range of traffic loads of the network.**

Keywords*—*Case-Based Reasoning, Distributed Reinforcement Learning, Dynamic Spectrum Assignment, Cellular Networks

I. Introduction

Spectrum assignment is a fundamental task of a mobile cellular network, concerned with dividing the available spectrum into a set of channels or resource blocks and assigning them to the incoming calls in a way which provides a good quality of service (QoS) to the users. Modern communication systems, such as cognitive radio and LTE networks, require more intelligent and flexible schemes for spectrum assignment than static spectrum allocation. Such schemes belong to the area of dynamic spectrum assignment.

Reinforcement learning (RL) is a machine learning technique for learning solutions to various decision problems only by trial-and error [1]. In terms of dynamic spectrum assignment, RL is a state-of-the-art technique which has recently been attracting a lot of attention in the wireless communications research community. It has been successfully applied to a range of problems such as LTE pico cells [2], cognitive radio [3] [4] and multi-hop backhaul networks [5].

Topology management is an increasingly popular area of research, especially in green communications, where a trade-off between the QoS provided to the users and energy savings of the network is achieved by dynamically switching various base stations on/off, e.g. [6], [7]. This results in networks with dynamic topologies, which are challenging environments for reinforcement learning based dynamic spectrum assignment algorithms. Furthermore, this paper is concerned with distributed dynamic spectrum assignment, where no information exchange is assumed among individually learning base stations. This makes it more challenging for the base stations to learn good policies in dynamic environments. Nevertheless, the distributed RL approach has advantages over centralised methods in that no communication overhead is required to achieve the learning objective, and the network operation does not rely on a single computing unit.

Case-based reinforcement learning is RL augmented with case-based reasoning functionality [8]. Case-based reasoning is broadly defined as the process of solving new problems by using the solutions to similar problems solved in the past [9]. Combining case-based reasoning and reinforcement learning means that these solutions are learned by using an RL algorithm. The combination of these two techniques has been successfully applied to dynamic inventory control [10], computer games [11] and RoboCup Soccer [12] [13]. However, there is no evidence in the literature of applications of this method in the wireless communications domain.

The purpose of this paper is to demonstrate how a case-based reinforcement learning algorithm for distributed dynamic spectrum assignment could be applied in a cellular network with dynamic topology, and how it improves its temporal blocking and dropping probability performance.

The rest of the paper is organised as follows: in Section II the network model and the learning problem are defined. In Section III the development of the RL algorithm for distributed dynamic spectrum assignment is described. Section IV introduces the concept of case-based RL and how it could be applied to the given spectrum assignment learning problem. In Section V the simulation results are discussed. Finally, conclusions are given in Section VI.

II. Dynamic Spectrum Assignment Learning Problem

A. Cellular Network Model

The cellular network used in this paper is a 6x6 km rural service area covered by a 3x3 grid of base stations spaced 2 km apart. The user equipment (UE) is geographically static and randomly distributed across the whole area. The network architecture is depicted in Figure 1. We stress that the spectrum assignment scheme used in this paper is fully distributed and does not employ any backhaul communications among

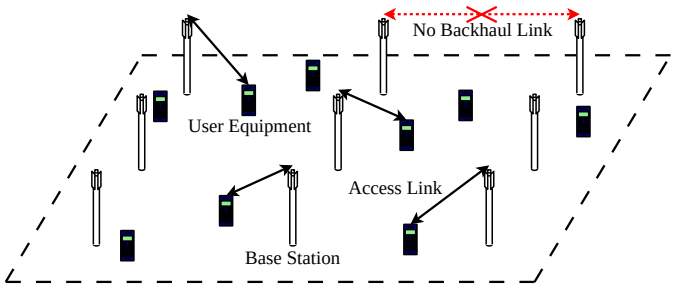

Figure 1. Network architecture

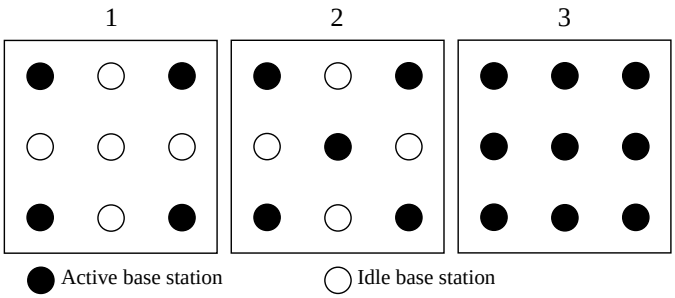

Figure 2. 3 phases of the network topology

the base stations or a centralised control unit. The backhaul network is only used to carry the mobile data traffic.

The other assumptions used in the network model are listed below:

- The available resources are divided into 20 logical channels. Adjacent-Channel Interference is assumed to be negligible and only uplink communications are considered.
- Fixed transmission power of 23 dBm and 2.6 GHz frequency band are used by all UEs.
- The minimum Signal to Interference plus Noise Ratio (SINR) for accepting a new call is 5 dB, and the calls are dropped if the SINR falls below 1.8 dB.
- The receiver noise floor is -100 dBm, obtained by assuming 290 K temperature, 10 MHz total bandwidth and 4dB noise figure.
- Each UE chooses which base station to connect to such that the overall attenuation of its signal is minimised.
- The transmission is continuous until a call is completed.

B. Dynamic Topology

Green topology management schemes switch the base stations on and off to optimise a trade-off between network QoS and energy savings, based on the amount of local traffic received at each base station [6] [7]. This results in cellular networks with dynamic topologies, where the environment for reinforcement learning based dynamic spectrum assignment algorithms is highly dynamic and challenging to control.

The experiments reported in our paper do not implement any particular topology management scheme. However, the dynamic nature of the network topology is simulated by making the network alternate among 3 different phases shown in Figure 2. This assumption was made because this paper focuses on the spectrum assignment algorithms and how they perform in the networks with dynamic topologies. Topology management itself is beyond the scope of this paper.

C. Radio Propagation Model

The propagation model used to calculate the path loss between the UE transmitters and the base station receivers is the WINNER II model [14]. In particular, the variation designed for line-of-sight (LOS) rural macro-cell scenarios is used (WINNER II D1), since it is most relevant to the network architecture discussed in this paper. The equation for calculating path loss using this model is given below:

$$PL = 21.5log_{10}(d) + 44.2 + 20log_{10}(0.2f_c) + SFL \quad (1)$$

where PL is the path loss in dB, d is the distance between the receiver and the transmitter in metres, f_c is the carrier frequency in GHz and SFL is the log-normal shadow fading loss with the standard deviation of 4dB and 0dB mean.

D. Traffic Model

The call arrival rate is modelled as a Poisson process with a constant mean arrival rate of λ_{UE} (calls per minute per UE) for all UEs in the network. Therefore, the traffic load is approximately uniform across the whole service area. The call duration is also an exponentially distributed random variable with the mean holding time of 1 minute.

E. Learning Objective

The objective of the learning problem investigated in this paper is for all base stations to prioritise among the available channels in a fully distributed fashion, only by trial-and-error. No communication between the base stations is assumed in order to achieve this objective. Therefore, it is a problem of distributed dynamic spectrum assignment.

Network-wide probabilities of call blocking (BP) and dropping (DP) are used to assess the performance of the spectrum assignment algorithms described in this paper. The network is assumed to be serviceable only if the BP does not exceed 5% and the DP does not exceed 0.5%. In general, call dropping is considered significantly less tolerable than blocking. Therefore, it is justifiable to set the DP threshold 10 times lower than that for BP [15] [16].

III. Reinforcement Learning

Reinforcement learning is a model-free type of machine learning which is aimed at learning the desirability of taking any available action in any state of the environment only by trial-and error [1]. This desirability of an action is represented by a numerical value known as the Q-value - an expected cumulative reward for taking a particular action in a particular state. The job of a RL algorithm is to estimate the Q-values for every action in every state, which are all stored in an array known as the Q-table. In some cases where an environment

is not represented by states, only the action space and a 1-dimensional Q-table are considered [17]. This is also the case investigated in this paper.

A. Stateless Q-Learning

One of the most successful and widely used RL algorithms is Q-learning, introduced in [18]. Since the learning problem described in the previous section does not require a state representation, a simple stateless variation of this algorithm, formulated in [17], is used in this paper.

Each base station maintains a Q-table such that every channel has an expected reward or Q-value associated with it. The Q-value represents the desirability of assigning a particular channel to an arriving call. Upon each call arrival, the base station either assigns an available channel to the call or blocks it if no channels are available.

The Q-table is updated by the corresponding base station each time it attempts to assign a channel to an arriving call. The update equation for stateless Q-learning, as defined in [17], is given below:

$$Q'(c) = Q(c) + \alpha(r - Q(c)) \tag{2}$$

where $Q(c)$ and $Q'(c)$ represent the Q-value of the selected channel before and after the update respectively, r is the reward associated with the most recent trial and determined by the reward function, and $\alpha \in [0, 1]$ is the learning rate parameter which weights recent experience with respect to previous estimates of the Q-values.

B. Q-table Initialisation and Reward Function

The values in the Q-table are initialised to zero, so all base stations start learning with equal choice among all available channels.

The reward function returns two discrete values:

- $r = -1$, if the call is blocked due to SINR being lower than 5 dB on the selected channel.
- $r = 1$, if the connection is successfully established using the channel chosen by the base station, i.e. if SINR is higher than 5 dB.

C. Action Selection Strategy

The main role of an action selection strategy is to provide a balance between exploration and exploitation in an RL problem [1]. However, the problem discussed in this paper is simpler than most classical RL problems in one fundamental aspect - it is stateless. It is also a multi-agent (i.e. distributed) RL problem, which means that the decisions made by each learning agent will affect the learning process of the other independent agents.

Therefore, a greedy action selection policy is used in this paper, i.e. each base station always selects an available channel with the highest Q-value, if any. In this way, if a base station discovers a good set of channels, it will continue using it to maximise performance and to make it easier for neighbouring base stations to learn to avoid the same channels. Investigating the effect of different action selection strategies on the algorithm performance is beyond the scope of this paper.

D. Learning Rate

Each base station in the network learns independently, and the learning environment, as perceived by each individual learning agent, depends on the choices made by other learning agents. Therefore the environment is locally dynamic from the viewpoint of each individual base station. Furthermore, the environment investigated in this paper is also globally dynamic due to changes in network topology explained in Subsection II-B.

Fixed values of the learning rate (α) are well-suited to such dynamic learning problems, since they essentially introduce the effect of a moving window, where the impact of older rewards on the current estimate gradually fades away, as seen from Equation (2).

The benefits of employing the Win-or-Learn-Fast (WoLF) learning rates [19] for RL-based dynamic spectrum assignment in cellular networks has been demonstrated in [20]. The WoLF principle states that the learning agent should learn faster when it is losing and more slowly when winning [19]. The learning rates used in the spectrum assignment scheme in [20] were $\alpha_{pos} = 0.05$ for a successful call arrival and $\alpha_{neg} = 0.2$ for a blocked call.

Figure 3 shows an average time response of 3 learning rate schemes. It was obtained by conducting 50 experiments with the same parameters and taking an average of 50 samples for every corresponding point on the graph. 450 UEs were randomly distributed across the network with all 9 base stations switched on (phase 3 from Figure 2). The call arrival rate was $\lambda_{UE} = 0.025$ calls per minute per UE which corresponds to an 11.25 Erlang network-wide traffic load. The first 2 schemes used a fixed learning rate of 0.05 and 0.2, but the 3rd scheme used the WoLF learning rate ($\alpha_{pos} = 0.05$ and $\alpha_{neg} = 0.2$). The latter scheme has consistently outperformed the others by showing a better learning speed at the start and still having lower BP and DP after 24 hours of learning.

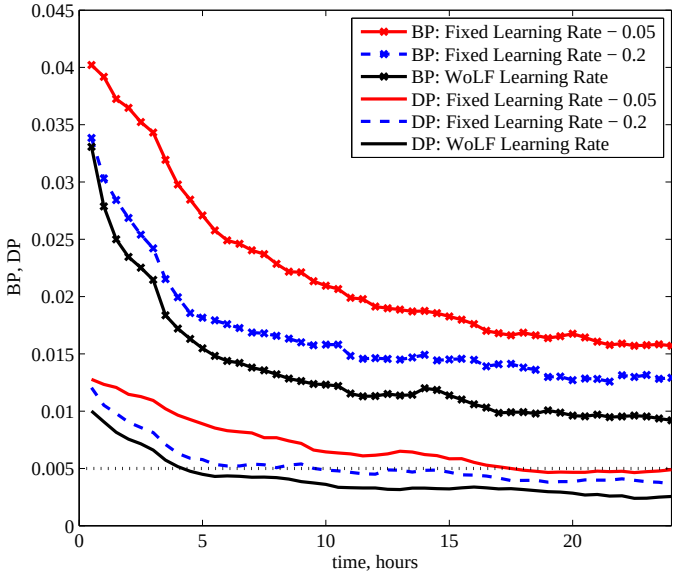

Figure 3. Blocking (BP) and dropping (DP) probability time responses using fixed and Win-or-Learn-Fast (WoLF) learning rate schemes

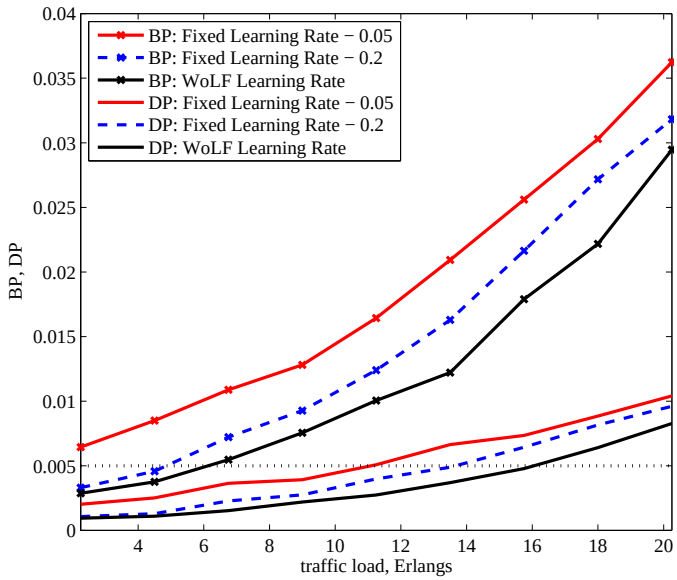

Figure 4. Steady state blocking (BP) and dropping (DP) probabilities using fixed and Win-or-Learn-Fast (WoLF) learning rate schemes at different traffic loads

Figure 4 shows that this improvement in BP and DP performance after 24 hours of learning is consistent across the whole range of serviceable traffic loads, constrained by the DP which at some point exceeds its 0.5% limit for all 3 schemes. Therefore, this WoLF learning rate scheme was used in the rest of the experiments presented in this paper.

IV. Case-Based Reinforcement Learning

Case-based reinforcement learning is a combination of reinforcement learning and case-based reasoning, where the solutions to previously known problems are used for helping to learn solutions to new problems. This approach is naturally suited to learning in dynamic environments with several identifiable phases such as cellular networks with dynamic topologies described in Section II.

Figure 5 shows a flow diagram of the processes involved in case-based RL. It also demonstrates that it is an extension of regular RL, i.e. regular RL can be viewed as a special case of case-based RL.

In Figure 5, the unfilled blocks and solid lines constitute a flow diagram of a regular RL algorithm. There is an outer output-state-action loop, where outputs of the environment are observed and processed to yield the environment state information, and then the best action is chosen for the current state based on the policy of the learning agent. In the context of our dynamic spectrum assignment problem, the output of interest is whether or not the last call got blocked, and the action is a channel allocated to an arriving call. There is also an inner learning loop, whose role is to learn a good policy to be used by the learning agent. It achieves this goal by observing the actions taken by the learning agent and their outcomes and directly estimating the entries in the Q-table. A policy is then derived from the estimated Q-table and used for choosing an action in the current environment state.

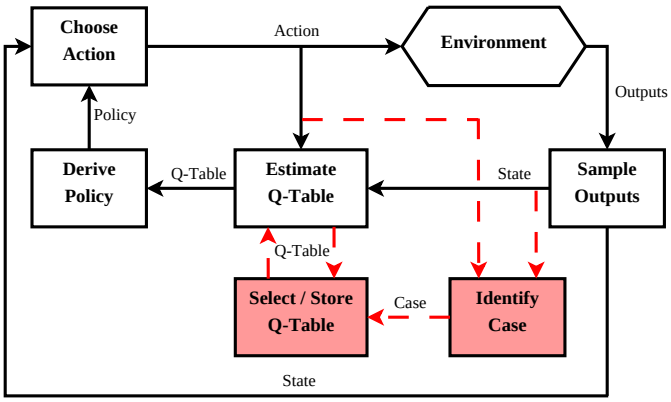

Figure 5. Flow diagram of case-based reinforcement learning

The highlighted blocks and dashed arrows represent additional features of case-based RL to enable the system to learn several solutions to different phases of the environment at once. It introduces another parallel inner loop which continuously observes the input/output relationship of the environment and identifies its current model or *case*, as referred to in the case-based reasoning domain. In some circumstances it may also have access to other information supplied from elsewhere to aid the identification process. The idea is that for different phases of the environment the estimated models will be sufficiently different to be detected by the identification algorithm, and for every identified model of the environment there will be a stored Q-table associated with it. In this way, a case-based RL algorithm will always know what phase the environment is currently in and will be able to use a Q-table most suitable for this phase.

Algorithm 1 shows how a case-based stateless Q-learning algorithm was implemented for dynamic spectrum assignment at each base station. The lines in italics are specific to case-based reinforcement learning. If they are removed, the algorithm becomes simple stateless Q-learning used in [20].

Algorithm 1 Case-based stateless Q-learning algorithm for distributed spectrum assignment in cellular networks

```
1:  Initialise Q-table
2:  while base station is on do
3:      Wait for a call arrival
4:      if all channels are occupied then
5:          Block call
6:      else
7:          Identify network model
8:          Choose Q-table based on identified model
9:          Assign a free channel with the highest Q-value
10:         Observe the outcome
11:         Update Q-table using (2)
12:         Store Q-table and associate it with identified model
13:     end if
14: end while
```

The network model is defined as a 9 element binary vector indicating the on/off status of every base station in the network. Investigating the methods of case identification for case-based RL is an important and challenging task. However, it is outside of the scope of this paper. Therefore, it is assumed that every

base station is capable of identifying a topology phase of the network, for example, by observing the set of UEs connected to it or by sending short broadcast messages via a backhaul link.

V. Simulation Results

The RL algorithm described in Section III with and without case-based reasoning features was simulated on a 9 base station cellular network, introduced in Section II, with 450 UEs randomly distributed across the service area. All simulation results shown in this section were obtained by averaging over 50 experiments with identical parameters for each traffic load value to ensure their general validity. Furthermore, every pair of case-based RL vs regular RL experiments were simulated using identical call arrival and departure times to guarantee fair comparison between 2 schemes.

The time responses shown in this section are for a network with dynamic topology which changes its phase every 5 hours as explained in Subsection II-B. The sequence of phases used for this simulation is $[1, 2, 3, 2, 3, 1, 2, 1, 3, 1, 2, 3, 2, 3, 1, 2, 1, 3]$. The phase indices correspond to those defined in Figure 2. This sequence was designed to include each phase 6 times, as well as every possible transition between any 2 of them.

Figure 6 shows the network-wide BP time response using case-based RL and its equivalent regular RL alternative at 2 different traffic loads - 4.5 and 11.25 Erlangs. Case-based RL does not cause any significant improvement in performance over regular RL at a lower traffic load. However, at a higher traffic load, the case-based RL algorithm performs significantly better after transitions to the previously visited phases, which results in a lower BP level overall. The high positive rate of change of BP straight after phase transitions suggests that a large number of calls is blocked in those short periods of time. This deterioration in performance is significantly mitigated using case-based RL. The slight mismatch in performance during the first 15 hours is due to the randomness of channel selection at the early stages of learning, where all base stations start with an all-zero Q-table.

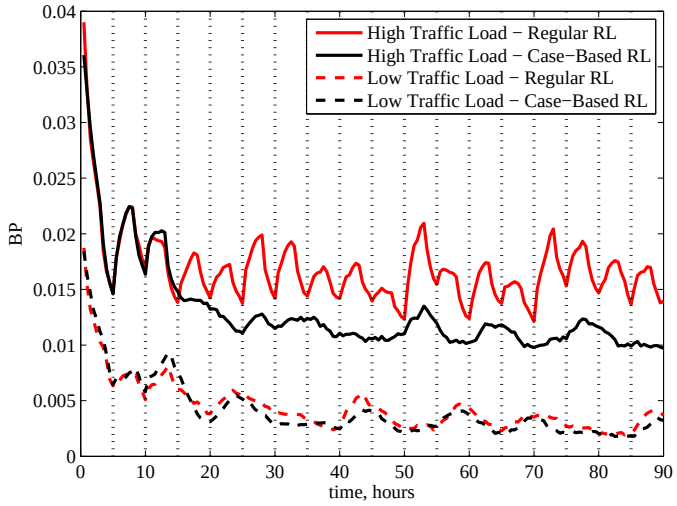

Figure 6. Blocking probability (BP) time response using regular and case-based reinforcement learning

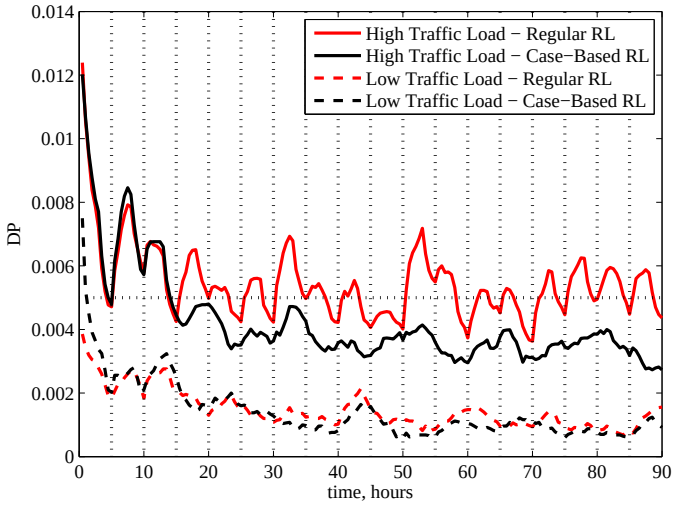

Figure 7. Dropping probability (DP) time response using regular and case-based reinforcement learning

Figure 7 shows that the DP time responses for the same simulation experiments follow very similar characteristics. However, the DP using regular RL at a higher traffic load exceeds its 0.5% acceptable limit after most transitions. This effectively makes the network unusable, whereas the case-based RL algorithm kept the DP safely within its limits.

Figures 8 and 9 show a comparison of the initial (first 15 hours) and final (last 15 hours) performance of the case-based and regular RL algorithms at different traffic loads. As expected, the initial performance of the 2 schemes is approximately the same in terms of BP. There is a bigger difference between the initial DP curves, because it is more random and less controllable than BP. In distributed dynamic spectrum assignment schemes used in this paper the base stations are not aware if they caused any dropped calls in other cells, the only feedback they get is blocking of their own calls.

In terms of the final steady state performance during

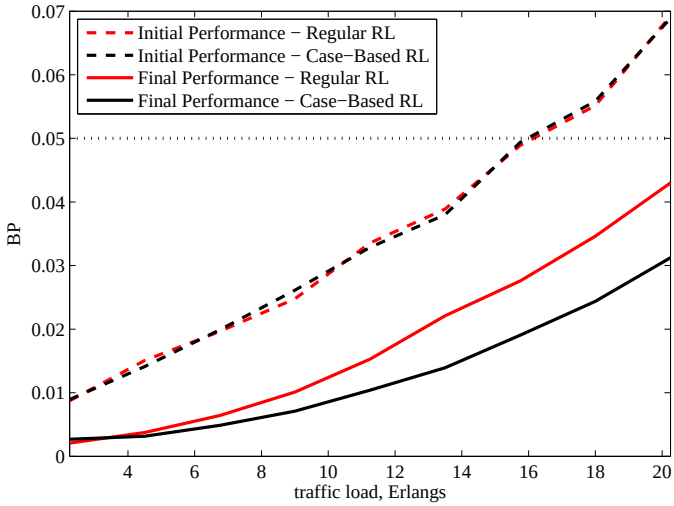

Figure 8. Initial and final blocking probability (BP) of the network using regular and case-based reinforcement learning

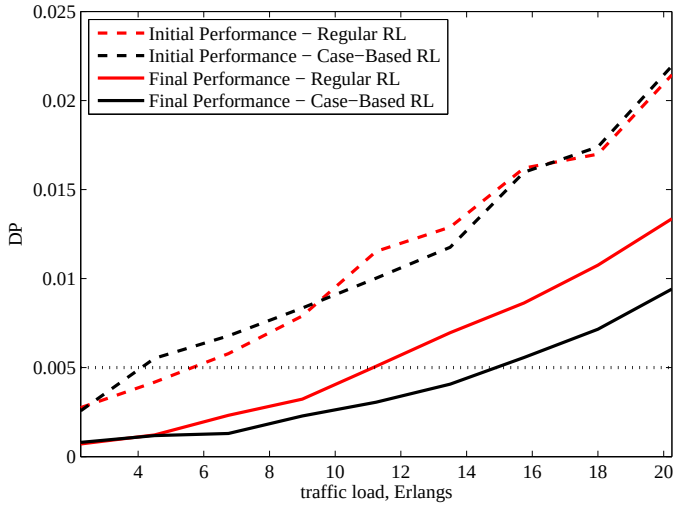

Figure 9. Initial and final dropping probability (DP) of the network using regular and case-based reinforcement learning

the last 15 hours, augmenting RL with case-based reasoning functionality caused a significant improvement at higher traffic loads. Once again, the serviceable range of traffic loads is determined by the DP of the network which, unlike the BP, exceeds its maximum acceptable limit (0.5%) at some point for both schemes. The improvement of case-based RL over regular RL in steady state performance resulted in $\approx 35\%$ increase in the maximum usable network traffic load.

These results clearly demonstrate the type of expected performance improvements that can be obtained by employing case-based reinforcement learning approach for distributed dynamic spectrum assignment in cellular networks with dynamic topologies.

VI. Conclusion

We have developed a case-based reinforcement learning (RL) algorithm, i.e. a combination of RL and case-based reasoning, for distributed dynamic spectrum assignment in mobile cellular networks. Simulations have shown that augmenting a stateless Q-learning algorithm with case-based reasoning functionality has significantly improved the temporal performance of a 9 base station network with dynamic topology. It has mitigated the performance degradation in terms of the probabilities of blocking and dropping after transitions between different phases of the network topology. This increased the maximum serviceable traffic load of the network by $\approx 35\%$. Although case-based RL has not been applied in the wireless communications domain, this paper has demonstrated that this approach is naturally suited to dynamic spectrum assignment problems in cellular networks with dynamic topologies.

Acknowledgment

This work has been funded by the ABSOLUTE Project (FP7-ICT-2011-8-318632), which receives funding from the 7th Framework Programme of the European Commission.

References

[1] R. Sutton and A. Barto, *Reinforcement Learning: An Introduction*. MIT Press, Cambridge, MA, 1998.

[2] A. Feki, V. Capdevielle, and E. Sorsy, "Self-organized resource allocation for lte pico cells: A reinforcement learning approach," in *Vehicular Technology Conference (VTC Spring), 2012 IEEE 75th*, 2012, pp. 1–5.

[3] Y. Teng, Y. Zhang, F. Niu, C. Dai, and M. Song, "Reinforcement learning based auction algorithm for dynamic spectrum access in cognitive radio networks," in *Vehicular Technology Conference Fall (VTC 2010-Fall), 2010 IEEE 72nd*, 2010, pp. 1–5.

[4] T. Jiang, D. Grace, and P. D. Mitchell, "Efficient exploration in reinforcement learning-based cognitive radio spectrum sharing," *Communications, IET*, vol. 5, no. 10, pp. 1309–1317, Jul. 2011.

[5] Q. Zhao and D. Grace, "Application of cognition based resource allocation strategies on a multi-hop backhaul network," in *Communication Systems (ICCS), 2012 IEEE International Conference on*, 2012, pp. 423–427.

[6] Y. Han, D. Grace, and P. Mitchell, "Energy efficient topology management for beyond next generation mobile broadband systems," in *Wireless Communication Systems (ISWCS), 2012 International Symposium on*, 2012, pp. 331–335.

[7] S. Rehan and D. Grace, "Combined green resource and topology management for beyond next generation mobile broadband systems," in *Computing, Networking and Communications (ICNC), 2013 International Conference on*, 2013, pp. 242–246.

[8] T. Gabel and M. Riedmiller, "Multi-agent case-based reasoning for cooperative reinforcement learners," in *Proceedings of the 8th European conference on Advances in Case-Based Reasoning*, ser. ECCBR'06. Berlin, Heidelberg: Springer-Verlag, 2006, pp. 32–46.

[9] I. Watson, "Case-based reasoning is a methodology not a technology," *Knowledge-Based Systems*, vol. 12, no. 56, pp. 303 – 308, 1999.

[10] C. Jiang and Z. Sheng, "Case-based reinforcement learning for dynamic inventory control in a multi-agent supply-chain system," *Expert Syst. Appl.*, vol. 36, no. 3, pp. 6520–6526, Apr. 2009.

[11] B. Auslander, S. Lee-Urban, C. Hogg, and H. Muñoz Avila, "Recognizing the enemy: Combining reinforcement learning with strategy selection using case-based reasoning," in *Proceedings of the 9th European conference on Advances in Case-Based Reasoning*, ser. ECCBR '08. Berlin, Heidelberg: Springer-Verlag, 2008, pp. 59–73.

[12] R. Bianchi, R. Ros, and R. Lopez De Mantaras, "Improving reinforcement learning by using case based heuristics," in *Proceedings of the 8th International Conference on Case-Based Reasoning: Case-Based Reasoning Research and Development*, ser. ICCBR '09. Berlin, Heidelberg: Springer-Verlag, 2009, pp. 75–89.

[13] L. Celiberto, J. Matsuura, R. Lopez de Mantaras, and R. Bianchi, "Reinforcement learning with case-based heuristics for robocup soccer keepaway," in *Robotics Symposium and Latin American Robotics Symposium (SBR-LARS), 2012 Brazilian*, 2012, pp. 7–13.

[14] P. Kyösti, J. Meinilä, L. Hentilä, X. Zhao, T. Jämsä, C. Schneider, M. Narandzić, M. Milojević, A. Hong, J. Ylitalo, V. Holappa, M. Alatossava, R. Bultitude, Y. de Jong, and T. Rautiainen, "IST-4-027756 WINNER II D1.1.2 v1.2 WINNER II channel models," Feb. 2008.

[15] D. Akerberg and F. Brouwer, "On channel definitions and rules for continuous dynamic channel selection in coexistence etiquettes for radio systems," in *Vehicular Technology Conference, 1994 IEEE 44th*, 1994, pp. 809–813 vol.2.

[16] D. Grace, "Distributed Dynamic Channel Assignment for the Wireless Environment," Ph.D. dissertation, University of York, UK, 1998.

[17] C. Claus and C. Boutilier, "The dynamics of reinforcement learning in cooperative multiagent systems," in *Proceedings of the fifteenth national/tenth conference on Artificial intelligence/Innovative applications of artificial intelligence*. American Association for Artificial Intelligence, 1998, pp. 746–752.

[18] C. Watkins, "Learning from Delayed Rewards," Ph.D. dissertation, University of Cambridge, England, 1989.

[19] M. Bowling and M. Veloso, "Multiagent learning using a variable learning rate," *Artificial Intelligence*, vol. 136, pp. 215–250, 2002.

[20] N. Morozs, T. Clarke, and D. Grace, "A novel adaptive call admission control scheme for distributed reinforcement learning based dynamic spectrum access in cellular networks," in *The Tenth International Symposium on Wireless Communication Systems 2013 (ISWCS 2013)*, Ilmenau, Germany, Aug. 2013.