Self-Organizing Networks: A Packet Scheduling Approach for Coverage/Capacity Optimization in 4G Networks Using Reinforcement Learning

Moazzam Islam Tiwana[1], Syed Junaid Nawaz[2], Ataul Aziz Ikram[3], Mohsin Islam Tiwana[4]

[1,2]Department of Electrical Engineering, COMSATS Institute of Information Technology (CIIT), Islamabad, 44000, Pakistan

[3]Department of Electrical Engineering, National University of Computer and Emerging Sciences, Islamabad, 44000, Pakistan

[4]Department of Mechatronics Engineering, National University of Sciences and Technology, H-12, Islamabad, 44000, Pakistan

moazzam_islam@comsats.edu.pk

***Abstract*—The next generation mobile networks LTE and LTE-A are all-IP based networks. In such IP based networks, the issue of Quality of Service (QoS) is becoming more and more critical with the increase in network size and heterogeneity. In this paper, a Reinforcement Learning (RL) based framework for QoS enhancement is proposed. The framework achieves the coverage/capacity optimization by adjusting the scheduling strategy. The proposed self-optimization algorithm uses coverage/capacity compromise in Packet Scheduling (PS) to maximize the capacity of an eNB subject to the condition that minimum coverage constraint is not violated. Each eNB has an associated agent that dynamically changes the scheduling parameter value of an eNB. The agent uses the RL technique of Fuzzy Q-Learning (FQL) to learn the optimal scheduling parameter. The learning framework is designed to operate in an environment with varying traffic, user positions, and propagation conditions. A comprehensive analysis on the obtained simulation results is presented, which shows that the proposed approach can significantly improve the network coverage as well as capacity in terms of throughput.**

***Index Terms*—Packet Scheduling, LTE, Reinforcement Learning, Fuzzy Q-Learning, SON.**

I. INTRODUCTION

The mobile networks have undergone an enormous growth in terms of size and complexity during the last few years. This has resulted in a significant increase in the Operational Expenditure (OPEX) of the Next Generation Mobile Networks (NGMN) [1]. In this context, self-optimization has been included in LTE standardisation as part of the Self Organizing Networks (SON) [2], [3]. The objective of self-optimization is to decrease the network OPEX by introducing automation into the network. While at the same time, we enhance the Quality of Service (QoS) of the network by the optimal setting of Radio Resource Management (RRM) parameters. The network QoS is measured in terms of Key Performance Indicators (KPIs) related to coverage and capacity. SON entities are supposed to operate in an environment with varying traffic, changing propagation conditions, newly introduced services, and evolving management policies of the operator.

Academia and industry have worked on the self-optimisation in Radio Access Networks (RANs) since the last decade [4], [5]. Automated self-optimisation enables the operators to enhance the network performance and profitability while at the same time reducing the amount of management operations. The self-optimization algorithms can be implemented in the Operation and Maintenance Centre (OMC) of a network. Self-optimisation has been investigated for the land mobile radio cellular communication technologies of GSM and UMTS as in [6]–[8]. However, despite all these industrial and academic research efforts, self-optimization was not included as a part of UMTS standard. Research has been extended to self-optimization in heterogeneous applications, mainly for load balancing purposes [9]. With the advent of LTE, the focus of research shifted to the self-optimization of LTE. The recent research on LTE self-optimization has mainly focused on dynamically optimising Radio Resource Management (RRM) parameters, like: resource and bandwidth allocation [10], Inter-Cell Interference Coordination (ICIC) [11], [12] and load balancing [13], [14].

It has been shown in [9] and [15] that rules of Fuzzy Logic Controller (FLC) can be optimized using Q-Learning (QL). Consequently, these rules are used for automatic network parameter optimization. FLC has the ability to model a controller as a set of ’IF-THEN’ rules. Such rules may be designed by using some previous history of the network behaviour. However, in the case when no such previous knowledge is available, Reinforcement Learning (RL) techniques such as QL can be used to derive/optimize FLC rules. Such Fuzzy QL (FQL) algorithm has been used in [9] to achieve performance optimization by dynamic load balancing between UMTS and LAN networks. FQL has also been used for the optimization of mobility parameters of

Manuscript received January 11, 2014; accepted September 28, 2014.

both GSM Edge Radio Access Network (GERAN) [15] and LTE network [16]. More recently, FQL has been used for coverage/capacity optimization of LTE networks by adjusting the vertical tilt angle of the antenna employed at eNBs [17]–[19].

This paper examines the use of FQL to optimize the Packet Scheduling (PS) to achieve maximum eNB capacity while satisfying the minimum coverage constraint. α-fair scheduling is the type of PS used in this work [20]. α-fair scheduler provides a generalization of the well-known schedulers including Proportional Fair (PF), Max Throughput (MTP), and Max-Min Fair (MMF) schedulers. The α parameter of an eNB can be tuned to achieve a compromise between capacity (higher throughput for its mobile users) and coverage (serving higher number of users at a time). Hence, in the case of eNBs with degraded performance, the optimization process trades capacity for coverage to achieve the required minimum coverage constraint. For the eNBs satisfying the minimum coverage requirement, coverage is traded for capacity to achieve additional capacity gain for such eNBs.

The contribution of this paper is the proposal of a novel self-optimization procedure for coverage/capacity optimization based on PS. Furthermore, this approach has the advantage of being scalable with increasing network size; this is because, adjusting α-parameter of an eNB has very little impact on the KPIs of its neighbours [21]. The targeted network architecture for the proposed scheme is LTE and LTE-A. The simulation results have been obtained for the case study of LTE network, which show significant improvement in the network performance. Since, the basic network architecture of LTE and LTE-A is the same, apart from some new added features in LTE-A, like carrier aggregation, enhanced MIMO, and Coordinated Multipoint (CoMP) transmission. Therefore, the proposed scheme is also valid for the case of LTE-A networks.

The rest of the paper is organized as follows: Section II presents the details of α-fair scheduler used in our case study. Section III describes the Multi-Agent RL based framework. Section IV details the FQL algorithm along with its various components to solve the Multi-Agent RL problem. Section V describes the simulation environment and provides the obtained simulation results along with a thorough analysis of the results. Section VI concludes the paper.

II. α-FAIR SCHEDULER

LTE uses OFDMA as the radio access technology. Consider an LTE eNB with frequency bandwidth subdivided into K Physical Resource Blocks (PRBs). N users are attached to the eNB. P denotes the scheduling policy that schedules a user on a given PRB at the scheduling instant $(t_u)_{u\in N}$ $.P^{k}_{i,t_u}$ represents that the user i is assigned to PRB k at instant t_u. $r^{(k)}_{i,t_u}$ is the instantaneous throughput of user i at instant t_u on PRB k. While the mean throughput of user i during time

$$\overline{r}^{(k)}_{i,t_{u+1}} = \left(1-\epsilon\right)\overline{r}^{(k)}_{i,t_u} + \epsilon\text{¢}_{P^{(k)}_{i,\,t_{u+1}}} r^{(k)}_{i,t_{u+1}}, \tag{1}$$

where, $\epsilon > 0$ is a small averaging parameter and δ represents the Kronecker's delta.

The user to be scheduled on PRB k at time t_{u+1} is selected as

$$i^{*(k)} = \arg\max_{0\le i\le N} \frac{r^{(k)}_{i,t_{u+1}}}{\left(\overline{r}^{(k)}_{t,t_u} + d\right)^{\sim}}, \tag{2}$$

where $\overline{r}_{i,t_0} = 0\,\forall i$. Here, $d > 0$ is chosen to have very small value that avoids singularity at zero.

Hence, the mean throughput of user i during the time interval $\left[t_0, t_u\right]$ can be calculated as

$$\overline{r}_{i,t_u} = \sum_{k=1}^{K} \overline{r}^{(k)}_{i,t_u}. \tag{3}$$

An eNB utilizes all its scheduling resources, i.e., PRBs even if at least a single user is connected with it. Therefore, changing the α parameter of an eNB will result in little effect on the neighbouring eNBs' KPIs.

Equations (2) and (3) show that for $\alpha = 0$, the α-fair scheduler acts as the MTP scheduler. Similarly, the α-fair scheduler changes from MTP to PF scheduler for $\alpha = 0 \rightarrow 1$. Furthermore, for $\alpha = 1 \rightarrow \infty$, the PF scheduler evolves into MMF scheduler. The capacity of the eNB given as $\overline{r}_i$ changes from a maximum value to a minimum value as $\alpha = 0 \rightarrow \infty$. While at the same time the coverage given as number of users served changes from a minimum to maximum value as PF scheduler tries to achieve fairness.

III. QL FOR SELF-OPTIMIZATION IN LTE

QL models the LTE network as a Multi-Agent RL [23] system where an agent is associated with each eNB. The agents interact in real time with the environment by sensing its state and taking an appropriate action to maximize the reward. The agent also exploits the knowledge gained from the experiences as a result of the past actions. The learning process is characterised as a Markov Decision Process (MDP) [12]. As due to the phenomenon like interference, mobility, change in UE traffic distribution and propagation conditions etc. the mobile network inherent dynamics follow a transitionary model.

QL is particularly useful for the optimization problems where the system model is not available as a closed-form expression. In such case, the learning problem is incrementally solved using Temporal Difference (TD) method [23]. In QL, an agent selects those actions which maximize the long term received reward, given as

$$R_t = r_t + \text{¥}\, r_{t+1} + \text{¥}^2 r_{t+2} + \text{¥}^3 r_{t+3} \cdots = \sum_{k=0}^{\infty} \text{¥}^k r_{t+k}, \tag{4}$$

where *r* denotes the instantaneous reward as a result of an action. γ represents the discount factor. If γ is close to 0, the agent/controller gives more importance to the maximization of immediate rewards. While for γ close to 1 the future rewards almost as important as immediate ones. The γ value is set to 0.95 in the present work [12].

Consider, an agent senses the initial environment state to be *s* and takes an action $b \in B$ as it follows a fixed policy π. *(s,b)* denotes the state-action pair. QL continuously updates and estimates the state-action pair to achieve objective in (4), as shown below

$$Q^{\text{®}}\left(s,b\right)=E_{\text{®}}\left[\sum_{k=0}^{\infty}\text{¥}^{k}r(s_t,b_t)\middle|s_o=s,b_o=b\right]. \tag{5}$$

The (5) is solved iteratively as follows [23]

$$Q_{t+1}\left(s_t,b_t\right)=Q_t\left(s_t,b_t\right)+\text{©}\left(s_t,b_t\right)\times$$
$$\times\left[r_{t+1}+\text{¥}\max\nolimits_{b'}Q_t\left(s_{t+1},b'\right)-Q_t\left(s_t,b_t\right)\right], \tag{6}$$

where, κ is learning rate with value between 0 and 1.

IV. FUZZY Q-LEARNING

QL algorithm solves the optimization problems where system state space is discrete. However, in the case of our LTE network optimization problem the KPIs and RRM parameters are continuous. Hence, the system states are also continuous, leading to enormous complexity. The problem is solved by using fuzzy logic to discretize the state and action spaces. A Fuzzy Inference System (FIS) [24] is shown in Fig. 1.

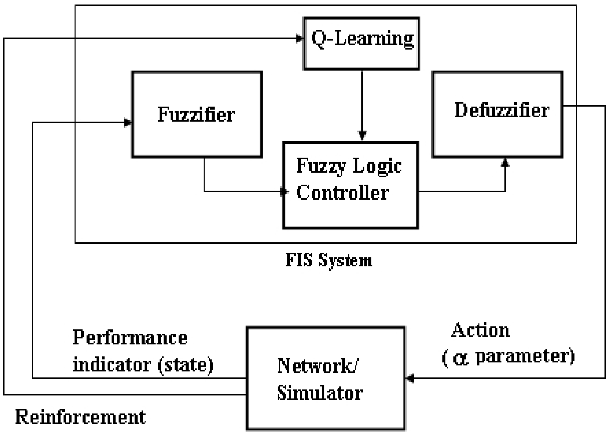

Fig. 1. Architecture of self-optimization procedure.

The state vector *s* is input to the FIS. Fuzzifier is the first element of FIS. It determines the degree to which each continuous (crisp) element of the state vector *s* belongs to each of the fuzzy sets using membership functions. This procedure is known as fuzzification. This degree of membership information is then used by Fuzzy Logic Controller (FLC) [12], [25] to calculate output action for each of the triggered rules. The process of defuzzification maps these actions into a crisp (continuous) value. The fuzzy rules are optimized using QL to form a Fuzzy QL (FQL) optimization process.

V. COMPONENTS OF FQL RL SYSTEM

The main components of the FQL based RL system, proposed in this paper, are given as below.

A. State

The proposed state vector, corresponding to eNB *c*, which is input to the FQL controller, is defined as follows

$$s_c=\left[\sim_c \quad BCR_c\right], \tag{7}$$

where, $\sim_c$ is the value of α parameter for the eNB *c*. While, BCR_c denotes the Block Call Rate (BCR) of eNB *c*.

B. Policy

The action of each eNB is to change its α according to the policy π. $\pi: s \rightarrow b$ maps the state *s* of an eNB to the action $b \in B$. Where, *B* is the set of all possible actions (α value for the eNB).

C. Instantaneous Reward

The reward in the proposed FQL system is the instantaneous average throughput per user r_t. Let *M* denote the total number of mobiles in active communication with the network at any given instant *t*, r_t is given as

$$r_t=\begin{cases}\sum\limits_{m\in M}\dfrac{m\left(th_t\right)}{M} & \text{for } BCR_{network}\leq BCR_{th},\\ -100 & \text{for } BCR>BCR_{th},\end{cases} \tag{8}$$

where $m\left(th_t\right)$ denotes the instantaneous throughput of the mobile *m*. The mean BCR of the network, denoted as $BCR_{network}$, is constrained to be less than the threshold value BCR_{th}.

D. FQL Algorithm Description

This section presents the FQL algorithm [24]. Let the state vector, $s=\left[s_1,\ldots,s_j,\ldots,s_J\right]$, where *j* is the *j*th element of state vector before fuzzification. After fuzzification, the membership function *T*(*s*) quantifies the degree of membership of an input value s_j to a specific fuzzy set corresponding to a fuzzy label. The fuzzy label of s_j, denoted as F_j, can be 'LOW', 'MEDIUM' and 'HIGH'. If *R* denotes all the rule of a FLC, then rule $r \in R$ is given as

$$\text{IF}\left(s_1\text{ is }F_1^r\right)\ldots\text{AND}\left(s_j\text{ is }F_j^r\right)\ldots\text{.AND}\left(s_j\text{ is }F_j^r\right)THEN$$
$$b=a^r\,with\;q(F^r,a^r), \tag{9}$$

where $F^r=\left[F_1^r,\ldots,F_j^r,\ldots,F_J^r\right]$ is the modal vector corresponding to rule *r* and represents a fuzzy state. While a^r is the fuzzy label for the action corresponding to F^r. The q-value $q\left(F^r,a^r\right)$ corresponding to fuzzy state F^r and action a^r, is initialized to zero. The degree of truth T_r

for each rule *r*∈*R* is given as

$$T_r\left(s\right)=\prod_{j=1}^{J} m_{F_j^r}\left(s_j\right), \qquad (10)$$

where $m_{F_j^r}$ is the membership function of s_j for label F_j^r.

Exploration/exploitation policy (EEP) dictates the action chosen for each of the activated rules. EEP policy uses *ε*-greedy method for choosing the actions:

$$\begin{cases} \forall r\in R: a^r=\arg\max_{l\in L} q\left(F^r,a^l\right), & \text{with prob.}\,\epsilon\ , \\ \forall r\in R: a^r=\text{random}_{l\in L}\, q\left(a^l\right), & \text{with prob.}\,1-\epsilon\ , \end{cases} \qquad (11)$$

where *L* denotes the indices of the of the set of possible actions for a given triggered rule *r*. The ϵ can be assigned a value between the interval [0, 1] to determine the exploration/exploitation compromise. The inferred action, after the *defuzzification,* for a given input state vector *s* and the triggered rules in *R* are given as

$$a\left(s\right)=\sum_{r\in R} T_r(s)a^r. \qquad (12)$$

The associated quality of the rules is calculated as

$$Q\left(s,b\left(s\right)\right)=\sum_{r\in R} T_r\left(s\right)q\left(F^r,a^r\right). \qquad (13)$$

Now as a result of the applied action, the eNB transits to a new state s_{t+1}. The value function $V\left(s_{t+1}\right)$ is calculated as

$$V\left(s_{t+1}\right)=\sum_{r\in R} T_r\left(s_{t+1}\right)\ \max_{l\in L}\ q\left(F^r,a^l\right). \qquad (14)$$

The updating of q value requires that first difference between the quality value ΔQ of the old and the new state be calculated as

$$\Delta Q=r_{t+1}+¥V_t\left(s_{t+1}\right)-Q\left(s_t,b\left(s_t\right)\right). \qquad (15)$$

The *q* values can now be updated using the normal gradient descent method

$$q_{t+1}\left(F^r,a^r\right)=q_t\left(F^r,a^r\right)+©T_r\left(s_t\right)\Delta Q, \qquad (16)$$

where © is the learning rate.

E. Simulation Scenario

An LTE network consisting of 12 eNBs is simulated using a semi-dynamic simulator. The details of simulator are given in [13]. The traffic model used is the downlink streaming to support H.264 with variable bitrates from 64 Kbits/sec to 50 Mbits/sec. The detailed parameters for the simulated dense urban scenario are given in Table I.

Monte Carlo simulations are performed by taking the snapshots of the network evolution with the resolution of one second time step. At each time step Call Admission Control (CAC) is performed for new users, mobile positions are updated and Handover (HO) events are processed. Furthermore, the mobiles that are dropped or complete their streaming session duration, leave the network.

TABLE I. SYSTEM LEVEL SIMULATION PARAMETERS.

Parameter	Settings
System bandwidth	5 MHz
Cell layout	12 eNBs, single sector
Maximum eNB transmit power	32 dBm
Inter-site distance	1.5 KM to 2 KM
Subcarrier spacing	15 kHz
PRBs	25
Propagation Model	L = 128.1 + 37.6 $\log_{10}(R)$, R in kilometers
Shadowing standard deviation	6 dB
Traffic model	streaming to support H.264 video bitrates
Streaming session duration	5 s
Packet scheduling scheme	α-fair scheduling
Mobility of mobiles	0 %

The description of CAC procedure is given as follows: when a new mobile user arrives, (3) using (2) calculates its bitrate along with calculating the bitrate of already scheduled users. (2) uses the quality tables, obtained from link level simulations, to calculate $r_{i,t_{u+1}}^{(k)}$ from $S_{i,t_{u+1}}^{(k)}$. Here $S_{i,t_{u+1}}^{(k)}$ denotes the SINR of user *i* at instant t_{u+1} on PRB *k*. If the bitrate of the new mobile user is above 64 Kbits/sec, it is admitted to the network. Otherwise, it is blocked. The streaming session of a mobile is terminated prematurely (dropped) if its bit rate falls below the threshold value of 64 Kbits/sec. The mean Average Bitrate (ABR) of mobiles in an eNB, is used as KPI of an eNB’s capacity. While, mean BCR is used as KPI of an eNB’s coverage. A lower value of α for an eNB signifies that lower SINR users are assigned less resources (PRBs). Hence, the CAC procedure may not allow a lower SINR user with bitrate less than 64 Kbps to be accepted in the network. This further results in an increase of mean BCR of the eNB (i.e., bad coverage). While at the same time, mean ABR also increases (i.e., good capacity) as higher SINR users are assigned more resources. On the contrary, higher α value results in more resources being assigned to lower SINR users to achieve fairness among all the users. Hence, mean BCR decreases with an increase in admitted mobile user to an eNB (i.e., good coverage); while, mean ABR decreases (i.e., bad capacity).

The simulator operates in two modes i.e., static and dynamic mode. In static mode, there is no self-optimization. The simulator runs for 5000 time steps with default α value set to 1 for all eNBs. The KPIs are calculated by computing the average for the time steps from 500 to 5000. Here, the initial 499 seconds are not considered, as initially the network is in transient state. In the dynamic mode or self-optimization mode, the FQL algorithm adapts the α of an eNB with the periodicity of 50 seconds. The learning rate is set to value of $\kappa=0.1$, as taken in [12]. The simulations are performed over a time period of 150000 seconds.

F. *Simulation Results*

The results obtained by the α adaptation using the FQL approach have been compared with the reference system, where α is fixed to the value of 1. Here, global mean ABR of mobiles in network is an indicator of network capacity while global mean Access Probability (AP), which is (1-mean BCR), is an indicator of network coverage.

Figure 2 compares the global mean ABR of mobiles of the two systems. The application of self-optimization results in significant improvement in the performance as compared to the case with no self-optimization. A maximum improvement of up to 10 % can be observed for the traffic value of 4 arrivals/sec. On the other hand, Fig. 3 shows a slight degradation in the global mean AP for the self-optimisation case. The coverage has been compromised/traded with capacity (smaller α parameter values for eNBs). However, the degraded value of global mean AP does not fall below the threshold of 90 % up till the traffic value of 5 arrivals/sec. Beyond this traffic value, the mean AP falls below the threshold of 90 %. Hence, it is no further possible to trade coverage for capacity. It can thus be established that the proposed optimisation technique achieves a substantial improvement of up to 10 % in the capacity i.e., mean ABR from traffic value of 1 up to 5 arrivals/sec.

This analysis is further elaborated by the CDF plots of ABR of individual mobiles for the traffic values of 1, 3 and 5 arrivals/sec in Fig. 4, Fig. 5, and Fig. 6, respectively. For convenience of the readers, the average bit rate values for which the trend of the comparative graphs start to show converse behaviour have been marked as points A, B, and C in Fig. 4, Fig. 5, and Fig. 6, respectively. In Fig. 4, it is evident for $|F(x)|$ = 0.2, that the ABR of optimized curve is 320 Kbps less than that of non-optimized curve. On the other hand, for $|F(x)|$ = 0.6, the ABR of the optimized curve is observed to be 780 Kbps more compared to that of non-optimized curve. This difference is due to the fact that for smaller values of α, the α-fair scheduler assigns more resources to high SINR users at the cost of low SINR resources, to maximize the throughput. Hence, high SINR users have more bitrate as compared to low SINR users. For the traffic value of 3 arrivals per second, the network load starts to increase and the global mean AP decreases.

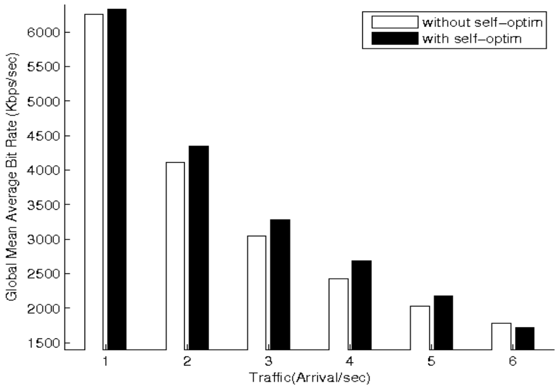

Fig. 2. Mean Average Bit Rate as a function of the traffic intensity for auto-tuned α parameter compared with fixed $\alpha = 1$.

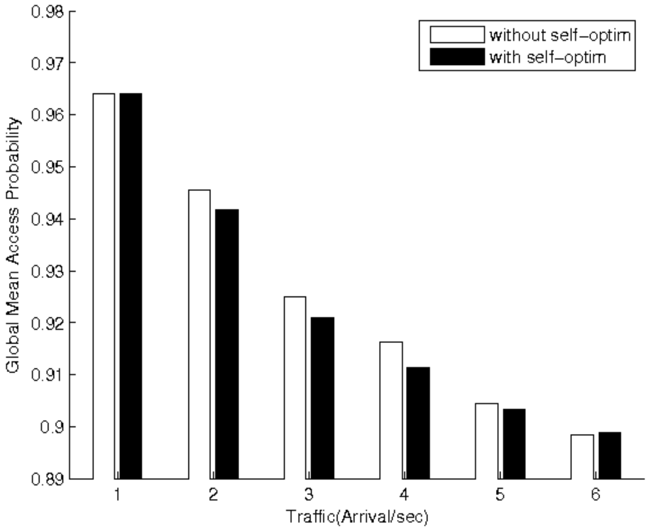

Fig. 3. Mean Access Probability as a function of the traffic intensity for auto-tuned α parameter compared with fixed $\alpha = 1$.

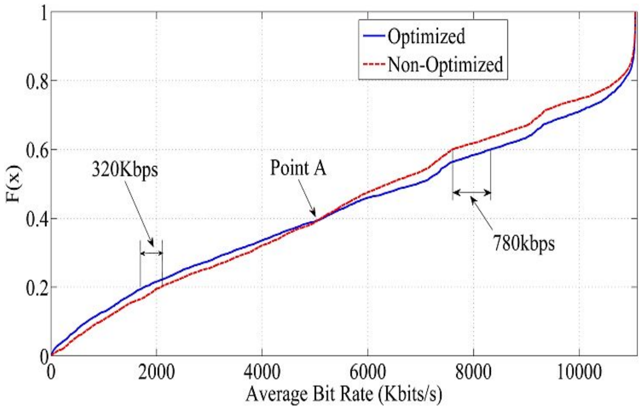

Fig. 4. CDF of the Average Bit Rate for traffic arrival rate of 1 arrival/sec.

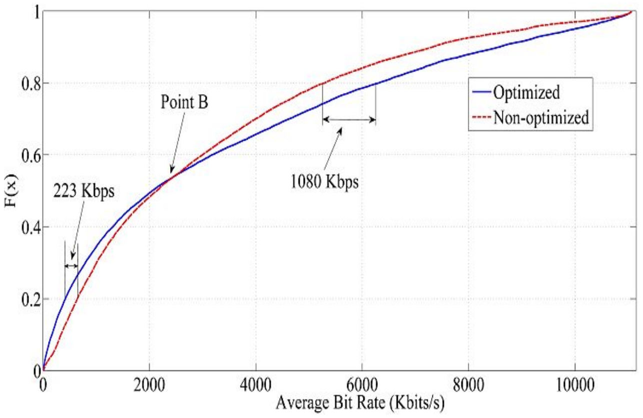

Fig. 5. CDF of the Average Bit Rate for traffic arrival rate = 3 arrivals/sec.

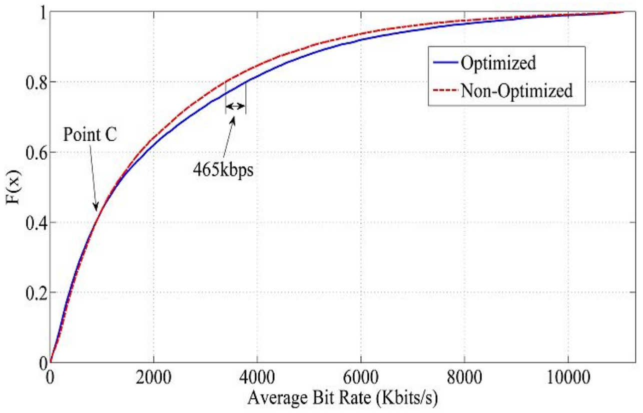

Fig. 6. CDF of the Average Bit Rate for traffic arrival rate = 5 Arrivals/sec.

However, the global mean AP is still above the threshold of 90 %. As a result, the FQL controller decreases the value of α for the desired eNBs, to increase throughput till global mean AP threshold is not violated. The low SINR users are not as much penalized as in the case for traffic of 1 arrival/sec.

Hence, for $\left|F(x)\right|$ = 0.2, ABR of the optimized curve is 223 Kbps less than the non-optimized curve. However, for $\left|F(x)\right|$ = 0.8, the ABR of the optimized curve is 1080 Kbps more than the non-optimized curve. For traffic value of 5 arrivals/sec, it can be observed that only a marginal improvement in ABR as α scheduler tends to be even more fair so that mean AP does not fall below 90 %. Whereas, the low SINR users are not penalized.

VI. Conclusions

In this paper, we have tackled the problem of coverage/capacity optimization in Self Organizing Networks. The optimal resource sharing between the mobile users has been achieved to maximize network throughput, provided the minimum coverage constraint is not violated. FQL is the optimization technique used to achieve the optimization objective. FQL is a model-less optimization technique, well suited for wireless networks with sporadic changes in mobile positions and propagation conditions etc. In the performed case study, it has been observed that the improvement in terms of mean ABR are in the order of magnitude of 10 % while network access probability does not fall below the threshold of 90 %. The case study illustrates the potential benefit of the proposed approach in the real operating networks.

References

[1] T. Kato, “Next-generation mobile network”, *FUJITSU Scientific and Technical Journal*, vol. 48, no. 1, pp. 11–16, 2012.

[2] 3GPP TS 36.300: Evolved Universal Terrestrial Radio Access (E-UTRA) and Evolved Universal Terrestrial Radio Access Network (E-UTRAN), *Overall description*; Stage 2, version 11.2.

[3] NGMN, Recommendation on SON and O&M Requirements, version 1.1, December 2008.

[4] O. Sallent, J. Perez-Romero, J. Sanchez-Gonzalez, R. Agusti, M. A. Diaz-Guerra, D. Henche, D. Paul, “A Roadmap from UMTS Optimization to LTE Self-Optimization”, *IEEE Communications Magazine*, vol. 49, no. 6, pp. 172–182, 2011. [Online]. Available: http://dx.doi.org/10.1109/MCOM.2011.5784003

[5] H. Hu, J. Zhang, X. Zheng, Y. Yang, P. Wu, “Self-configuration and self-optimization for LTE networks”, *IEEE Communication Magazine*, vol. 48, no. 2, pp. 94–100, 2010. [Online]. Available: http://dx.doi.org/10.1109/MCOM.2010.5402670

[6] P. Magnusson, J. Oom, “An Architecture for self-tuning cellular systems”, in *Proc. of the IEEE/IFIP Inter. Symp. on Integrated Network Management*, pp. 231–245, 2001.

[7] Hoglund, K. Valkealahti, “Automated optimization of key WCDMA parameters”, *Wireless Communications and Mobile Computing*, vol. 5, pp. 257–271, 2005. [Online]. Available: http://dx.doi.org/10.1002/wcm.212

[8] Z. Altman *et al.*, *Understanding UMTS Radio network modelling, planning and automated optimization: theory and practice*. Wiley, 2006, ch. 16.

[9] R. Nasri, A. Samhat, Z. Altman, “A new approach of UMTS-WLAN load balancing; algorithm and its dynamic optimization”, in *1st IEEE WoWMoM Workshop on Autonomic Wireless Access (IWAS 2007)*, Helsinki, Finland, 2007.

[10] P. P. Hasselbach, A. Klein, I. Gaspard, “Dynamic resource assignment (DRA) with minimum outage in cellular mobile radio networks”, *Vehicular Technology Conf.*, 2008.

[11] B. Ghimire, G. Auer, H. Haas, “Busy burst for trading-off throughput and fairness in cellular OFDMA-TDD”, *EURASIP J. Wireless Communications and Networking*, vol. 10, 2009.

[12] M. Dirani, Z. Altman, “A cooperative reinforcement learning approach for inter-cell interference coordination in OFDMA cellular networks”, in *Proc. of the 8th Int. Symposium on Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks (WiOpt)*, Avignion, France, 2010.

[13] R. Nasri, Z. Altman, “Handover adaptation for dynamic load balancing in 3GPP long term evolution systems”, in *5th Int. Conf. on Advanced in Mobile Computing & Multimedia (MoMM 2007)*, Jakarta, Indonesia, 2007.

[14] A. Lobinger, S. Stefanski, T. Jansen, I. Balan, “Load Balancing in Downlink LTE Self Optimizing Networks”, in *Proc. of IEEE 71st Vehicular Technology Conf. (VTC 2010)*, 2010.

[15] P. Munoz, R. Barco, I. de la Bandera, M. Toril, S. Luna, “Optimization of a Fuzzy Logic Controller for Handover-based Load Balancing”, *Int. Workshop on Self-Organzsing Networks (IWSON), IEEE Vehicular Technology Conf. (VTC)*, Budapest, Hungary, 2011.

[16] J. Rodriguez, I. de la Bandera, P. Munoz, R. Barco, “Load Balancing in a Realistic Urban Scenario for LTE Networks”, in *IEEE Vehicular Technology Conference (VTC Spring)*, Budapest, Hungary, 2011.

[17] M. Naseer ul Islam, A. Mitschele-Thiel, “Reinforcement learning strategies for self-organized coverage and capacity optimization”, *IEEE Wireless Communications and Networking Conf. (WCNC)*, 2012, pp. 2818–2823.

[18] R. Razavi, S. Klein, H. Claussen, “Self-optimization of capacity and coverage in LTE networks using a fuzzy reinforcement learning approach”, *IEEE 21st Int. Symposium on Personal Indoor and Mobile Radio Communications (PIMRC)*, 2010, pp. 1865–1870, [Online]. Available: http://dx.doi.org/10.1109/PIMRC.2010.5671622

[19] J. Li, J. Zeng, X. Su, W. Luo, J. Wang, “Self-optimization of coverage and capacity in LTE networks based on central control and decentralized fuzzy Q-learning”, *Int. Journal of Distributed Sensor Networks*, 2012, [Online]. Available: http://dx.doi.org/10.1155/2012/878595

[20] J. Mo, J. Warland, “Fair end-to-end window based congestion control”, in *IEEE Trans. networking*, vol. 8, pp. 556–566, 2000.

[21] M.I. Tiwana, “Enhancemant of the Statistical Learning Automated Healing (SLAH) technique using packet scheduling”, *Int. Conf. Emerging Technologies (ICET)*, 2012, pp. 1–5. [Online]. Available: http://dx.doi.org/10.1109/ICET.2012.6375437

[22] H. Kushner, P. Whiting, “Convergence of proportional-fair sharing algorithms under general conditions”, in *IEEE Trans. on wireless communications*, vol. 3, pp. 1250–1259, 2004.

[23] R. S. Sutton, A. G. Barto, *Reinforcement Learning: An Introduction*. MIT Press, 1998.

[24] L. Jouffe, “Fuzzy Inference System Learning by reinforcement Methods”, *IEEE Trans. on Systems, Man, and Cybernetics*, 1998, vol. 28, pp. 338–355. [Online]. Available: http://dx.doi.org/10.1109/5326.704563

[25] R. Nasri, Z. Altman, H. Dubreil, “Fuzzy Q-Learning based autonomic management of macrodiversity algorithm”, *UMTS networks, Annals of Telecommunications*, pp. 1119–1135, 2006.