Online ISSN 1848-3380, Print ISSN 0005-1144
ATKAFF 55(4), 504–513(2014)

Moazzam Islam Tiwana

Self Organizing Networks: A Reinforcement Learning approach for self-optimization of LTE Mobility parameters

DOI 10.7305/automatika.2014.12.502
UDK 621.395.345:004.738.5; 004.716
IFAC 2.5; 5.8

Original scientific paper

With the evolution of broadband mobile networks towards LTE and beyond, the support for the Internet and Internet based services is growing. Self Organizing Network (SON) functionalities intend to optimize the network performance for the improved user experience while at the same time reducing the network operational cost. This paper proposes a Reinforcement Learning (RL) based framework to improve throughput of the mobile users. The problem of spectral efficiency maximization is modeled as co-operative Multi-Agent control problem between the neighbouring eNodeBs (eNBs). Each eNB has an associated agent that dynamically changes the outgoing Handover Margin (HM) to its neighbouring cells. The agent uses the RL technique of Fuzzy Q-Learning (FQL) to learn the optimal mobility parameter i.e., HM value. The learning framework is designed to operate in an environment with the variations in traffic, user positions and propagation conditions. Simulation results have shown the proposed approach improves the network capacity and user experiences in terms of throughput.

Key words: Handover Margin, LTE, Reinforcement Learning, Fuzzy Q-Learning, SON

Samoorganizirajuće mreže: Podržano učenje za optimizaciju LTE mobilnosti. Razvoj širokopojasne mobilne mreže prema LTE mrežama uvjetuje pojačani rast internetskih servisa i usluga. Samoorganizirajuće mreže namijenjene su optimizaciji performansi mreže s ciljem poboljšanja korisnikovog zadovoljstva i smanjenja troškova rada. U radu se predlaže pristup zasnovan na podržanom učenju kako bi se popravila propusnost mobilnog korisnika. Problem maksimizacije spektralne učinkovitosti modelira se kao kooperativni više agentski problem upravljanje između susjednih čvorova (eNBs). Svaki čvor ima pridruženog agenta koji dinamički mijenja marginu primopredaje prema susjednim ćelijama. Agent koristi tehniku neizrazitog Q učenja (FQL) kako bi naučio optimizirati parametre mreže. Učenje je organizirano za rad u uvjetima raznovrsnog prometa, korisničkih položaja i uvjeta propagacije. Simulacijski rezultati pokazuju kako predloženi pristup poboljšava kapacitet mreže i korisnički doživljaj u smislu propusnosti mreže.

Ključne riječi: margina primopredaje, LTE, podržano učenje, neizrazito Q-učenje, samoorganizirajuće mreže

1 INTRODUCTION

The development and advancement in the wireless networks has posed a major challenge to the design and standardization of the Next Generation Mobile Networks (NGMN) [1] [2]. In this context, the goals are to reduce both the complexity of management tasks and the operational cost. While at the same time, the objective is to the maximize the spectral efficiency. The Self-Organizing Networks (SON) have been introduced as one of the solutions in NGMNs [3] to achieve these ambitious targets. SON entities will operate in an environment with varying traffic, changing propagation conditions, newly introduced services, and evolving management policies of the operator.

SON is the term commonly used for autonomic functions in Radio Access Networks (RANs). These functions encompass self-x functionalities such as self-configuration, self-optimization, self-diagnosis and self-healing. SON functionalities have received a particular attention in the standardization of LTE [4] [5]. Different SON mechanisms have already been identified by various actors in the field. In this regard, the use cases studied show considerable enhancement in the network performance.
The idea of self-optimization in Radio Access Networks (RANs) has attracted industry and academia since the last decade [6] [7]. As, it serves as a means to enhance network performance, profitability and to simplify operations. In LTE [4], self-optimization is defined as: a process that uses User Equipment (UE) and eNB performance measures to tune the system parameters in order to achieve optimal per-

formance. The work in the field of self-optimization focuses on dynamically optimizing Radio Resource Management (RRM) parameters such as resource allocation, mobility and traffic balancing. One of the first important contributions in this regard is the dynamic load balancing in hierarchical GSM networks [8]. The concept was validated in a field trial. The Quality of Service (QoS) and capacity of the network were enhanced by dynamically controlling traffic flux from the macro-cells towards the lower layer micro-cells.

Many contributions on self-optimization functionalities in UMTS networks are present in the literature (see for example [9] [10] for review on applications and methodology). Research for the self-optimization using load balancing in UMTS has been reported [11]. 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 [12]. With the advent of LTE the focus of research shifted to the self-optimization of LTE. SON functionalities were included in the standardization of LTE [4], [5]. LTE self-optimizing applications focus mainly on resource and bandwidth allocation [13], Inter-Cell Interference Coordination [14] [15] and load balancing [16] [17].

The study of auto-tuning/self-optimization of mobility parameters has been identified as a relevant case study of self-configuration. It has been shown in [12], [18], [19] and [20] that Fuzzy Logic Controller (FLC) rules, that are optimized using Q-learning (QL), can be 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 using some previous experiences. However, in case 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 to achieve performance optimization by dynamic load balancing between UMTS/WLAN networks [12]. FQL has also been used for the optimization of the Handover Margin (HM) between cells of GSM Edge Radio Access Network (GERAN) [18]. The HM is the main parameter that governs the handover algorithm between two eNBs. All the above mentioned references prove that the FQL is particulary useful for dynamically changing network conditions and configurations. Especially in the case, when we do not have a priori knowledge about the behaviour of network Key Performance Indicators (KPIs). In [19] the concept of load-balancing using FQL is extended to LTE networks. However, the paper lacks some important details like how FQL algorithm is applied (its various problem specific components e.g., the reward/utility function e.t.c.). Furthermore, the mentioned paper also lacks the performance analysis of proposed scheme in terms of some important KPIs like Average Bit Rate (ABR) and File Tranfer Time (FTT). Especially, when the traffic conditions are varied.

This paper investigates the use of FQL for the optimization of the mobility/handover parameter between the neighbouring LTE eNBs. In our case, based on spectral efficiency of eNBs, the auto-tuning is performed by dynamically adapting the HMs. The eNBs co-operate with one another during learning process to speed up convergence to an optimal policy. Each eNB implements learned optimal policy independently to increase the scalability. The performance analysis of the proposed self-optimisation scheme for KPIs like Access Probability (1-Blocking Probability), ABR and FTT, has been made for different traffic values.

The paper is organized as follows: Section 2 presents LTE mobility/handover model used in our case study. Section 3 presents the intercell interference model. Section 4 describes the Multi-Agent RL based framework. Section 5 details the FQL algorithm along with its various components to solve the Multi-Agent RL problem. Section 6 describes the simulation environment and provides the numerical results of the proposed scheme. Section 7 concludes the paper.

2 SYSTEM MODEL FOR MOBILITY

We now consider user mobility between neighbouring cells. The 3G LTE mobility parameter considered in our study is HM. HM refers to the minimum difference in power between the neighbouring cell and the current one, necessary for the mobile to make the handover.

The LTE standard has adopted hard handover wherein a mobile terminal will not be simultaneously connected to the current cell and the new cell [4]. Hard handover is implemented here using a similar algorithm to the one used in GSM systems. This handover is based on the comparison of the received signal strength from the serving cell and from the neighbouring cells. The corresponding algorithm is given below.

The handover algorithm:

While in communication, the mobile periodically measures the received power from its serving eNB and from the neighbouring eNBs. The mobile, initially connected to eNB_i, triggers a handover to eNB_j if the following conditions are satisfied:

1. The Power Budget Quantity (PBQ) is higher than the HM: a mobile connected to eNB_i triggers a handover to an adjacent eNB_j if:

$$PBQ = P_j - P_i \geq HM_{ij} + Hysteresis, \quad (1)$$

where P_j is the received power from eNB_j expressed in dB and HM_{ij} is the outgoing HM of eNB_i towards eNB_j. Hysteresis is a constant independent of the eNBs and of the mobile stations, and is fixed here to 0.

2. The power being received from the target cell must be higher than a given threshold.

3. There are sufficient number of resources/chunks in the target cell. Otherwise, the mobile is bounced back to the original cell.

3 INTERFERENCE MODEL

In this section, we present the downlink intercell interference model used in the simulation of downlink FTP traffic. LTE system allows all the eNBs to use the same frequency band. Furthermore, LTE has high spectral efficiency as it uses OFDMA (Orthogonal Frequency Division Multiple Access) as the access technology on the air interface. OFDMA subdivides the bandwidth into many subcarriers [4]. The bandwidth allocated to a user is in the form of Physical Resource Blocks (PRBs). Each PRB is exclusively assigned to a single user at a time. Hence, the intracell-interference is eliminated. The interference suffered by an UE is the intercell interference, modelled as follows:

Consider an UE u is attached to an eNB e using a PRB. The average interference I_{eu} suffered by u per sub-carrier is given as

$$I_{eu} = \sum_{f \neq e} M(e,f) \nu_f \frac{P_f G_f}{\xi_{fu}}, \quad (2)$$

where $M(e,f)$ denotes the interference matrix element, it is 1 if eNB e and eNB f use the same frequency band, else it is 0. ν_f is the load on the eNB f i.e., ratio of the number of allocated PRBs to the total number of PRBs in eNB f. P_f is the downlink transmit power of eNB f while G_f indicates its antenna gain. ξ_{fu} is the path loss between the eNB f and UE u.

The signal to noise ratio received at UE u, denoted as $SINR_{eu}$, is given as:

$$SINR_{eu} = \frac{P_e G_e}{\xi_{eu}\left(I_{eu} + \sigma^2\right)}, \quad (3)$$

where σ is the thermal noise per subcarrier.

4 QL FOR SELF-OPTIMIZATION IN LTE

The LTE network has been modeled as a Multi-Agent RL system such that each eNB has an associated agent (see [21] for more details on RL). The agents cooperate with one another during the learning or exploration phase to speed up convergence to an optimal policy. However, during the exploitation phase each multiagent implements the learned policy independently. Hence, this increases the scalability of the proposed auto-tuning/self-optimization scheme.

The inherent dynamics of a mobile network follow a transitionary model due to the phenomenon like mobility of User Equipments (UEs), fading phenomenon, changing traffic distribution and interference etc. The learning process is a Markov Decision Process (MDP) [22]. The agents interact in real time with the environment and at the same time exploit the experience of the past. In response to the changing system states, each agent should select those actions tried in the past which produced a lot of reward. However, the agent learns about such actions by selecting the actions that have not been tried before. Thus, this leads to the exploration/exploitation trade-off. The agent uses its current knowledge to maximize the reward. On the other hand, it has to explore actions that yield maximum reward in long term.

Q-Learning (QL) is a special type of Reinforcement Learning (RL) that can solve optimization problems when the system model is not available as a closed-form expression. Instead, it relies on the Temporal Difference (TD) method to incrementally solve the learning problem [21]. The objective of an agent in QL is to select those actions that maximize the received long term reward, given as:

$$R^k = r^k + \gamma r^{k+1} + \gamma^2 r^{k+2} + \gamma^3 r^{k+3} ..... = \sum_{t=0}^{\infty} \gamma^t r^{t+k}, \quad (4)$$

where k-th time instant corresponds to the initial state of the agent. r^{t+k} denotes the numerical instantaneous reward obtained as a consequence of taking an action at the time step $t+k$. γ is the discount factor. For γ close to 0 the controller tries to optimize immediate rewards whereas for γ close to 1 the controller considers future rewards almost as important as immediate ones. In the present work γ is set to 0.95 [15].

Fundamental to the QL is the estimation of value functions which are a function of state-action pairs [21]. These functions are an estimate of "how good" it is for an agent to take an action in a given state. This notion of "goodness" is estimated in terms of future expected rewards. Of course, these rewards also depend upon the actions executed by the agent in the subsequent/future states when it follows a policy π. π maps the perceived states to the corresponding actions to be executed in those states. The value function is given as [23]:

$$\begin{aligned} V_\pi(s^k) &= E_\pi\left[r^k + \gamma r^{k+1} + \gamma^2 r^{k+2} .....\right] \\ &= E_\pi\left[\sum_{t=0}^{\infty} \gamma^t r^{t+k}\right], \end{aligned} \quad (5)$$

where s^k is the initial state of the agent. Now assuming $s^k = s^0 = s$, the (5) becomes:

$$V_\pi(s) = E_\pi\left[\sum_{t=0}^{\infty}\gamma^t r^t \mid s^0 = s\right]. \quad (6)$$

The optimal policy π^* is defined as a policy that maximizes $V_\pi(s)$, given as:

$$\pi^* = \underset{\pi}{\text{argmax}}\ V_\pi(s). \quad (7)$$

$V_{\pi^*}(s)$ is the maximum cumulative discounted reward that an agent receives while starting from state s and following π^*.

Let $r^t(s^t, b^t)$ denote the immediate reward an agent receives when it executes an action $b^t \in B$ in the state s^t and transits to the state s^{t+1}. It is difficult for an agent to learn π^* directly in a real-time environment, as the rewards for different state-action pairs are not available offline. Rather, the sequence of immediate rewards $r^t(s^t, b^t)$ for $t = 0, 1, 2, \ldots$ are calculated at each time instant. This problem is solved by defining the quality function $Q(s, b)$ as the immediate reward received when an action b is performed in state s, plus the value function (discounted by γ) of following the optimal policy thereafter [23]:

$$Q(s, b) = E\left[r(s, b) + \gamma V^{\pi^*}(\delta(s, b))\right], \quad (8)$$

where $\delta(s, b)$ represents the next state when an action b is applied to the state s. Here, we have used the expectation operator because for the dynamics of a mobile network, the $r(s, b)$ and the $\delta(s, b)$ have probabilistic rather than deterministic outcomes. We can rewrite the optimal policy π^*:

$$\pi^*(s) = \underset{b}{\text{argmax}}\, Q(s, b). \quad (9)$$

This rewrite in terms of $Q(s, b)$ signifies that the agent reacts to the local values of Q for the current state to choose the globally optimal action sequences. Furthermore, the agent neither needs to perform a full look ahead search nor explicitly know a priori the resulting state as a consequence of taking an action in a given state.

Comparing (7) and (9), the solution to maximization problem involves estimating the quality function $Q_\pi(s, b)$, which is given as:

$$Q_\pi(s, b) = E_\pi\left[\sum_{t=0}^{\infty}\gamma^t r^t(s^t, b^t) | s^o = s, b^o = b\right]. \quad (10)$$

Equation (10) can be estimated iteratively using (8).

Now, the definition of Q-learning has been generalized to include the model-free/non-deterministic environment. Hence, a new training rule is required that should converge even if each time we get a different value of $r(s, b)$ for a given state-action pair. This is achieved by modifying the training rule so that we take the decaying weighted average of the current and the revised quality estimate. Denote by $Q^{t+1}(s^t, b^t)$ as the learner estimate of the Q function at time $t + 1$, calculated using the learning rule as [24]:

$$\begin{aligned} Q^{t+1}(s^t, b^t) \quad &= \quad (1-\kappa)Q^t(s^t, b^t) + \kappa[r^t + \\ & \gamma(\max_{\acute{b}} Q^t(s^{t+1}, \acute{b}))], \end{aligned} \quad (11)$$

where κ denotes the learning rate having value within the interval $[0, 1]$. (11) can be rewritten as:

$$\begin{aligned} Q^{t+1}(s^t, b^t) \quad &= Q^t(s^t, b^t) + \kappa[r^t + \\ & \gamma(\max_{\acute{b}} Q^t(s^{t+1}, \acute{b})) - Q^t(s^t, b^t)] \end{aligned} (12)$$

5 FUZZY Q-LEARNING

QL algorithm is used to map the discrete states to the actions. However, our problem presents the case of continuous state and action spaces, leading to enormous complexity. This problem is solved by using Fuzzy Logic to discretize the state and action spaces. Fuzzy Q-Learning (FQL) combines fuzzy logic with Q-Learning to form a Fuzzy Inference System (FIS) [25] [26] as shown in Figure 1.

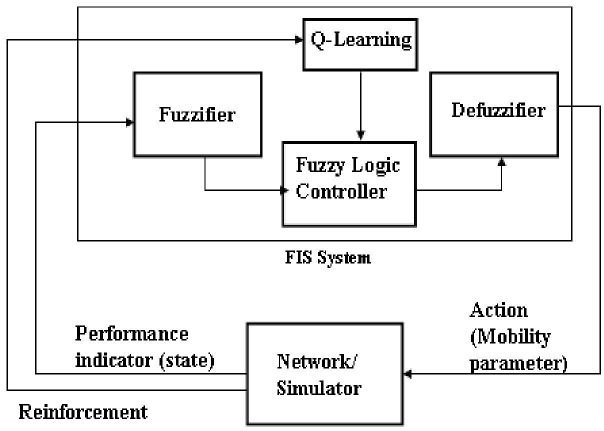

Fig. 1. Architecture of self-optimization procedure.

The input to the FIS is in the form of a state vector s. The first element of the FIS is the fuzzifier that maps each continuous crisp (continuous) element of s into one or two fuzzy sets. This transforms the continuous variables into finite number of membership functions. This process is called *fuzzification*. The Fuzzy Logic Controller (FLC) [15] [27] uses these membership functions to calculate the output action for each triggered rules. These output actions are then again mapped back to continuous domain output through the *defuzzification* process.

6 COMPONENTS OF FQL RL SYSTEM

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

6.1 State

The proposed state vector, corresponding to an eNB c, that is input to the FQL controller, is defined as follows:

$$s_c = \begin{bmatrix} HM_{out} & SE_c & SE_{NC} \end{bmatrix}$$

Where HM_{out} is the outgoing HM value from an eNB c to all its neighbouring eNBs NC. SE_c is the mean spectral efficiency of eNB c. While SE_{NC} denotes the mean aggregated spectral efficiency of NC, calculated as below

$$SE_{NC} = \sum_{i \in NC} \phi_{ic} SE_i, \tag{13}$$

where ϕ_{ic} indicates the normalized traffic flux from $i \in NC$ to c and is a measure of degree of neighbourhood of cell i with cell c. The weights ϕ_{ic} satisfy the condition $\sum_i \phi_{ic} = 1$.

6.2 Action And Policy

The action of each eNB is to change its HM_{out} (outgoing HM) to all its neighbouring eNBs NC according to 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.

6.3 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:

$$r^t = \sum_{m \in M} \frac{m(th^t)}{M}, \tag{14}$$

where $m(th^t)$ denotes the instantaneous throughput of mobile m.

6.4 FQL Algorithm description

This section presents the FQL algorithm as given in [25]. Let the state vector $s^t = \left[s_1^t, ..., s_j^t, ..., s_J^t\right]$, where j is the j^{th} element of state vector before fuzzification. After fuzzification, the membership function $T(s^t)$ quantifies the degree of membership of an input value s_j^t to a specific fuzzy set corresponding to a fuzzy label. The fuzzy label of s_j^t, denoted as F_j, can be 'LOW', 'MEDIUM' and 'HIGH' as shown in Figure 2. The degree of membership of a given fuzzy state s_j^t to a fuzzy label (or fuzzy set) F_j, is calculated using membership function m_{F_j}. If R denotes all the rules of a FLC, then rule $r \in R$ is given as:

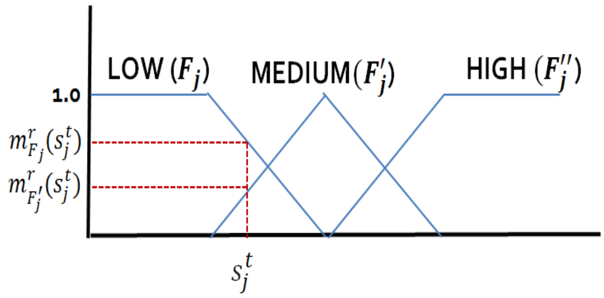

Fig. 2. The three fuzzy sets.

$$\text{IF } \left(s_1^t \text{ is } F_1^r\right) \text{....AND } \left(s_j^t \text{ is } F_j^r\right) \text{....AND } \left(s_J^t \text{ is } F_J^r\right)$$
$$\text{THEN } b = a^r \text{ with } q^t(F^r, a^r) \tag{15}$$

where $F^r = \left[F_1^r, ..., F_j^r, ...F_J^r\right]$ 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 elementary quality $q(F^r, a^r)$ corresponding to fuzzy state F^r and action a^r, is initialized to zero.

The degree of truth T_r for each rule $r \in R$ is given as

$$T_r(s^t) = \prod_{j=1}^{J} m_{F_j^r}(s_j^t). \tag{16}$$

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

$$\left\{ \begin{array}{lll} \forall r \in R : a^r = \underset{l \in L}{\text{argmax}}\, q^t(F^r, a^l) & & \text{with prob. } \epsilon \\ \forall r \in R : a^r = \underset{l \in L}{\text{random}}(a^l) & & \text{with prob. } 1 - \epsilon \end{array} \right\}. \tag{17}$$

, 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^t and the triggered rules in R is given as:

$$a(s^t) = \sum_{r \in R} T_r(s^t) a^r. \tag{18}$$

The associated quality of the inferred action is calculated as the linear interpolation of elementary quality values (q values):

$$Q(s^t, a(s^t)) = \sum_{r \in R} T_r(s^t) q^t(F^r, a^r) \tag{19}$$

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

$$V(s^{t+1}) = \sum_{r \in R} T_r(s^{t+1}) \max_{l \in L} q^t(F^r, a^l). \quad (20)$$

The updation of q value requires that first difference between the Quality value ΔQ of the old and the new state be calculated as:

$$\Delta Q = r^t + \gamma V(s^{t+1}) - Q(s^t, a(s^t)). \quad (21)$$

The q values can now be updated using an iterative procedure similar to the update of Q values as in (12):

$$q^{t+1}(F^r, a^r) = q^t(F^r, a^r) + \kappa T_r(s^t)\Delta Q. \quad (22)$$

The complete FQL algorithm applied to the mobile communication network environment is listed in the table

Initialization:
1. The elementary quality values look up table is initialized as $q(F^r, a^l) = 0 \forall r \in R, l \in L$.
time t=0.
Repeat:
2. *Fuzzification*: the continuous input state vector from the system is fuzzified as discrete state $s^t \in S$.
3. Calculate the degree of truth $T_r(s^t)$ for each rule $r \in R$ using (16).
4. For each rule $r \in R$, calculate the action a^r using the EEP policy in (17).
5. Calculate the inferred action $a(s^t)$ and its associated quality using (18) and (19) respectively
6. Execute the action $a(s^t)$ and the system transits to the state s^{t+1}. The controller receives the reward r^t.
7. Calculate the degree of truth $T_r(s^{t+1})$ for the new state.
8. Use (20) to calculate the value function for the state s^{t+1}
9. Update the elementary quality $q^{t+1}(F^r, a^r)$ for each rule using (22).
10. $t \leftarrow t + 1$.

7 CASE STUDY

7.1 Simulation Scenario

A LTE network comprising of 45 eNBs in a dense urban environment, as shown in Figure 3, has been simulated. A MATLAB LTE simulator described in [16] has been used. We consider downlink transmissions. The simulation parameters are listed in Table 1.

The simulator performs correlated Monte Carlo snapshots with time steps of one second to account for the time evolution of the traffic. The arrival of the new users is simulated as a Poisson process. At the end of each time step

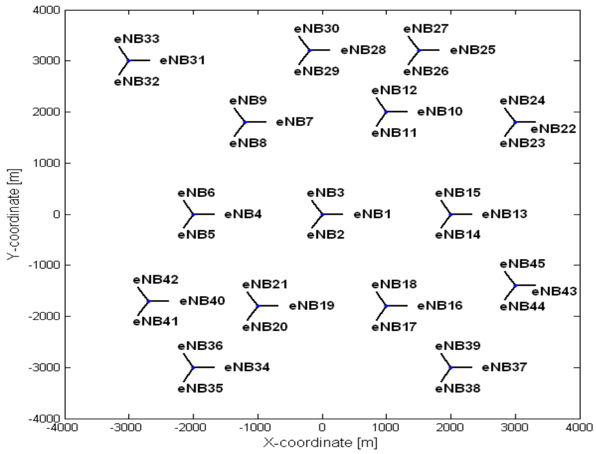

Fig. 3. The network diagram of the simulated system.

of one second, new mobile positions are updated, handover events are processed, new users are admitted according to the conditions of access and some other users leave the network (end their communications or are dropped).

A Call Admission Control (CAC) procedure based upon signal strength has been implemented. A mobile searches and selects the eNB with the highest Reference Signal Received Power (RSRP). The eNB admits the mobile if RSRP is above -104 dBm and at least one PRB is available. Admission control based on minimal bit-rate is not implemented for FTP service. A call is dropped if mobile roams into an area with low coverage i.e., RSRP falls below -104 dBm and as a result, Radio Link Failure occurs. The quality tables, obtained from link level simulations, are used to calculate mobile bit rate from the SINR. The mobile SINR and hence the bit rate is updated after every simulation time step.

The simulator has two modes of operation viz., static mode and dynamic mode. In static mode, no self-optimization is taking place. The simulations are run for 5000 time steps, with a fixed HM value, and the KPIs are averaged between the interval of 500 to 5000. The first 499 seconds are excluded to account for the initial transient effects. In the dynamic mode or self-optimization mode the FQL algorithm adapts the HM of the cells with a periodicity of 50 seconds. Here, the KPIs are also averaged with the same periodicity of 50 seconds. The learning rate is set to value of $\kappa = 0.1$ [15]. The HM value can vary from 1 to 12. The simulations are done over a period of 150000 seconds. HM_{out} is fuzzified using the membership functions given in Figure 4. While both SE_c and SE_{NC} are fuzzified using the membership functions given in Figure 5.

By definition, the Access Probability (AP) of a system

Table 1. The system level simulation parameters.

Parameters	Settings
System bandwidth	5 MHz
Cell layout	45 eNBs, single sector
Maximum eNB transmit power	32 dBm
Inter-site distance	1.5 to 2 km
Subcarrier spacing	15 kHz
PRBs per eNB	15
Path loss	$L = 128.1 + 37.6 \log_{10}(R)$, R in kilometers
Thermal noise density	-173 dBm/Hz
Shadowing standard deviation	6 dB
Traffic model	FTP
File size	5700 Kbits
PRBs assigned per mobile	1 to 4 (First-come, first-serve basis)
Mobility of mobiles	90%
Mobile speed	15 m/s
HM_{max}	12 dB

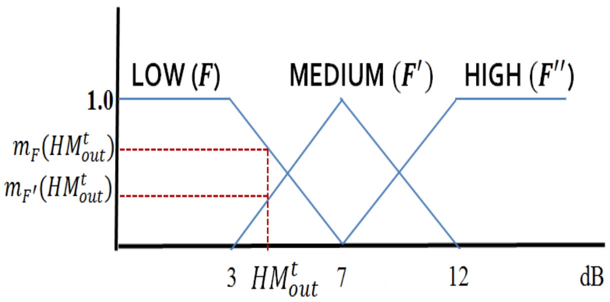

Fig. 4. The three fuzzy sets and corresponding membership functions for HM_{out}.

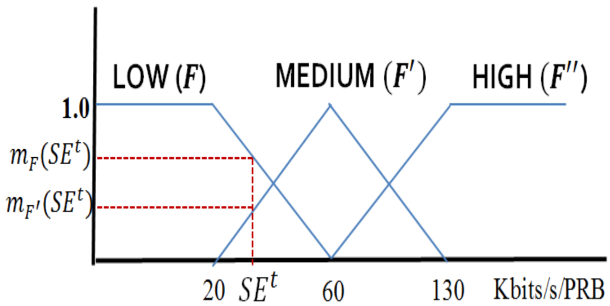

Fig. 5. The three fuzzy sets and corresponding membership functions for Spectral Efficiency (SE).

is given as:

$$AP = \frac{N_{accepted}}{N_{accepted} + N_{blocked}}, \qquad (23)$$

where $N_{accepted}$ and $N_{blocked}$ are the counters for the number of calls accepted and the number of calls blocked by CAC of the system, respectively.

Similarly, the instantaneous Average Bit Rate (ABR) of a system is defined as:

$$ABR = \sum_{m=0}^{M} \frac{m(th^t)}{M} \qquad (24)$$

where $m(th^t)$ denotes the instantaneous throughput/bit rate of mobile m.

The average File Transfer Time (FTT) of overall mobiles accepted in the network is defined as:

$$FTT = \sum_{v=0}^{N_{accepted}} \frac{v(FTT)}{N_{accepted}}, \qquad (25)$$

where $v(FTT)$ is the time taken by the vth accepted mobile to download the 5700 Kbits file.

Reference Solution:

An optimal default value for HM is chosen as 6 dB for all eNBs in the network and will serve as the reference (default) solution. This reference solution will be used as a starting point for the self-optimization process. The default HM value is determined in [28].

7.2 Simulation Results

The results obtained for the HM adaptation using the FQL approach have been compared with the reference system where HM value is fixed to the reference value of 6dB.

Figure 6 depicts the access probability of the two systems. The access probability is an indicator of the capacity. It indicates the traffic intensity that can be served by the network for a given access probability. The self-optimization of HM gives a better network capacity as compared to the system without any such mechanism. Initially, for the low traffic value of 1 arrival per second we do not see any improvement in the access probability as all the mobile gets accepted. However, with the increase in traffic the advantage of the self-optimization mechanism becomes clear and for the traffic value of 6 arrivals per second we see that access probability improves by 2.1%

Figure 7 compares the mean File Transfer Time (FTT) of mobiles of the two systems. Significant improvement with respect to the no self-optimization case are observed. For the traffic values of 4, 5 and 6 arrivals per second the gains of upto 9% reduction in the mean FTT are observed.

Figure 8 shows the gain brought about by the self-optimization in the mean Average Bit Rate (ABR) of the

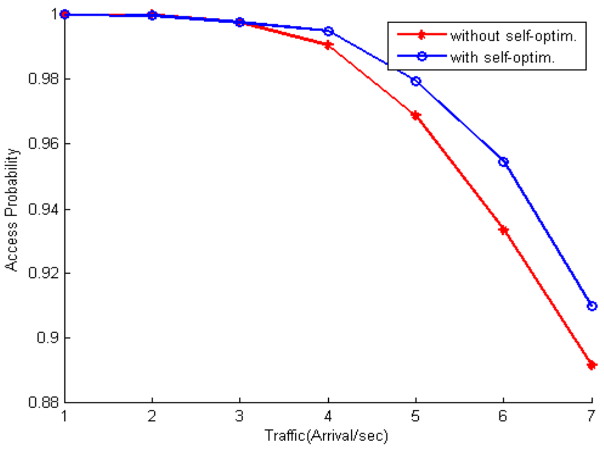

Fig. 6. Access probability as a function of the traffic intensity for auto-tuned handover compared with fixed handover margin (6dB).

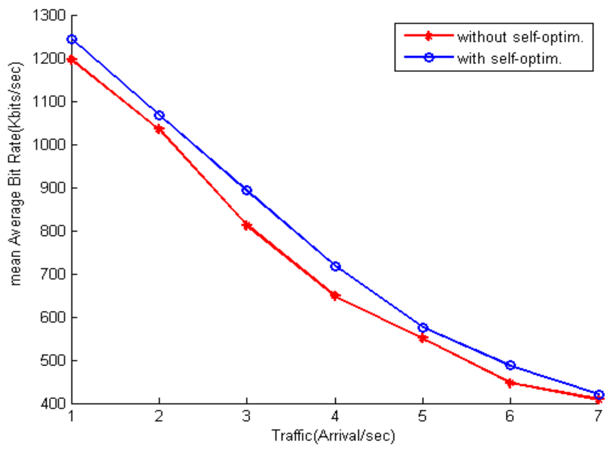

Fig. 8. mean Average Bit Rate as a function of the traffic intensity for auto-tuned handover compared with fixed handover margin (6dB).

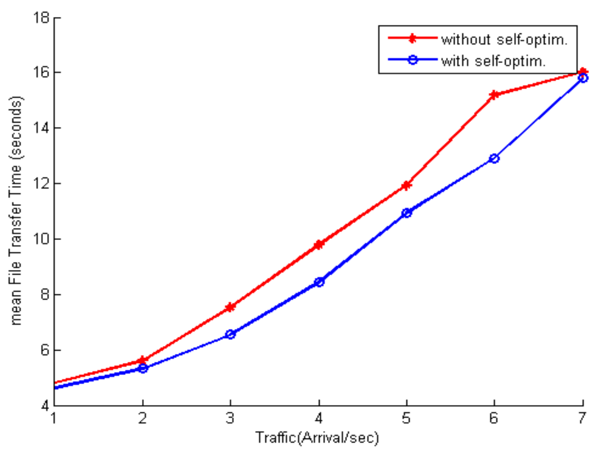

Fig. 7. mean File Transfer Time as a function of the traffic intensity for auto-tuned handover compared with fixed handover margin (6dB).

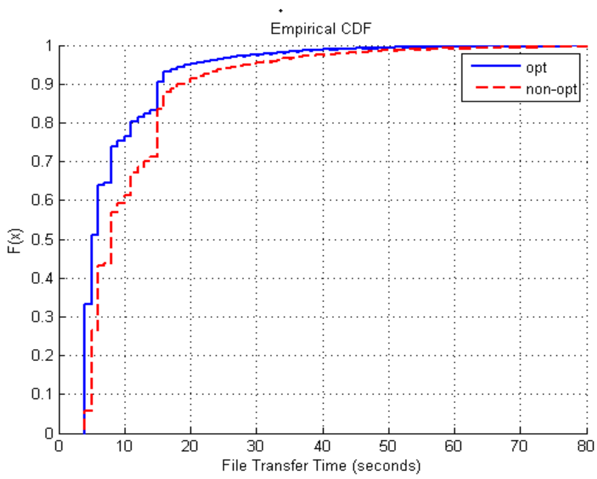

Fig. 9. CDF of the File Transfer Time.

mobiles in the network. It is observed that for the traffic value of 4 we can have a maximum increase in the mean ABR of 10%.

The Cumulative Distribution Function (CDF) of File Transfer Time (FTT) of mobiles are compared for the two systems in Figure 9. The HM FQL solution results in lower values of the FTT. This is evidently due to the fact our objective function of FQL tends to maximize throughput.

8 CONCLUSION AND DISCUSSION

This paper has presented a distributed solution for self-optimization of mobility parameters in LTE networks using FQL. FQL learning is a model-less optimization technique, suited for the wireless networks with the sporadic changes in mobile positions and propagation conditions e.t.c. In the learning/exploration the agents co-operate to learn the optimal action policy for the optimization of FLC rules. During the exploitation phase each agent independently uses the learned FLC rules to dynamically adapt the HM. In the case study done, it is observed that the improvement in terms of FTT and ABR are in the order of magnitude of 10 %. This methodology can easily be extended for the self-optimization of other RRM parameters related to e.g., scheduling and cell selection/re-selection.

REFERENCES

[1] T. Kato (2012), "Next-Generation Mobile Network," *FUJITSU Scientific and Technical Journal*, 2012: 48(1); 11-16.

[2] H. -C. Chao, S. Zeadally, Y.-S. Chen, G. Martinez, and R.-C. Wang (2010), "Next Generation Networks (NGNs)," *International Journal of Communication Systems*, 2010: 23: 691-693. doi: 10.1002/dac.1144.

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

[4] 3GPP TS 36.300 (2012): Evolved Universal Terrestrial Radio Access (E-UTRA) and Evolved Universal Terrestrial Radio Access Network (E-UTRAN), *Overall description*; Stage 2, version 11.2.

[5] 3GPP TR 36.902 (2011), Evolved Universal Terrestrial Radio Access Network (E-UTRAN); *Self configuration and self-optimization network use cases and solutions*, version 9.3.1.

[6] O. Sallent, J. Pérez-Romero, J. Sánchez-González, R. Agustí, M.A. Díaz-Guerra, D. Henche, D, Paul (2011), "A Roadmap from UMTS Optimization to LTE Self-Optimization," *IEEE Communications Magazine*, 2011: 49(6); 172-182.

[7] H. Hu, J. Zhang, X. Zheng, Y. Yang, P. Wu (2010), "Self-configuration and self-optimization for LTE networks," *IEEE Communication Magazine*, 2010: 48(2); 94-100

[8] P. Magnusson, J. Oom (2001), "An Architecture for self-tuning cellular systems," *Proc. of the 2001 IEEE/IFIP Inter. Symp. on Integrated Network Management*, 2001: 231-245.

[9] A. Höglund and K. Valkealahti (2005), "Automated optimization of key WCDMA parameters," *Wireless Communications and Mobile Computing*, 2005: 5; 257-271, doi: 10.1002/wcm.212.

[10] Z. Altman at al. (2006), "Auto-tuning of RRM parameters in UMTS networks," chapter 16 in book *Understanding UMTS Radio network modelling, planning and automated optimization: theory and practice*, Editors: M. Nawrocki, M. Dohler and H. Aghvami, Wiley, 2006.

[11] J. Li, C. Fan, D. Yang, and J. Gu, (2005), "UMTS Soft Handover Algorithm with Adaptive Thresholds for Load Balancing," *in Proc. of IEEE 62nd Vehicular Technology Conference (VTC)*.

[12] R. Nasri, A. Samhat and Z. Altman (2007), "A new approach of UMTS-WLAN load balancing; algorithm and its dynamic optimization," *1st IEEE WoWMoM Workshop on Autonomic Wireless Access 2007 (IWAS07)*, Helsinki, Finland.

[13] P. P. Hasselbach, A. Klein and I. Gaspard (2008), "Dynamic resource assignment (DRA) with minimum outage in cellular mobile radio networks," *Vehicular Technology Conference 2008-Spring*.

[14] B. Ghimire, G. Auer, and H. Haas (2009), "Busy burst for trading-off throughput and fairness in cellular OFDMA-TDD," *EURASIP J. Wireless Communications and Networking*, vol. 2009, article no. 10.

[15] M. Dirani and Z. Altman (2010), "A cooperative reinforcement learning approach for inter-cell interference coordination in ofdma cellular networks," in *Proceedings of the 8th International Symposium on Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks (WiOpt)*, Avignion, France, 2010.

[16] R. Nasri and Z. Altman (2007), "Handover adaptation for dynamic load balancing in 3GPP long term evolution systems," *5th Int. Conf. on Advanced in Mobile Computing & Multimedia (MoMM2007)*, Jakarta, Indonesia, 2007.

[17] A. Lobinger, S. Stefanski, T. Jansen, and I. Balan (2010), "Load Balancing in Downlink LTE Self-Optimizing Networks," *in Proc. of IEEE 71st Vehicular Technology Conference (VTC)*, 2010.

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

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

[20] Naseer ul Islam, M., and Andreas Mitschele-Thiel (2012), "Reinforcement learning strategies for self-organized coverage and capacity optimization," *IEEE Wireless Communications and Networking Conference (WCNC)*, pp. 2818-2823, 2012.

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

[22] Y. Li, H. Ji, Xi. Li, and V. C.M. Leung (2012), "Dynamic channel selection with reinforcement learning for cognitive WLAN over fiber," Int. J. Commun. Syst., 2012: 25: 1077-1090. doi: 10.1002/dac.2338.

[23] T. MITCHELL (1997), *Machine Learning*, McGraw Hill Inc., New York, NY, 1997.

[24] T. Jaakkola, M.I. Jordan, S.P. Singh, "On the Convergence of Stochastic Iterative Dynamic Programming Algorithms," Neural Computation, 6(6):1185-1201, 1994.

[25] L. Jouffe (1998), "Fuzzy Inference System Learning by reinforcement Methods," *IEEE Transaction on Systems, Man, and Cybernetics*, 1998: 28; 338-355.

[26] S. M. Tseng (2007), "A fuzzy PIC multiuser detection/vector channel estimation scheme in Rayleigh fading channels," *International Journal Communication Systems*, 2007: 20: 1285-1298. doi: 10.1002/dac.876.

[27] R. Nasri, Z. Altman and H. Dubreil (2006), "Fuzzy Q-Learning based autonomic management of macrodiversity algorithm in UMTS networks," *Annals of Telecommunications*, 2006: 1119-1135.

[28] M. I. Tiwana, B. Sayrac, Z. Altman, T. Chahed (2009), "Troubleshooting of 3G LTE mobility parameters using iterative statistical model refinement," *IFIP Wireless Days 2009*, Paris.

Moazzam I. Tiwana received a B.Sc. degree in Electrical and Electronics Engineering from the University of Engineering and Technology, Taxilla, Pakistan, in 2001, and a M.Sc. degree in Digital Telecommunication Systems from ENST, Paris, France in 2007 and a Ph.D. degree in Mobile Communications from Telecom Sud-Paris Paris, France, in 2010. His PHD was with the R&D Group of Orange Labs of France Telecom. He has more than seven years of industrial and academic experience with research publications in the reputed international journals.

AUTHORS' ADDRESSES

Asst. Prof. Moazzam I. Tiwana, Ph.D.
Department of Electrical Engineering,
COMSATS Institute of Information Technology,
Islamabad, Pakistan.
email: moazzam_islam@comsats.edu.pk
Tel.: +92-302-5150047

Received: 2013-03-02
Accepted: 2014-03-25