Adaptive Proportional Fair Parameterization Based LTE Scheduling Using Continuous Actor-Critic Reinforcement Learning

Ioan Sorin Comşa, Sijing Zhang, Mehmet Aydin
Institute for Research in Applicable Computing
University of Bedfordshire
Luton, LU1 3JU, United Kingdom
{Ioan.Comsa,Sijing.Zhang,Mehmet.Aydin}@beds.ac.uk

Jianping Chen, Pierre Kuonen, Jean-Frederic Wagen
Institute for Complex Systems
University of Applied Sciences of Western Switzerland
Fribourg, CH-1705, Switzerland
{Jianping.Chen, Pierre.Kuonen, Jean-Frederic.Wagen}@hefr.ch

***Abstract*—Maintaining a desired trade-off performance between system throughput maximization and user fairness satisfaction constitutes a problem that is still far from being solved. In LTE systems, different tradeoff levels can be obtained by using a proper parameterization of the Generalized Proportional Fair (GPF) scheduling rule. Our approach is able to find the best parameterization policy that maximizes the system throughput under different fairness constraints imposed by the scheduler state. The proposed method adapts and refines the policy at each Transmission Time Interval (TTI) by using the Multi-Layer Perceptron Neural Network (MLPNN) as a non-linear function approximation between the continuous scheduler state and the optimal GPF parameter(s). The MLPNN function generalization is trained based on Continuous Actor-Critic Learning Automata Reinforcement Learning (CACLA RL). The double GPF parameterization optimization problem is addressed by using CACLA RL with two continuous actions (CACLA-2). Five reinforcement learning algorithms as simple parameterization techniques are compared against the novel technology. Simulation results indicate that CACLA-2 performs much better than any of other candidates that adjust only one scheduling parameter such as CACLA-1. CACLA-2 outperforms CACLA-1 by reducing the percentage of TTIs when the system is considered unfair. Being able to attenuate the fluctuations of the obtained policy, CACLA-2 achieves enhanced throughput gain when severe changes in the scheduling environment occur, maintaining in the same time the fairness optimality condition.**

Keywords- LTE-A, TTI, CQI, throughput, fairness, scheduling rule, GPF, MLPNN, RL, policy, CACLA-1, CACLA-2.

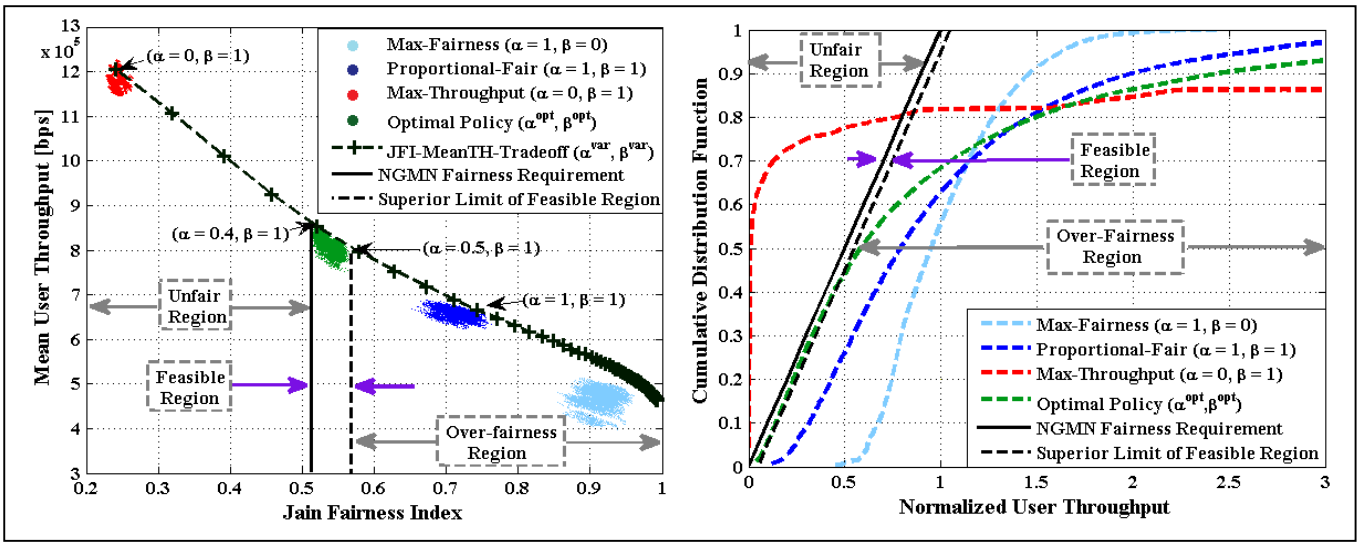
I. INTRODUCTION

The optimal allocation of channel and rate resources under a given set of Quality of Service (QoS) requirements constitutes an important throughput maximization task of the scheduling procedure. In particular, the fairness-guaranteed scheduling becomes a complex problem to solve since multiple active users are connected to the base station through the fast fading radio channels and LTE schedulers are designed in the opportunistic manner intended to exploit the multiuser diversity. Hence, by using simple scheduling rules, near Pareto optimal user throughputs should be obtained under a given fairness performance requirement among multiple users. The fairness target selection and the modalities of applying the best scheduling rules in order to satisfy the considered requirement become the main concerns in designing a self-learning LTE scheduler. The Channel Quality Indicator (CQI) feedback as achievable user rate information should be considered in the fairness performance evaluation metric in order to avoid the unfair treatment of some users with unfavorable channel conditions. For this study, the Next Generation Mobile Networks (NGMN) fairness requirement is considered as a fairness criterion in such a way that *a system is considered fair* if and only if at each TTI t at least (*100-x)% of active users achieve at least x% of each normalized user throughput* [1]. The fairness criterion can be achieved by using a satisfactory parameterization of the GPF scheduling rule [2].The objective function of the current study is designed to maximize the system throughput under the NGMN fairness constraints. With a given input state at each TTI, the scheduler should be able to find the best policy of GPF parameters set to be applied in the current TTI in order to meet the grand NGMN objective.

The continuous and multidimensional scheduler state space is modeled by using the Markov Decision Process (MDP) in which the selected CACLA-2 actions are rewarded based on the transition performance from previous to the current state. Based on given MDP problems, CACLA-2 criticizes each action set in order to localize much faster the optimal scheduler state [3]. The experiments show that CACLA-2 performs much better in comparison with other RL algorithms by maximizing the mean user throughput and minimizing, at the same time, the percentage of TTIs when the system stays unfair. The rest of this paper is organized as follows: Section II promotes the relevant techniques proposed in the literature. Section III presents the optimization problem. Section IV presents the architecture of the novel self-learning scheduler. Section V shows the results, and the paper concludes with Section VI.

II. RELATED WORK

The idea of applying the RL principles for the LTE scheduler state space generalization constrained by multiple QoS objectives is originally proposed in [4], [5]. In particular, the Q-Learning algorithm with the MLPNN function approximation is used to achieve different static tradeoff levels between system throughput and user fairness [6]. The packet scheduling optimization problem in terms of the static Jain Fairness Index (JFI) constraint is analyzed in [7]. By imposing



a) Jain Fairness Index vs. Mean User Throughput

b) CDF distribution

Fig.1. Fairness evaluation criteria (benchmarks) for a 60-user scenario equally distributed from ENodeB base station to the edge of cell under uniform power allocation and FDD downlink transmission with a system bandwidth of 20MHz

the fairness limit regardless of the channel conditions makes the approach impractical for the real time schedulers. For this reason, the qualitative fairness measures based on channel statistics, rather than the quantitative fairness thresholds are preferred to be used in practice. The NGMN qualitative fairness measure adaptation techniques in LTE systems were first elaborated in [8] in which the cumulative distribution function (CDF) of the normalized user throughputs is adjusted by using a simple parameterization of the GPF scheduling metric. The CDF curve adaptation to the fairness requirement is achieved at each 1s dealing with the waste of system capacity, especially when the traffic load varies drastically TTI-by-TTI. A slightly improved method proposed in [7] is introduced in [9], in which the JFI constraint is replaced by the NGMN fairness requirement in the CDF domain (continuous oblique black line from Fig. 1.b.).

A set of RL algorithms that are able to match at each TTI the CDF curve under the NGMN fairness constraint (Fig.1.b) is proposed in [9]. The CACLA-1 actor critic algorithm outperforms any of the methods proposed in [7], [8] and other RL algorithms by maximizing the percentage of TTIs when the system respects the NGMN fairness requirement. It is important to notice that all the proposed methods being illustrated above use a simple parameterization of the GPF scheduling rule. The method proposed in this paper uses the double parameterization of GPF scheduling discipline. It is proved that by using CACLA-2 with two continuous GPF parameters, the obtained policy is able to converge much better to the optimal scheduler state when compared with CACLA-1.

III. GPF Optimization Problem

The GPF scheduling metric proposed in [2] exploits two parameters in order to obtain near optimal user throughputs and to adjust the fairness performance in such a way that the NGMN requirement is accomplished. The system model considers a set $\mathcal{U}_t$ of preselected users with an infinite buffer model with the minimum requested bit rate of 0kbps. At each TTI t a set of $\mathcal{B}$ orthogonal sub-carriers called Resource Blocks (RBs) [10] should be shared among the active users in order to solve the GPF integer linear programming optimization problem subject of convex set of constraints as shown by Eq. 1:

$$\begin{aligned} &\max_{b_{i,j}} \sum_{i\in\mathcal{U}_t}\sum_{j\in\mathcal{B}} b_{i,j}[t]\cdot\left\{\left(r_{i,j}[t]\right)^{\beta_i} / \left(\overline{T}_i[t]\right)^{\alpha_i}\right\} \\ &s.t. \sum_{i\in\mathcal{U}_t} b_{i,j}[t] = 1, \forall i \in \mathcal{U}_t \\ &\quad b_{i,j}[t] \in \{0,1\}, \forall i \in \mathcal{U}_t \end{aligned} \tag{1}$$

where $b_{i,j}[t]$ represents the RB allocation decision for user i and RB j, $r_{i,j}[t]$ is the achievable user rate determined based on the instantaneous CQI reports, and $\overline{T}_i[t]$ denotes the achieved user throughput averaged with the exponential moving filter. By using a fine tuning of $\alpha_t \in [0,1]$ and $\beta_t \in [0,1]$ parameters, a varying level of fairness can be obtained at each TTI t. For this reason, the technique is entitled *double GPF parameterization* (*GPF-2*). If $\alpha_t = 0$ and $\beta_t = 1$, the GPF scheme becomes the maximum throughput (MT) scheduling rule whereas when $\alpha_t = 1$ and $\beta_t = 0$, the obtained metric becomes the max-fairness technique. For the particular case of $\alpha_t = 1$ and $\beta_t = 1$, the well-known proportional fair metric (PF) is obtained. The illustrative mean user throughput and JFI fairness tradeoffs for the aforementioned particular GPF rules are highlighted in Fig. 1.a. The simple GPF parameterization used by other adjusting policies in [6], [7], [8] and [9] is represented by the special case of GPF simple parameterization (GPF-1) when $\beta_t = 1$ and $\alpha_t \in [0, \alpha_{max}]$ where

$\alpha_{max} >> 1$. Obviously, it is expected that CACLA-1 requires more time to optimize α_t than CACLA-2 which explores for a more restrictive domain of parameters. However, the double parameterization learning technique should adapt the set of (α_t, β_t) parameters TTI-by-TTI in order to reach the optimal or feasible scheduler state (green tradeoff values from Fig. 1. a) such that:

$$\begin{cases} \alpha_t = \alpha_{t-1} + \Delta\alpha_t \\ \beta_t = \beta_{t-1} + \Delta\beta_t \end{cases} \quad (2)$$

where $\mathcal{A}_t = (\Delta\alpha_t, \Delta\beta_t)$ is the MLPNN output space or the RL algorithm action space. For CACLA-2, the action space is $\mathcal{A}_t \in \mathbb{R}^2_{[-1,1]}$ whereas for CACLA-1, the simple parameterization involves $\mathcal{A}_t = (\Delta\alpha_t) \in \mathbb{R}_{[-1,1]}$. For other RL algorithms with discrete action spaces exposed and analyzed in this paper, the action at TTI t becomes $\mathcal{A}_t = \{\Delta\alpha_k\}$ where $k = 1,..,|\mathcal{A}_t|$. Let us define $\mathcal{S}_t^S$ the continuous and multidimensional scheduler state at TTI t. The scheduler evolves to the next state $\mathcal{S}_{t+1}^S$ when the discrete or continuous action $\mathcal{A}_t$ is applied for the scheduling procedure.

The role of the RL approaches is to drive the scheduler in the feasible state $(\mathcal{S}_{t+1}^S \in \mathcal{FA})$, where $\mathcal{FA} \in \mathbb{R}^{|\mathcal{S}|}_{[-1,1]}$ represents the collection of multi-dimensional data points when the scheduler meets the feasible state for different channel and network conditions. When the applied action moves the throughput-JFI domain on the left side of $\{\mathcal{FA}\}$ zone, the scheduler is declared unfair (MT scheduling rule case) and $(\mathcal{S}_{t+1}^S \in \mathcal{UF})$, where $\{\mathcal{UF}\} \in \mathbb{R}^{|\mathcal{S}|}_{[-1,1]}$ denotes the region of unfair states. Otherwise, the scheduler is considered to be over-fair (MF scheduling rule case) and $(\mathcal{S}_{t+1}^S \in \mathcal{OF})$. By translating the quantitative tradeoff evaluation (Fig.1.a.) to qualitative NGMN fairness evaluation (Fig.1. b.), the scheduler state space status is decided based on the NGMN Objective Function (NOF) $\Phi\left[\hat{\hat{T}}_i[t]\right]$, where the sub-space $\left\{\hat{\hat{T}}_i[t]\right\} \subset \mathcal{S}_t^S$ represents the normalized user throughput (NUT) for $\forall i = 1,..,|\mathcal{U}_t|$. Let us define $\Upsilon\left(\hat{\hat{T}}_i\right)$ as the CDF function when all observations $\left\{\hat{\hat{T}}_i[t]\right\}$ are log-normal distributed. If $\Upsilon^{Req}\left(\hat{\hat{T}}_i\right)$ represents the NGMN fairness requirement (continuous oblique black line), then the aggregated NOF function is calculated based on Eq. 3:

$$\Phi\left[\mathcal{S}_t^S\right] = (1/|\mathcal{U}_t|)\sum_{i\in\mathcal{U}_t}\left[\Upsilon\left(\hat{\hat{T}}_i\right) - \Upsilon^{Req}\left(\hat{\hat{T}}_i\right)\right] \quad (3)$$

where $\Upsilon^{Req}\left(\hat{\hat{T}}_i\right) = \hat{\hat{T}}_i$ if $\hat{\hat{T}}_i \le 1$, and otherwise $\Upsilon^{Req}\left(\hat{\hat{T}}_i\right) = 1$. Based on NGMN specifications [1], the scheduler state is fair $(\mathcal{S}_{t+1}^S \in \mathcal{F})$ only and only if $\Phi\left[\mathcal{S}_t^S\right] \le 0$, where the fairness region is $\{\mathcal{F}\} = \{\mathcal{FA}\} \cup \{\mathcal{OF}\}$. The delimitation between feasible area and over-fairness area is given by the superior CDF limit Υ^{Max} (oblique dot black line in Fig. 1.b) such that:

$$\Upsilon^{Max}\left(\hat{\hat{T}}_i\right) = \begin{cases} \Upsilon^{Req}\left(\hat{\hat{T}}_i\right) - \xi, \ if \ \hat{\hat{T}}_i \le 1 + \xi \\ 1,, \ if \ \hat{\hat{T}}_i > 1 + \xi \end{cases} \quad (4)$$

where $\xi \in \mathbb{R}^+_{[0,1]}$ is the confidence parameter that can guarantee the feasible region detection during the exploration period. For a larger ξ parameter, the $\{\mathcal{FA}\}$ region can be detected much faster by degrading the system throughput whereas when the confidence parameter is small enough, more exploration time is required for CACLA-2 to localize the feasible state. The scheduler state status is decided based on Eq. 5:

$$\mathcal{S}_{t+1}^S \in \begin{cases} \{\mathcal{UF}\}, \ if \ \exists\Upsilon_i > \Upsilon_i^{Req}, \forall i = 1,..,|\mathcal{U}_t| \\ \{\mathcal{FA}\}, \ if \ \Upsilon_i^{Req} > \Upsilon_i > \Upsilon_i^{Max}, \forall i = 1,..,|\mathcal{U}_t| \\ \{\mathcal{OF}\}, \ if \ \exists\Upsilon_i < \Upsilon_i^{Max}, \forall i = 1,..,|\mathcal{U}_t| \end{cases} \quad (5)$$

The purpose of CACLA-2 is to find the feasible state $(\mathcal{S}_{t+1}^S \in \mathcal{FA})$ based on action $\mathcal{A}_t$ applied at TTI t and to keep this desirable state as long as possible. Other regions such as $\{\mathcal{UF}\}$ or $\{\mathcal{OF}\}$ are considered undesirable for the learning procedure.

IV. The Self-Learning LTE-A Scheduler Architecture

The proposed actor-critic RL algorithm learns the optimal policy of $(\Delta\alpha_t, \Delta\beta_t)$ actions based on the interaction between the conventional LTE scheduler and the novel controller. At each TTI t, the controller receives from the scheduler a new $\mathcal{P}_t \in \{\mathcal{S}_{t-1}^S, \mathcal{A}_{t-1}, \mathcal{RW}_t, \mathcal{S}_t^S\}$ MDP problem, where $\mathcal{RW}: \mathbb{R} \to \mathbb{R}$ constitutes the reward function which is a modified version of aggregated NOF function. For each $\mathcal{P}_t$, the LTE controller has to decide based on RL approach which action $\mathcal{A}_t$ should be applied at the current TTI t (Fig. 2). In other words, the controller has to learn how to behave for many $\mathcal{P}$ problems. In this sense, the controller requires the state value function $V_t(\mathcal{S}_t^S)$ where $V_t: \mathbb{R}^{|\mathcal{S}|}_{[-1,1]} \to \mathbb{R}$ and the action-state value $A_t(\mathcal{S}_t^S)$ where $A_t: \mathbb{R}^{|\mathcal{S}|}_{[-1,1]} \to \mathbb{R}^{|\mathcal{A}|}_{[-1,1]}$. When RL approaches with discrete actions are used, different action-state values Q_t^k are requested for each $\mathcal{A}_t^k$ action. The idea is to upgrade these values based on an infinite number of iterations by using the temporal learning principles [11]. The way how the tuple $(V_t, A_t(Q_t))$ is updated TTI-by-TTI based on $\mathcal{P}_t$ MDP problems determines the RL algorithm type. The time period when (V_t, A_t) values are updated is called *training stage* and each RL approach has different performance impacts in the in the scheduling quality. When the learned state and action-state

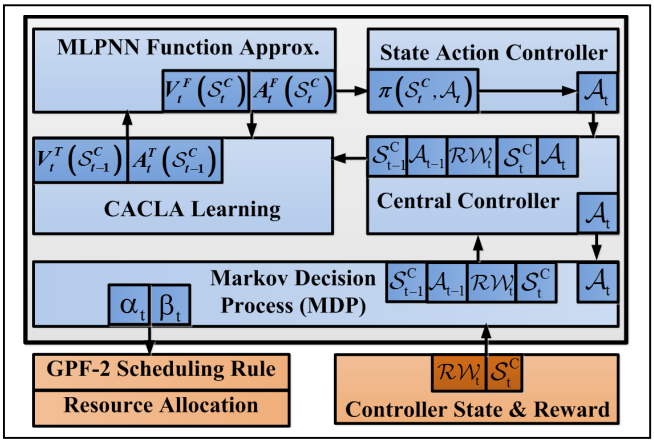


Fig.2 The architecture of the self-learning LTE-A scheduler

value functions are directly applied to the new $\mathcal{P}$, then the exploitation stage is performed. The purpose of the training stage is to find the optimal policy $\pi^*(\mathcal{S}_t^S, \mathcal{A}_t)$ which can provide the largest amount maximum rewards averaged over the number of training epochs or number of training TTIs. In order to reduce the MLPNN structure complexity and to speed up the convergence in the $\mathcal{FA}$ region, the initial scheduler state space has to be aggregated by using a special pre-processing stage.

A. LTE Controller State Space

Due to the continuous and high dimensionality characteristics of the original scheduler state space $\mathcal{S}_t^S$, the MLPNN non-linear function is used in order to offer good generalizations for the $\left(V_t, A_t\left(Q_t\right)\right)$ values. In order to avoid the state dependency on the number of active radio bearers, the initial state space $\mathcal{S}_t^S$ is converted into a more representative and compacted state such as $\mathcal{S}_t^C$ by keeping only the relevant information which can affects the action selection for the scheduling procedure. Then, the proposed controller state space becomes:

$$\mathcal{S}_t^C = \left\{\alpha_{t-1},\ \beta_{t-1},\ \mu_t^{\hat{T}},\ \sigma_t^{\hat{T}},\ d_t^R,\ |\mathcal{U}_t|,\ \rho_t\right\} \tag{6}$$

where $\mu_t^{\hat{T}}$ and $\sigma_t^{\hat{T}}$ are the mean and the standard deviations, respectively for the log-normal distribution of NUT observations, d_t^R is the minimum/maximum difference between Υ_i and Υ_i^{Req} percentiles when the system is fair/unfair as indicated in [9], and ρ_t represents the system status flags which indicates that $\rho_t = -1$ when $\mathcal{S}_t^C \in \mathcal{UF}$, $\rho_t = 0$ when $\mathcal{S}_t^C \in \mathcal{OF}$ and finally $\rho_t = 1$ for the feasible state. Basically, the distance d_t^R decides the state $\mathcal{S}_t^C$ status such that: if $\exists d_t^{i,R} = \Upsilon_i - \Upsilon_i^{Req} > 0,\ \forall i \in \mathcal{U}_t$ then $\mathcal{S}_t^C \in \mathcal{UF}$ and the distance is $d_t^R = -max\left(d_t^{i,R}\right)$, and when $d_t^{i,R} < 0, \forall i \in \mathcal{U}_t$ the controller is declared fair $\mathcal{S}_t^C \in \mathcal{F}$ and $d_t^R = min\left(d_t^{i,R}\right)$. For the particular case when $\mathcal{S}_t^C \in \mathcal{F}$ and $d_t^R \in [0, \xi]$ then the state becomes $\mathcal{S}_t^C \in \mathcal{FA}$. It is important to point out that all the above calculations are performed for the CDF domain of interest of $\Upsilon_i \in [0, 0.7]$. Beyond this interval, the CDF percentiles are not able to decide the controller state status.

B. Reward Function

The objective function from Eq. 3 is not suitable as a reward function due to the oscillated characteristics of the LTE scheduling procedure. Then, the particularities of GPF-2 should be highlighted for the reward function computation. Compared with CACLA-1 [9], CACLA-2 reveals the existence of multiple optimal solutions $\left(\alpha_t^{opt}, \beta_t^{opt}\right)$ when $\mathcal{S}_t^C \in \mathcal{FA}$. In Fig. 1, the feasibility is reached when let us say $\left(\alpha_t^{opt} = 0.5, \beta_t^{opt} = 1\right)$. If $\left(\alpha_t = \alpha^{opt}, \beta_t \searrow\right)$, then $\mathcal{S}_t^C \in \mathcal{UF}$; when $\left(\alpha_t \nearrow, \beta_t = 1\right)$, then the system tends to become over-fair $\mathcal{S}_t^C \in \mathcal{OF}$. Based on these principles, the feasibility can be reached for multiple optimal set of parameters when $\left(\alpha_t \searrow, \beta_t \searrow\right)$. Then the reward function for GPF-2 can be divided as indicated in Eq. 7:

$$\mathcal{RW}_t = \begin{cases} \mathcal{RW}_t^{UF},\ if\ \mathcal{S}_{t-1}^C \in \{\mathcal{UF}, \mathcal{FS}, \mathcal{OF}\}, \mathcal{S}_t^C \in \mathcal{UF} \\ 1,\ if\ \mathcal{S}_{t-1}^C \in \{\mathcal{UF}, \mathcal{FS}, \mathcal{OF}\}, \mathcal{S}_t^C \in \mathcal{FS} \\ \mathcal{RW}_t^{OF},\ if\ \mathcal{S}_{t-1}^C \in \{\mathcal{UF}, \mathcal{FS}, \mathcal{OF}\}, \mathcal{S}_t^C \in \mathcal{OF} \end{cases} \tag{7}$$

When $\mathcal{S}_t^C \in \mathcal{UF}$, the reward function $\mathcal{RW}_t^{UF}$ can be modeled by using the tuple of $\left(\Delta\alpha_{t-1}, \Delta\beta_{t-1}, \alpha_{t-1}, \beta_{t-1}\right)$. If the feasibility is reached, then the previous action should be granted with the maximum reward. The necessary conditions for the $\mathcal{FA}$ region convergence are: $\Delta\alpha_{t-1} > 0$ and $\Delta\beta_{t-1} < 0$. The decision can be further divided into two situations: when $\alpha_{t-1} < \beta_{t-1}$ the reward function should contain a weighted sum of $\left(\Delta\alpha_{t-1}, \Delta\beta_{t-1}\right)$; when $\alpha_{t-1} > \beta_{t-1}$, the role of β_{t-1} is not important anymore and the reward should focus only on $\Delta\alpha_{t-1}$. In the opposite case when $\Delta\alpha_{t-1} < 0$ and $\Delta\beta_{t-1} > 0$, the action should be severely punished since the state moves farer away from $\mathcal{FA}$ zone. Therefore, when $\mathcal{S}_t^C \in \mathcal{UF}$, the reward function is calculated based on Eq. 7.a.:

$$\mathcal{RW}_t^{UF} = \begin{cases} 0.5 \cdot \left(|\Delta\alpha_{t-1}| + |\Delta\beta_{t-1}|\right), if\ \Delta\alpha_{t-1} > 0, \Delta\beta_{t-1} < 0, \alpha_{t-1} < \beta_{t-1} \\ -0.5 \cdot \left(1 + |\Delta\beta_{t-1}|\right), if\ \Delta\alpha_{t-1} < 0, \Delta\beta_{t-1} < 0, \alpha_{t-1} < \beta_{t-1} \\ -0.5 \cdot \left(|\Delta\alpha_{t-1}| + 1\right), if\ \Delta\alpha_{t-1} > 0, \Delta\beta_{t-1} > 0, \alpha_{t-1} < \beta_{t-1} \\ |\Delta\alpha_{t-1}|, if\ \Delta\alpha_{t-1} > 0, \Delta\beta_{t-1} < 0, \alpha_{t-1} > \beta_{t-1} \\ -1, if\ \Delta\alpha_{t-1} < 0, \Delta\beta_{t-1} > 0 \end{cases} \tag{7.a}$$

For the over-fairness case, the reward function should follow the opposite direction of Eq. 7.a. The desirable situation is denoted by $\Delta\alpha_{t-1} < 0$ and $\Delta\beta_{t-1} > 0$ and the undesirable one by the case when $\Delta\alpha_{t-1} > 0$ and $\Delta\beta_{t-1} < 0$. In the first instance

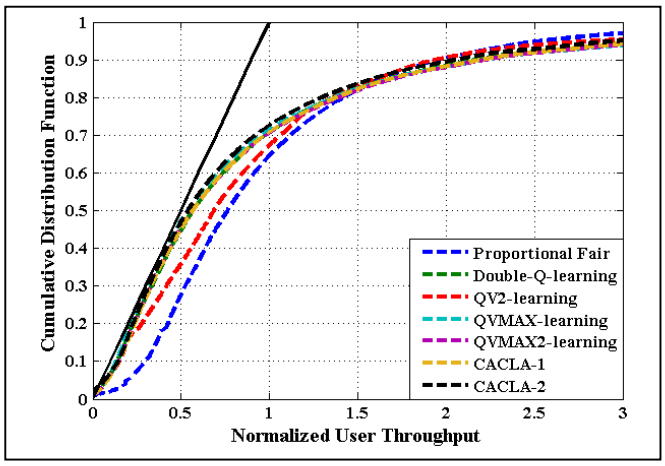


Fig.3. The CDF curve of the proposed policies

α_{t-1} and β_{t-1} must be compared in order to determine whether $\Delta\beta_{t-1}$ can help or not in reaching of the feasible state. Equation 7.b. highlights the proposed reward model for the particular situation when $\mathcal{S}_t^C \in \mathcal{OF}$.

$$\mathcal{RW}_t^{OF} = \begin{cases} 0.5\cdot\left(\left|\Delta\alpha_{t-1}\right|+\left|\Delta\beta_{t-1}\right|\right), if\ \ \Delta\alpha_{t-1}<0, \Delta\beta_{t-1}>0, \alpha_{t-1}>\beta_{t-1} \\ -0.5\cdot\left(1+\left|\Delta\beta_{t-1}\right|\right), if\ \ \Delta\alpha_{t-1}>0, \Delta\beta_{t-1}<0, \alpha_{t-1}>\beta_{t-1} \\ -0.5\cdot\left(\left|\Delta\alpha_{t-1}\right|+1\right), if\ \ \Delta\alpha_{t-1}<0, \Delta\beta_{t-1}<0, \alpha_{t-1}>\beta_{t-1} \\ \left|\Delta\alpha_{t-1}\right|, if\ \ \Delta\alpha_{t-1}<0, \Delta\beta_{t-1}>0, \alpha_{t-1}<\beta_{t-1} \\ -1, if\ \ \Delta\alpha_{t-1}>0, \Delta\beta_{t-1}<0 \end{cases} \quad (7.b)$$

During the training stage, the role of CACLA-2 RL algorithm is to collect the maximum rewards TTI-by-TTI and to select based on the MDP problem $\mathcal{P}_t$ such actions $\mathcal{A}_t$ in order to improve and to refine the final policy of GPF-2 parameters.

C. CACLA-2 RL Algorithm

Based on CACLA principles, two MLPNN functions are used for the approximations of state and action-state values. Let us define $A_t^F = \mathcal{F}_{MLP}^A\left(\mathcal{W}_t^A, \mathcal{S}_t^C\right)$ and $V_t^F = \mathcal{F}_{MLP}^V\left(\mathcal{W}_t^V, \mathcal{S}_t^C\right)$ the forwarded action-state and state values respectively, where $\left(\mathcal{W}_t^A, \mathcal{W}_t^V\right)$ are the trained MLPNN weights at TTI t. The MLPNN weights are updated according to the gradient-descent principle which aims to minimize the mean-square errors $\widetilde{E}_t^A$ and $\widetilde{E}_t^V$ based on Eq. 8.a and Eq.8.b, respectively:

$$\widetilde{E}_t^A\left(\mathcal{W}_{t-1}^A, \mathcal{S}_{t-1}^C\right) = \eta_A \cdot \sum_{\mathcal{H}}\left[A_t^T\left(\mathcal{S}_{t-1}^C\right) - A_t^F\left(\mathcal{S}_{t-1}^C\right)\right] \quad (8.a)$$

$$\widetilde{E}_t^V\left(\mathcal{W}_{t-1}^V, \mathcal{S}_{t-1}^C\right) = \eta_V \cdot \sum_{\mathcal{H}}\left[V_t^T\left(\mathcal{S}_{t-1}^C\right) - V_t^F\left(\mathcal{S}_{t-1}^C\right)\right] \quad (8.b)$$

where $A_t^T\left(\mathcal{S}_{t-1}^C\right)$ and $V_t^T\left(\mathcal{S}_{t-1}^C\right)$ are the target values, η_A and η_V are the learning rates, and $\mathcal{H}$ is the total number of hidden nodes for all MLPNN layers. The errors from (8.a) and (8.b) are back-propagated layer-to-layer for each MLPNN node. The target state value is updated at TTI t when the reward value is received as a result of applying the previous action

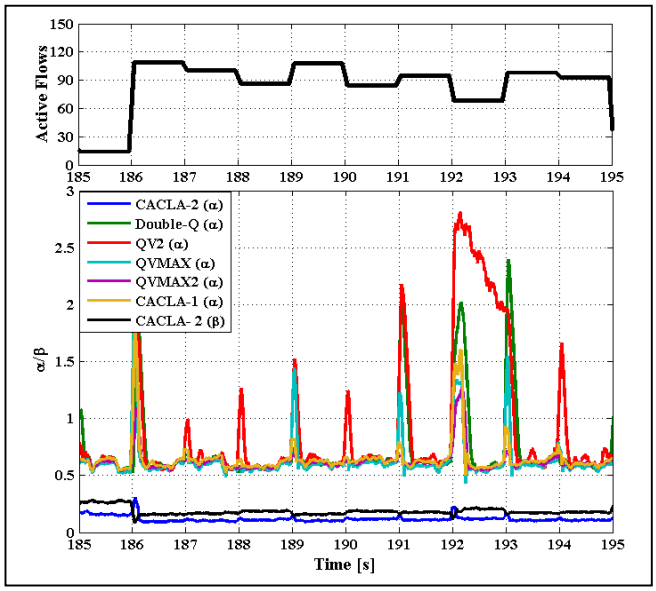


Fig.4. Obtained policies

$\mathcal{A}_{t-1}$ in the previous state $\mathcal{S}_{t-1}^C$ according to Eq. 9:

$$V_t^T\left(\mathcal{S}_{t-1}^C\right) = \mathcal{RW}_t\left(\mathcal{A}_{t-1}, \mathcal{S}_{t-1}^C\right) + \gamma \cdot V_t^F\left(\mathcal{S}_t^C\right) \quad (9)$$

where $\gamma \in [0,1]$ is the discount factor that indicates the importance of future scheduler rewards.

In order to find the best policy of GPF parameters, the training procedure uses a combination of exploration and exploitation stages which permit to select greedy actions with a probability of $(1-\varepsilon_t)$ such as:

$$\mathcal{A}_t = \begin{cases} A_t^F\left(\mathcal{S}_t^C\right)\ \ if\ \ \left(1-\varepsilon_t\right) \geq \varepsilon_{Th}\ \ (a) \\ \mathcal{K}_t \in \mathbb{R}_{[-1,1]}^2\ \ if\ \ \left(1-\varepsilon_t\right) < \varepsilon_{Th}\ \ (b) \end{cases} \quad (10)$$

where $\mathcal{K}_t$ is the two-dimensional real random number and ε_{Th} is the greedy threshold which decides if there is policy evaluation (10.a) or policy improvement (10.b). Equation 10 is entitled actor scheme for the CACLA-2 algorithm. The critic scheme updates the action target value in state $\mathcal{S}_{t-1}^C$ based on:

$$A_t^T\left(\mathcal{S}_{t-1}^C\right) \leftarrow A_t^F\left(\mathcal{S}_t^C\right)\ \ if\ \ V_t^T\left(\mathcal{S}_{t-1}^C\right) > V_t^F\left(\mathcal{S}_{t-1}^C\right) \quad (11)$$

According to Eq. 11, the target action value for the state $\mathcal{S}_{t-1}^C$ is updated at TTI t only and only if the reward $\mathcal{RW}_t\left(\mathcal{A}_{t-1}, \mathcal{S}_{t-1}^C\right)$ value can improve the state $\mathcal{S}_{t-1}^C$ value. The same principles are applied to CACLA-1[9].

D. Other RL Candidates

The performance of actor-critic schemes is compared against the actor RL schemes which are well known in the specialty literature. Double Q-Learning [11] is a modified version of the standard Q-learning that uses a double estimator function in the sense that two agents store two action value functions. QV-2 learning works by keeping a track of both

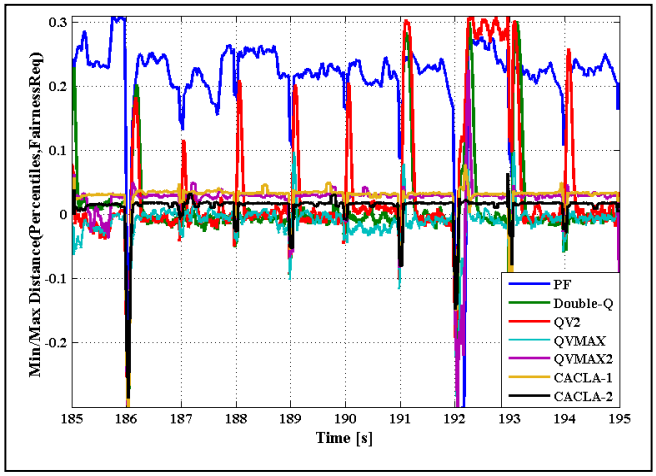


Fig.5. Measured min/max distances from the NGMN requirement

TABLE I. LTE SCHEDULER PARAMETERS

Parameter Names	Values
Main Parameters	3GPPP[14]/NGMN[1]
System bandwidth	20MHz
Cell radius/ User Speed	1000 m/30km/h
Channel Model	Rayleigh Fading (Vehicular A)
Shadowing std. deviation	8 dB
Path Loss/Penetration Loss	128.1 + 37.6 log(d)/10 dB
Interfered cells	0
Carrier frequency/DL power	2GHz/43dBm
Exploration/Experience Replay	1000 s / 500 s
/Exploitation periods	/ 200 s
Discrete $\Delta\alpha$	$\pm10^{-4},\pm10^{-3},\pm10^{-2},\pm10^{-1},\pm5\cdot10^{-2},0$
No. of MLPNN hidden layers	3
Activation function	Tangent Hyperbolic
No. of hidden nodes/layer	50
Confidence Parameter (ξ)	0.05

action and state values and differs from the original QV-learning from the MLNN error functions point of view [12]. QVMAX and QVMAX2 algorithms are off-line RL procedures in the sense that they combine the state, action-state values and error function computation based on QV2 and Q learning approaches [13]. The RL candidates use discrete action sets by adjusting the GPF-1 parameterization problem.

V. SIMULATION RESULTS

In order to prove the eligibility of CACLA-2 actor-critic RL algorithm in comparison with other methods, the considered scenario fluctuates at each 1s the number of active users based on the ε -greedy probability in the interval of [10,120], and the user mobility is considered to be random walk with 30kmph speed. The evaluation of each scheduling algorithm is achieved by using the same conditions for interference, path loss, shadowing and fast fading. Each base station transmits with the same power which is equally distributed for all RBs. The best effort traffic type is considered for the downlink transmission purpose. The CQI feedback which is sent in the uplink direction is considered to be errorless. The rest of the parameters of the LTE scheduler can be found in TABLE I.

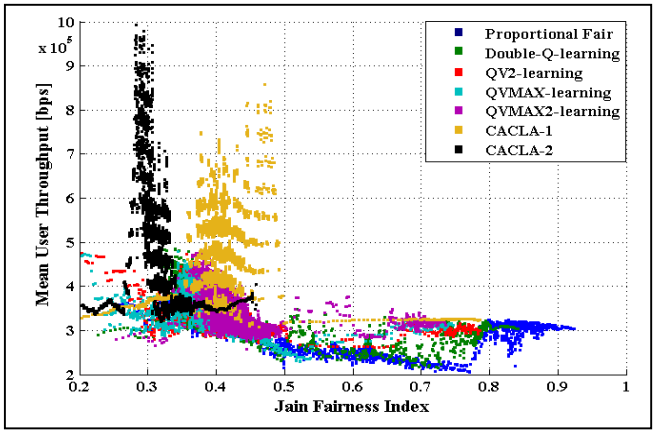


Fig.6. JFI –Mean user throughput tradeoff

The LTE controller trains each RL algorithm for 1000s. Then, the resulted policy is exploited for 200s. Due to the fact that Double-Q-learning algorithm spends too much time in the irrelevant state-space regions, after the exploration period, the controller is decoupled from the LTE scheduler and is re-trained based on the visited MDP problems for a duration of about 500s. Except CACLA-1, CACLA-2 and Double-Q algorithms, the other candidates use the Boltzmann policy for the action selection procedure [13]. The rest of the parameters for LTE controller can be found in TABLE II.

As shown in Fig.3, CACLA-2, CACLA-1, QVMAX, QXMAX2 and Double-Q RL algorithms satisfy the NGMN fairness requirement when the number of active users remains constant. From the CDF perspective, the PF scheduling metric and QV-2 RL algorithm localize the scheduler in the $\mathcal{OF}$ region. The evolution of learned policies times is depicted in Fig. 4 based on the number of bearers in the active state. QV-2, Double-Q and QVMAX learning algorithms provide the highest fluctuations of α_t parameter with the lowest policy revenue capacity to the optimal value. On the other side, CACLA-1 and CACLA-2 exploit the critic scheme advantage by keeping the policy oscillations in acceptable limits. By using two continuous actions, CACLA-2 is able to recover the $\mathcal{FA}$ state much faster when the traffic load is varying. By increasing α_t and decreasing β_t when the number of users increases from 12 to 115 (Fig. 4), the policy stabilizes in less than 10ms by recovering the stability of the policy. A significant system throughput gain can be achieved by minimizing in the same time the percentage of TTIs when the scheduler is located in the $\mathcal{UF}$ operating area.

When the minimum/maximum NGMN distance d_t^R is considered (Fig. 5), CACLA-2 outperforms the main candidate CACLA-1 by maintaining the system in the minimum distances range of [0, 0.03]. The result of the policy fluctuations of other candidates is directly impacted in Fig. 5 where the NGMN distances converge much slower than the actor-critic schemes. The PF scheduling rule indicates the highest NGMN distance for all transmission period. These concepts explain the higher throughput gain from Fig. 6 of actor-critic schemes when compared against the other candidates. In particular, CACLA-2 indicates a throughput gain

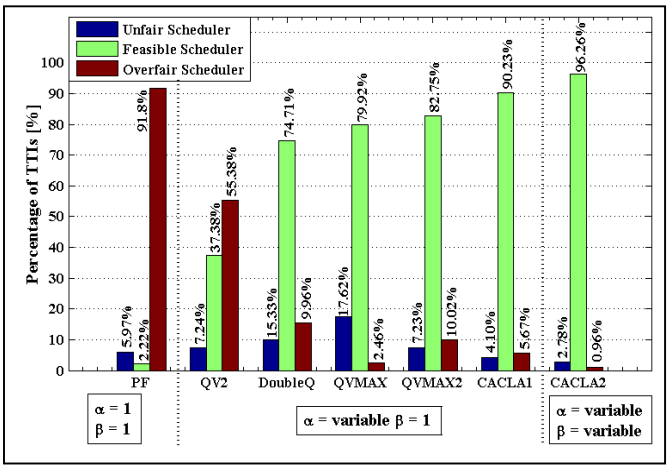


Fig. 7 Percentage of TTIs when the scheduler seats on the UF/FEA/OF states regions

TABLE II. LTE CONTROLLER PARAMETERS

Method	η_V	η_Q	η_A	γ	Exploration Type
Double-Q	-	0.01	-	0.99	ε -greedy (ε_{Th} =0.05)
QV2	0.0001	0.01	-	0.95	Boltzmann (τ =1)
QVMAX	0,0001	0.01	-	0.95	Boltzmann (τ =10)
QVMAX2	0.0001	0.01	-	0.95	Boltzmann (τ =10)
CACLA-1	0.01	-	0.01	0.99	ε -greedy (ε_{Th} =0.5)
CACLA-2	0.01	-	0.01	0.99	ε -greedy (ε_{Th} =0.5)

when compared with CACLA-1 of about 0.2Mbps. The PF metric shows the worst performance even when the JFI-mean user throughput tradeoff is considered leading to the waste of system capacity when the NGMN fairness requirement is considered. From the percentage of TTIs when the system state is $\mathcal{UF}$, $\mathcal{FA}$ or $\mathcal{OF}$ (Fig. 7) in the exploitation state, the static parameterization of PF scheduling rule shows the highest amount of TTIs when the scheduler is over-fair and the lowest percentage of TTIs when the system is feasible. When the simple parameterization is used, QV2-learning constitutes the worst choice when the scheduler state is $\mathcal{S}_t^C = \{\mathcal{FA}, \mathcal{OF}\}$. From the view point of the number of TTIs when the scheduler is over-fair, the best performance is obtained by using the QVMAX policy. CACLA-1 algorithm outperforms the other candidates with the simple parameterization scheme when the feasible region is met. When the proposed GPF-2 parameterization is used, CACLA-2 outperforms any of other RL methods. CACLA-2 gains more than 3000 TTIs from the $\mathcal{UF}$ region which are valued by the $\mathcal{FA}$ zone. In the same time, CACLA-2 gains around 6% feasible TTIs when compared with the main candidate CACLA-1 algorithm. From other perspective, by increasing the number of feasible TTIs, the number of reward punishments $\left(\mathcal{RW}_t = -1\right)$ in the exploitation stage is strongly reduced. This concept highlights the quality of the proposed policy and the ability of recovering the desired feasible state in less than 10 TTIs when severe changes in the traffic load and user channel conditions appear.

VI. CONCLUSIONS

The current work shows that the use of the double GPF parameterization increases the percentage of TTIs with 6% when the scheduler is feasible in comparison with the simple parameterization technique. The percentages of TTIs when the system is considered $\mathcal{UF}$ or $\mathcal{OF}$ indicate a real improvement of about 1.32% and 4.71%, respectively, when CACLA-2 is performed. By using double action space, the resulted policy indicates lower fluctuations when the traffic load drastically changes. In conclusion, the double parameterization of the GPF scheduling rule presents real improvements in terms of system throughput gains and percentages of TTIs when the system is considered feasible.

REFERENCES

[1] R. Irmer, *Radio Access Peiformance Evaluation Methodology*, Next Generation Mobile Networks Std. V 1.3, January 2008.

[2] C. Wengerter, J. Ohlhorst, and A. von Elbwart, "Fairness and throughput analysis for generalized proportional fair frequency scheduling in OFDMA," in *Proc. IEEE Veh. Tech. Conf., VTC-Spring*, vol. 3, Stockholm, Sweden, May 2005, pp. 1903 – 1907.

[3] H. van Hasselt and M. Wiering, "Reinforcement Learning in Continuous Action Spaces, " in *Proceedings of IEEE International Symposium on Approximate Dynamic Programming and Reinforcement Learning (ADPRL07)*, Honolulu, HI, USA, pp. 272-279, 2007.

[4] I.S. Comşa, M. Aydin, S. Zhang, P. Kuonen and J. F. Wagen, "Reinforcement Learning based Radio Resource Scheduling in LTE-Advanced," in *17th International Conference on Automation and Computing (ICAC)*, pp. 219 – 224, Sept. 2011.

[5] I.S. Comşa, M. Aydin, S. Zhang, P. Kuonen and J. F. Wagen, "Multi Objective Resource Scheduling in LTE Networks Using Reinforcement Learning," in *International Journal of Distributed Systems and Technologies (IJDST)*, vol. 3(2), pp. 39-57, April 2012.

[6] I.S. Comşa, S. Zhang, M. Aydin, P. Kuonen, and J. F. Wagen, "A Novel Dynamic Q-Learning-Based Scheduler Technique for LTE-Advanced Technologies Using Neural Networks, " in *37th Annual IEEE Conference on Local Computer Networks (LCN)*, pp. 332-335, Oct. 2012.

[7] S. Schwarz, C. Mehlführer, and M. Rupp, "Throughput Maximizing Multiuser Scheduling with Adjustable Fairness," *in IEEE International Conference on Communications*, pp. 1-5, June 2010.

[8] M. Proebster, C. M. Mueller, and R. Bakker, "Adaptive Fairness Control for a Proportional Fair LTE Scheduler, " in *IEEE 21st International Symposium on Personal Indoor and Mobile Radio Communications (PMIRC)*, pp. 1504-1509, Sept. 2010.

[9] I.S. Comşa, M. Aydin, S. Zhang, P. Kuonen, J. F. Wagen and L. Yao, "Scheduling Policies Based on Dynamic Throughput and Fairness Tradeoff Control in LTE-A Networks," in *39th Annual IEEE Conference on Local Computer Networks (LCN)*, Sept. 2014.

[10] L. Zou, R. Trestian and G.-M. Muntean, "A Utility-based Priority Scheduling Scheme for Multimedia Delivery over LTE Networks," in *IEEE International Symposium on Broadband Multimedia Systems and Broadcasting (BMSB)*, pp. 1-7, June 2013.

[11] H. van Hasselt, "Double Q-learning, " in *Advances in Neural Information Processing Systems 23 (NIPS 2010)*, Vancouver, British Columbia, Canada, pp. 2613-2622.

[12] M.A. Wiering, "QV(lambda)-learning: A New On-policy Reinforcement Learning Algorithm, " in *Proceedings of the 7th European Workshop on Reinforcement Learning*, pp. 17-18, 2005.

[13] M.A. Wiering and H. van Hasselt, "The QV Family Compared to Other Reinforcement Learning Algorithms," in *Proceedings of IEEE International Symposium on Approximate Dynamic Programming and Reinforcement Learning (ADPRL)*, pp. 101-108, March 2009.

[14] *Physical layer aspects for evolved Universal Terrestrial Radio Access (UTRA)*, 3GPP Std. TR 25.814, Rev. V7.1.0, September 2006.