Reinforcement Learning-Based ACB in LTE-A Networks for Handling Massive M2M and H2H Communications

Luis Tello-Oquendo*, Diego Pacheco-Paramo†, Vicent Pla*, Jorge Martinez-Bauset*

*Instituto ITACA. Universitat Politècnica de València, Valencia 46022, Spain
†Universidad Sergio Arboleda, Bogotá, Colombia

***Abstract*—Using cellular networks for providing M2M connectivity offers numerous advantages regarding coverage, deployment costs, security, management, among others. Nevertheless, having a large number of M2M devices activated simultaneously is difficult to tackle at the eNB and it causes complications in the connection establishment. The random access channel (RACH) in LTE-A is adequate for handling H2H communications. However, for the efficient provision of simultaneous H2H/M2M communications, it is necessary to optimize the available access control mechanisms, so that network overload is avoided and a better QoS can be offered. Access Class Barring (ACB) has shown to be effective in reducing the number of simultaneous contending users. However, it is not clear how to dynamically adapt its parameters, especially in highly dynamic scenarios with bursty traffic as it can occur when M2M communications are involved. We propose a reinforcement learning algorithm to adapt the barring rate parameter of ACB. This algorithm can adapt it to different traffic conditions, reducing congestion and hence the number of collisions in the RACH. The results show that our proposed mechanism increases the access success probability for all the users while barely impacting H2H users and other KPIs.**

***Index Terms*—Access class Barring (ACB); cellular-systems; massive machine-to-machine communications; 5G; Mobile Traffic Analysis.**

I. Introduction

Internet of Things (IoT) is an important technology for the upcoming generation of wireless systems due to its capacity to provide connectivity for anyone/anything at any time and any location. It is anticipated that there will be 29 billion connected devices by 2022 [1], and the global mobile data traffic will achieve 49 exabytes (1018 bytes) by 2021 [2]. Machine-to-machine (M2M) communication is one of the fundamental parts for the realization of the IoT environment, it makes use of cellular networks, such as LTE/LTE-A, as they provide ubiquitous coverage thanks to a widely deployed infrastructure, global connectivity, high QoS, well-developed charging and security solutions, among others [3]–[5].

The ability to adapt to changing conditions while at the same time providing new services is a constant challenge that cellular network operators have to face and one that very often implies new investments on infrastructure. At the same time, the high level of success of mobile technologies and their ability to easily recollect large amounts of information on users' behavior allows for a better understanding of the demand on the network and hence the provision of new solutions for the optimization of its resources. This type of approach has been used for different purposes such as access optimization and improvement of quality of service in 3G networks [6], [7], or location management optimization [8], among others.

In LTE-A, when a user equipment (UE) desires to access the cellular network, it performs a random access procedure. The random access channel (RACH) is used to signal a connection request; it is allowed in predefined time/frequency resources, hereafter random access opportunities (RAOs) [9], [10]. The evolved Node B (eNB) has a number of preambles available for initial access to the network. These preambles are generated by Zadoff-Chu sequences due to their good correlation properties [9], [11] and are transmitted by the UEs for attempting the first access to the network, further details are explained in Section III-A.

An important problem in cellular networks that has received an important amount of attention is the management of the massive number of connections of a large number of UEs, e.g. M2M devices, because the RACH suffers from overload in these scenarios [12], [13]. Building on this, the access class barring (ACB) scheme has been included in the LTE-A Radio Resource Control specification [10] as a viable congestion control scheme. ACB is an access control scheme that redistributes the UE accesses through time by randomly delaying the beginning of the UE random access procedure according to a barring rate and a barring time. The ACB scheme is further explained in Section III-B.

There is a tradeoff between relieving congestion and the key performance indicators (KPIs) of the network when the ACB is operating and its parameters are adjusted adequately [14]. Therefore, the proper tuning of ACB parameters according with the traffic intensity is extremely important but the 3GPP does not specify how to do so dynamically. In this work, we propose a Reinforcement Learning (RL) approach to adjust dynamically the ACB barring rate. Concretely, we use Q-learning, a well-known RL technique [15], to tackle the dynamic tunning of the ACB barring rate so that the eNB can continuously adapt it to the ever-changing network traffic.

Our main contributions are summarized as follows:

- A Q-learning algorithm is designed to dynamically and autonomously tune the ACB barring rate in such a way it can react rapidly to the changes of the traffic using local information available at the eNB.
- Our proposed scheme is intelligent and does not require any modification of the network specifications; we evaluate our proposed scheme according with the KPIs defined in the 3GPP specifications [10].
- Our experiments are based on realistic traffic behavior by making use of traces from cellular network operators

to enhance the access control of simultaneous H2H and M2M communications in LTE/LTE-A networks.

The rest of the paper is organized as follows. Section II performs a review of the related work. Section III describes in detail the LTE-A random access procedure and the ACB scheme. Section IV presents the application of Q-learning to the ACB scheme. Section V describes the experiments and presents the numerical results. Finally, Section VI draws the conclusions and presents the future work.

II. Related Work

It is possible to find in the literature several works dealing with the optimization of the ACB scheme in LTE/LTE-A networks both through static and adaptive approaches. However, most of these works require considerable modifications to the network specifications. In [16], a self-organizing mechanism which aims to optimize the performance of the random access procedure is proposed for M2M and H2H traffic. However, unlike the standards, the authors assume that a control-loop for congestion between the UEs and the eNB is available, which creates more signaling. In [17], a dynamic mechanism for access control in LTE-A is proposed, for the purpose of reducing the impact that massive M2M communications can have on H2H traffic. Also, in this work they differentiate M2M traffic, allowing prioritization. However, this approach modifies ACB so it is able to send different parameters for different classes, in a similar way to extended access barring [18]. Since the number of UEs trying to access the cellular network is dynamic, and its number is not known a priori, any mechanism that aims for an optimization of ACB has to develop an estimation of this value. In [19], it is proposed a dynamic scheme for ACB that uses a Kalman filter and enhances the overall performance. Although in this work no modifications are done over the ACB mechanism, it is not possible to estimate the impact that M2M traffic has over H2H traffic, since only the first one was considered. Also in [20], an optimal value of the P_{ACB} parameter is obtained in the ideal case where the eNB has all the information about the system, and some heuristics which resemble this optimal solution are provided, where one of them changes the parameter P_{ACB} and the other changes both P_{ACB} and the number of preambles that can be acknowledged. Yet, this solution assumes that when a UE suffers a collision, it will retry in the following RAO, which is not consistent with the LTE-A specifications.

There have already been proposals based on reinforcement learning to optimize the access control of M2M UEs in cellular networks. In [21], the authors propose a Q-learning approach for a scenario where M2M and H2H traffic coexist. In this case, the reinforcement learning scheme is performed only on the M2M UEs to identify the moment on which they should transmit. Nonetheless, this scheme does not consider ACB, or the parameters that can enhance access control. In [22], the authors propose a Q-learning approach that aims to adapt the P_{ACB} as a function of the current traffic. However, they assume that the eNB knows the total number of contending users on each RAO to define the state space, which is not realistic. Also, they only consider a single type of traffic.

III. LTE-A Random Access Procedure

In this section, we provide a general overview of the random access procedure in LTE-A networks. Then, we explain both the contention-based random access in Section III-A and the Access Class Barring in Section III-B.

Two modes were defined for the random access: contention-free and contention-based. The former is used for critical situations such as handover, downlink data arrival or positioning. The latter is the standard mode for network access; it is employed by UEs to change the radio resource control state from idle to connected, to recover from a radio link failure, to perform uplink synchronization or to send scheduling requests [23].

The random access attempts of UEs are allowed in predefined time/frequency resources herein called RAOs. Two uplink channels are required for this purpose; namely, the physical random access channel (PRACH) for preamble transmission and the physical uplink shared channel (PUSCH) for data transmission. Particularly, the PRACH is used to signal a connection request when a UE attempts to access the cellular network. In the frequency domain, the PRACH is designed to fit in the same bandwidth as six resource blocks of normal uplink transmission ($6 \times 180\,\text{kHz}$); this fact makes it easy to schedule gaps in normal uplink transmission to allow for RAOs. In the time domain, the periodicity of the RAOs is determined by the parameter *prach-ConfigIndex*, provided by the eNB; a total of 64 PRACH configurations are available [9]. Thus, the periodicity of the RAOs ranges from a minimum of 1 RAO every two frames to a maximum of 1 RAO every subframe, i.e., from 1 RAO every 20 ms to 1 RAO every 1 ms [13], [24], [9], [10].

As mentioned before, the PRACH carries a preamble (signature) for initial access to the network; up to 64 orthogonal preambles are available per cell. In the contention-free mode, collision is avoided through the coordinated assignment of preambles, but eNBs can only assign these preambles during specific slots to specific UEs. In the contention-based mode, preambles are selected in a random fashion by the UEs, so there is a risk of collision, i.e., multiple UEs in the cell might pick the same preamble signature in the same RAO; therefore, contention resolution is needed. In the sequel, we focus on the analysis of the contention-based random access procedure.

A. Contention-Based Random Access Procedure

Before initiating the random access procedure, the UEs must first obtain some basic configuration parameters such as the RAOs in which the transmission of preambles is allowed. The eNB broadcasts this information periodically through the *Master Information Block* (*MIB*) and the *System Information Blocks* (*SIBs*). Once the UE has acquired this information, it may proceed with the four-message handshake illustrated in Fig. 1. Next, we describe both the four-message handshake and the backoff procedure. The interested reader is referred to [10], [23], [25], [26] for further details.

RACH preamble (*Msg1*): Whenever a UE attempts transmission, it sends a randomly chosen preamble in a RAO (*Msg1*). Due to the orthogonality of the different preambles, multiple UEs can access the eNB in the same RAO, using different preambles. The eNB can, without a doubt, decode a preamble transmitted (with sufficient power) by exactly one UE and estimate the transmission timing of the terminal. In this study, we assume that a collision occurs whenever two or more UEs transmit the same preamble at the same RAO. This goes in line with the 3GPP recommendations for the performance analysis of the RACH [27] and with most of the literature [14], [19], [28]–[31].

Random access response (*Msg2*): The eNB computes an identifier for each successfully decoded preamble, $ID =$

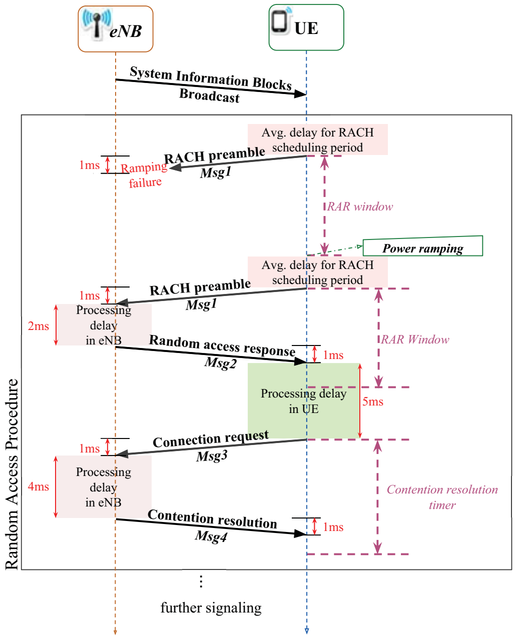

Figure 1. LTE-A contention-based random access procedure.

$f(preamble, RAO)$, and sends the *Msg2* through the physical downlink control channel (PDCCH). It includes, among other data, information about the identification of the detected preamble (ID), time alignment (TA), uplink grants (reserved PUSCH resources) for the transmission of *Msg3*, the backoff indicator (BI), and the assignment of a temporary identifier.

Exactly two subframes after the preamble transmission has ended (this is the time needed by the eNB to process the received preambles), the UE begins to wait for a time window, W_{RAR}, to receive an uplink grant from the eNB through *Msg2*.

There can be up to one RAR message in each subframe, but it may contain up to three uplink grants. Each uplink grant is associated to a successfully decoded preamble. The length of the W_{RAR}, in subframes, is broadcast by the eNB through the SIB Type 2 (SIB2) [10]. Hence, there is a maximum number of uplink grants that can be sent within the W_{RAR}. Only the UEs that receive an uplink grant can transmit the *Msg3*.

Connection request (*Msg3*): After receiving the corresponding *Msg2*, the UE adjusts its uplink transmission time according to the received TA and transmits a scheduled connection-request message, *Msg3*, to the eNB using the reserved PUSCH resources; hybrid automatic repeat request (HARQ) is used to protect the message transmission.

Contention Resolution (*Msg4*): The eNB transmits *Msg4* as an answer to *Msg3*. The eNB also applies an HARQ process to send *Msg4* back to the UEs. If a UE does not receive *Msg4* within the contention resolution timer, then it declares a failure in the contention resolution and schedules a new access attempt. For doing so, the failed UEs ramp up their power and re-transmit a new randomly chosen preamble in a new RAO, based on a uniform backoff scheme (explained next) that uses the BI received with *Msg2*.

Note that each UE keeps track of its preamble transmissions. When a UE has transmitted a certain number of preambles without success, *preambleTransMax* notified by the eNB through the SIB2 [10], the network is declared unavailable by the UE, an access problem is indicated to upper layers, and the random access procedure is terminated.

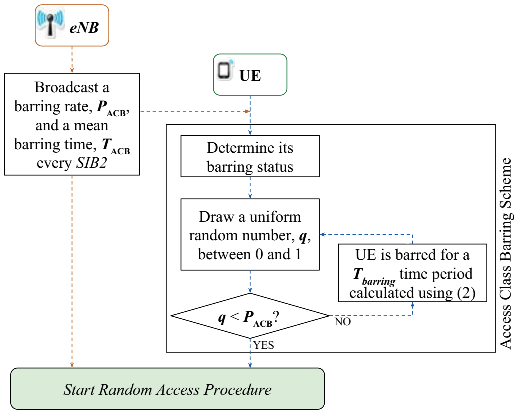

Figure 2. Access class barring scheme.

Backoff procedure: According to the LTE-A standard [23], if the RA attempt of a UE fails, regardless of the cause, the UE has to start the RA process all over again. For doing so, the UE should first perform a backoff procedure. In this procedure, the UE waits for a random time, T_{BO} [ms], until it can attempt a new preamble transmission as follows

$$T_{BO} = \mathcal{U}(0, BI), \tag{1}$$

where $\mathcal{U}(\cdot)$ stands for uniform distribution, BI is the backoff indicator defined by the eNB, and its value ranges from 0 to 960 ms. The value of BI is sent in the *Msg2*, which is read by all the UEs that sent a RACH preamble in the previous RAO. This means that every UE that did not get a *Msg2*, i.e., failed attempt, receives the BI.

B. Access Class Barring

Access Class Barring (ACB) is a congestion control scheme designed for limiting the number of simultaneous access attempts from certain UEs according to their traffic characteristics. For doing so, all UEs are assigned to 16 mobile populations, defined as access classes (ACs) 0 to 15. The population number is stored in UE's SIM/USIM. Each UE belongs to one out of the first 10 ACs (from ACs 0 to 9) and can also belong to one or more out of the five special categories (ACs 11 to 15). Thus, M2M devices may be assigned an AC between 0 and 9, and if a higher priority is needed, other classes may be used.

The main purpose of ACB is to redistribute the access requests of UEs through time to reduce the number of access requests per RAO. This fact helps to avoid massive-synchronized accesses demands to the PRACH, which might jeopardize the accomplishment of QoS objectives. Fig. 2 illustrates the ACB scheme [10], [18]. Note that ACB is applied only to the UEs that have not yet begun its RA procedure explained in Section III-A.

If ACB is not implemented, all ACs are allowed to access the PRACH. When ACB is implemented, the eNB broadcasts (through SIB2) mean barring times, $T_{\text{ACB}} \in \{4, 8, 16, \ldots, 512\,\text{s}\}$, and barring rates, $P_{\text{ACB}} \in \{0.05, 0.1, \ldots, 0.3, 0.4, \ldots, 0.7, 0.75, 0.8, \ldots, 0.95\}$, that are applied to ACs 0-9. Then, at the beginning of the RA procedure, each UE determines its barring status with the information provided from the eNB. For this, the UE generates a random number between 0 and 1, $\mathcal{U}[0, 1)$. If this number is less than or equal to P_{ACB}, the UE selects and transmits its preamble. Otherwise, the UE waits for a random time calculated as follows

$$T_{barring} = [0.7 + 0.6 \times \mathcal{U}[0,1)] \times T_{\text{ACB}}. \tag{2}$$

It is worth noting that ACB is only useful for relieving sporadic periods of congestion, i.e., when a massive number of UEs attempt transmission at a given time but the system is not continuously congested. In other words, ACB spreads the load offered to the system through time, but the total offered load is not affected.

IV. Reinforcement Learning Approach

Q-learning belongs to the category of temporal-difference RL techniques that consist of learning how to map situations to actions for maximizing a scalar reward. The learning is achieved through the interaction with the environment, so that the learner discovers which actions yield the highest reward by trying them.

Through this approach, the eNB stores a value function $Q(s,a)$ that measures the expected reward from being on a given state s and taking a given action a. In our model, the action set $\mathcal{A}$ =$\{1,2,..,16\}$ is composed of the actions that change P_{ACB} to one of its possible values, seen in section III-B. When the action chosen is a=1, then P_{ACB}=0.05, and the rest of the values are mapped sequentially. When the action chosen is a=16, then P_{ACB}=1 and the ACB mechanism is turned off. Due to the characteristics of ACB, changes on P_{ACB} can only be received by UEs through SIB2 messages, being T_{SIB2} its periodicity. Hence, the Q-learning actions that change P_{ACB} will only be taken before the transmission of an SIB2. Following the specifications [10], throughout this work we will use a value of 80 ms for T_{SIB2}. A state s, is defined as $s = \left(\overline{N_{PT}}, CV_{N_{PT}}, \Delta N_{PT}, P_{\text{ACB}}\right)$, where $\overline{N_{PT}}$ is the mean number of preamble transmissions that the eNB detected on the RAOs during a whole T_{SIB2}, $CV_{N_{PT}}$ is the variation coefficient of N_{PT} for the same period, ΔN_{PT} is the difference of mean number of preamble transmissions between the current period and the previous one, and P_{ACB} is the ACB probability that affected UEs during this period. The definition of states could be seen more clearly through Fig. 3. In time $n-1$ (which occurs just before the transmission of SIB2(n)) the eNB decides to take an action a_n based on the state s_{n-1}. The information about the action (i.e P_{ACB}) is sent in the following SIB2, and hence the access of UEs during the following 16 RAOs will depend on this information. At time n, just before sending SIB2(n+1), the eNB can calculate the values of the state s_n. For that, it will consider the 16 RAOs that lie between SIB2(n) and SIB2(n+1). It should be noted that $N_{PT} >= W_{RAR} * N_{RAR}$, and that N_{PT} only accounts for the preamble transmissions that the eNB could detect properly. Hence, it is a convenient indicator of the load on the access procedure for the eNB. Although there are 54 preambles available for the UEs, it was observed that even in very congested scenarios, it was very unlikely that $\overline{N_{PT}}$ would grow beyond 30. Therefore, and considering that these scenarios are related with very high congestion, and that changes on P_{ACB} provide little or no improvement over the KPIs of the system, we decided to aggregate all states where $\overline{N_{PT}} > 29$. Hence, the possible values for $\overline{N_{PT}}$ are between 0 and 29. On the other hand, the coefficient of variation values $CV_{N_{PT}} \in \{0, 0.2, 0.4, 0.6, 0.8\}$ were discretized to reduce the total number of states. Therefore, if the calculation of $CV_{N_{PT}}$ over the corresponding 16 RAOs lies between 0 and 0.19, the value that will be considered to define a state will be 0. The same procedure is done for the other intervals. The parameter $\Delta N_{PT}(n)$ is obtained as $\overline{N_{PT}(n)} - \overline{N_{PT}(n-1)}$. However, this value is also discretized as follows: if it is higher than 0, then it will be set as 1; if it is equal to zero, it will be set as 2; if it is smaller than zero, it will be set as 3. And finally, $P_{\text{ACB}}(n)$ is the barring factor that affected UEs during the period n, that is, the value sent in SIB2(n).

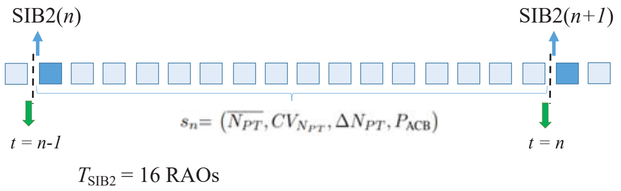
SIB2(n)
SIB2(n+1)
t = n-1
s_n= $(\overline{N_{PT}}, CV_{N_{PT}}, \Delta N_{PT}, P_{\text{ACB}})$
t = n
T_{SIB2} = 16 RAOs

Figure 3. State definition and T_{SIB2}.

As expected, the Q-value is updated according to the Q function:

$$Q(s,a) = Q(s,a) + \alpha * (\mathcal{R} + \gamma * max_{a'}(Q(s',a')) - Q(s,a)); \tag{3}$$

where each one of the parameters in (3) are explained as follows:

α is the learning rate that affects how aggressive the algorithm is in adopting a new reward value into its Q-value. A higher learning rate means the algorithm will adapt to a new environment faster. For simplicity, we choose a fixed α with non-zero value.

γ is the discount factor that affects the presence of the sum of all future rewards in the current time slot. A very small γ implies that the algorithm is not interested in the future rewards as they may be irrelevant.

ϵ is the exploration probability. ϵ-greedy approach is used in selecting an action. Let ϵ be a very small positive real number and $\epsilon \leq 1$. Then, with probability ϵ, the algorithm chooses equal-probably an action from the remaining feasible actions. With probability $1-\epsilon$, the algorithm will select the action with the highest Q-value value.

$\mathcal{R}$ is the reward, and it is a function of $\overline{N_{PT}}$, $CV_{N_{PT}}$, ΔN_{PT} and P_{ACB}. Due to the many possible combinations, we just show some of the possible state/reward combinations in Table I. In general terms, we aim to maintain ACB off when there is low occupation, and to decrease its value as traffic grows to reduce congestion. According to our observations, we consider that for N_{UL}=15, a value of $\overline{N_{PT}} \geq$10 indicates congestion, and the penalties/rewards reflect this observation. The RL-based ACB implementation is shown in Algorithm 1.

V. Experiments and Results

In this section, we evaluate the proficiency of our RL-based ACB scheme in terms of three KPIs, namely the probability to successfully complete the random access procedure, P_s; the number of preambles transmitted by the successfully accessed UEs, K, and the access delay, D.

A single cell environment is assumed to evaluate the network performance; the system accommodates both H2H and M2M UEs with different access request intensities. In order to assess the RL-based ACB scheme based on realistic H2H traffic behavior, we make use of call detail records (CDRs) obtained from a telco. Concretely, the italian operator Telecom

Table I
SOME PENALTIES AND REWARDS ASSOCIATED TO ACTIONS ON RL-ACB

Penalty	
$P_{ACB}(s') = 1$, $\overline{N_{PT}} > 10$, $CV_{Npt} \geq 0.2$, $\Delta N_{PT} > 0$	-100
$P_{ACB}(s') \geq 0.7$, $\overline{N_{PT}} > 10$, $CV_{Npt} \geq 0.2$, $\Delta N_{PT} > 0$	-90
$P_{ACB}(s') \geq 0.5$, $\overline{N_{PT}} > 10$, $CV_{Npt} \geq 0.2$, $\Delta N_{PT} > 0$	-60
$P_{ACB}(s') \geq 0.3$, $\overline{N_{PT}} > 10$, $CV_{Npt} \geq 0.2$, $\Delta N_{PT} > 0$	-50
$P_{ACB}(s') \geq 0.05$, $\overline{N_{PT}} < 7$, $CV_{Npt} \geq 0.4$, $\Delta N_{PT} > 0$	-20
Reward	
$P_{ACB}(s') = 1$, $\overline{N_{PT}} \leq 3$, $CV_{Npt} < 0.4$, $\Delta N_{PT} < 0$	100
$P_{ACB}(s') \geq 0.7$, $\overline{N_{PT}} \leq 3$, $CV_{Npt} < 0.4$, $\Delta N_{PT} < 0$	80
$P_{ACB}(s') \geq 0.5$, $\overline{N_{PT}} < 7$, $CV_{Npt} < 0.4$, $\Delta N_{PT} < 0$	40
$P_{ACB}(s') \geq 0.3$, $\overline{N_{PT}} < 7$, $CV_{Npt} < 0.4$, $\Delta N_{PT} < 0$	80
$P_{ACB}(s') \geq 0.05$, $\overline{N_{PT}} \leq 10$, $CV_{Npt} \leq 0.2$, $\Delta N_{PT} < 0$	40

Table II
RACH CONFIGURATION

Parameter	Setting
PRACH Configuration Index	*prach-ConfigIndex* $= 6$
Periodicity of RAOs	5 ms
Subframe length	1 ms
Available preambles for contention-based random access	$R = 54$
Maximum number of preamble transmissions	*preambleTransMax* $= 10$
RAR window size	$W_{\text{RAR}} = 5$ subframes
Maximum number of uplink grants per subframe	$N_{\text{RAR}} = 3$
Maximum number of uplink grants per RAR window	$N_{UL} = W_{\text{RAR}} \times N_{\text{RAR}} = 15$
Preamble detection probability for the kth preamble transmission	$P_d = 1 - \frac{1}{e^k}$ [27]
Backoff Indicator	$BI = 20$ ms
Re-transmission probability for *Msg3* and *Msg4*	0.1
Maximum number of *Msg3* and *Msg4* transmissions	5
Preamble processing delay	2 subframes
Uplink grant processing delay	5 subframes
Connection request processing delay	4 subframes
Round-trip time (RTT) of *Msg3*	8 subframes
RTT of *Msg4*	5 subframes

Algorithm 1: RL-based ACB Scheme

```
Controller: Q-learning($\mathcal{S}, \mathcal{A}, \gamma, \alpha, \epsilon, \mathcal{R}$)
Input     : $\mathcal{S}$ is the set of states, $\mathcal{A}$ is the set of actions,
            $\gamma$ is the discount factor, $\alpha$ is the learning
            rate, $\epsilon$ is the exploration probability, $\mathcal{R}$ is the
            reward
Local     : real array Q[$s$,$a$], previous state $s$, previous
            action $a$
Repeat until $i = \max RAO$
Select action $a'$ from $\mathcal{A}$ based on $\epsilon$
if $RAO(i) \mod T_{SIB2} = 0$ then
    observe reward $\mathcal{R}(s, a', s')$ and state $s'$;
    update $Q(s, a)$ by (3);
else
end
$s = s'$
```

Italia made available in 2014 a set of data from its network of the cities of Milan and Trento for what it defined as a “big data challenge” [32]. These data provides an intensity measure of data traffic for a constrained area, aggregated in periods of 10 minutes during two months (november and december of 2013). According to [33], the impact of data traffic on the RACH procedure can be 50 times higher than that of voice traffic, due mainly to the short-timed, high-frequency, low-data volume connections of apps in background mode. Although the data obtained from Telecom Italia is very useful to evaluate the temporal and geographical differences of H2H traffic for a specific service (data, voice, SMS), its values are only proportional to real measurements, and therefore, it is necessary to pre-process this data. In [33], it is stated that a base station (eNB) can support up to 55 eRAB setups per second in high load scenarios. Hence, we use this value as a reference, and normalize the original data accordingly. Since data from H2H traffic is aggregated every 10 minutes, we assume that during this period the traffic is constant. Considering H2H traffic as background traffic, we add M2M traffic in each period and evaluate a heavy-loaded scenario (30000 M2M UEs). This M2M traffic follows a Beta(3,4) distribution over 10 seconds (2000 RAOs) as described in [27]. We measure the KPIs once the M2M UEs have completed their random access procedure.

In this study, we consider the most typical PRACH configuration, *prach-ConfigIndex 6*, in conformance to the LTE-A specification [23], [27], where the subframe length is 1 ms and the periodicity of RAOs is 5 ms. Also $R = 54$ out of 64 available preambles are used for the contention-based random access and the maximum number of preamble transmissions of each UE, *preambleTransMax*, is set to 10. Table II lists additional parameters used throughout our analysis (unless otherwise stated). Although there is a high variation of traffic in H2H communications according to the day, time, or specific geographical position of the cell, its intensity is significantly smaller than that of M2M traffic. Hence, in this paper we focus on one of the most occupied cells found in the traces (cell 5161) located in the center of the city, near the Milan Cathedral at 4:20 pm, which is the time with the highest utilization on november 16.

Fig. 4 depicts the temporal distribution of UE arrivals on the above mentioned cell with a burst of M2M traffic. As it can be seen, a congestion control mechanism is necessary; besides, such a high number of preamble transmissions is the consequence of the fact that the higher the number of preamble transmissions in a RAO, the lower the probability of a successful preamble transmission. This fact, in turn, increases the probability of preamble re-transmissions in the following RAOs, hence the probability of a successful preamble transmission is further reduced. In Fig 7, we see the arrivals per RAO when the static ACB with parameters $P_{\text{ACB}} = 0.5$ and $T_{\text{ACB}} = 4$ s is implemented. These parameter values were picked based on a previous work [14] where it was identified that the combination of low values of T_{ACB} with high values of P_{ACB} leads to a reduction in the access delay; particularly, the lowest access delay for a highly congested scenario given an access success probability ≥ 0.95, is achieved when $P_{\text{ACB}} = 0.5$ and $T_{\text{ACB}} = 4$ s. However, the number of collisions is still high because the average number of preamble transmissions surpasses the RACH capacity which is 20.05 in a scenario with with 54 available preambles like this one [31].

For the experiments associated with Q-learning, unless otherwise stated, the values used for training were $\alpha = 0.15$, $\gamma = 0.7$, and $\epsilon = 0.9$. In this case, the algorithm was trained for one day (november 15) and tested on november 16 on the cell with the highest occupation. The training period was considered significant, since it represents around 6×10^5 epochs. Once the system was trained, we tested the scenario on the day mentioned earlier with different seeds for the M2M access distribution, which allowed us to test 200 different experiments. The results shown in Fig. 6 represent the mean of these 200 experiments. As it can be seen, the number of collisions was greatly reduced, and it is consistently smaller than number of successful transmissions. This is due to the fact that in our rewards/penalties system there was a strong bias towards avoiding congestion. As a result, the number of successful accesses and the number of first preamble transmissions are very close for the whole measured period. Also, the total number of preamble transmissions was considerably reduced when compared to the LTE-A system, and to the LTE-A system with static ACB. More importantly, this reduction was achieved under dynamic conditions and by adapting P_{ACB} accordingly. In Fig. 7 we can see the mean value of P_{ACB} as it adapts to different rates of UE arrivals. It can be seen that in the first RAO, P_{ACB} is equal to 1, then it quickly decreases to around 0.25 when the number of total preamble transmissions rises, but then grows again as the traffic diminishes, until it goes back to 1, where it settles. It should be noted that P_{ACB} changes dynamically with a granularity of T_{SIB2}, that is 16 RAOs. Hence, through an appropriate setting of the Q-learning parameters, it is possible to reduce collisions, although the cost is a higher delay.

In Table III, we can see different statistics for the same cell, during the same time period, for the three different schemes for access control. We separate the results for each type of service (M2M and H2H), and obtain the KPIs defined at the beginning of section V. Also, we add results corresponding to the percentiles for K and D. It is evident from the results that the solution without ACB suffers in terms of P_s and K. However, it has the smallest delay. On the other hand, our proposed Q-Learning based ACB reaches the best P_s, with practically a 100% success. This is consistent with the results seen earlier on Figure 6 and shows an improvement over the solution with fixed ACB. Also, the Q-learning solution reduces the mean number of preambles transmitted for M2M communications, which are the ones responsible for the bursty traffic. Also, our solution is able to reduce this KPI without increasing considerably the mean number of preamble transmissions for H2H traffic. This is important, because one of the main objectives when introducing M2M communications into an LTE network is that it does not affect the current users. In fact, the mean access delay for H2H users is lower for the Q-learning scheme than in the solution with fixed ACB. However, as expected, there is a trade-off, and this is reflected on an increment on the delay for M2M communications. This is expected since as it was shown in Figure 6 the collisions were considerably reduced.

VI. Conclusions

In this work, we proposed a dynamic mechanism for the setting of the ACB barring factor based on reinforcement learning, in a scenario with both M2M and H2H communications. In order to provide a more realistic analysis of this type of scenarios, the H2H traffic is obtained from CDRs. On the other hand, the M2M traffic follows the structure defined in the LTE-A specifications. The proposed solution adapts the ACB barring rate to sudden changes in traffic intensity, adjusts this traffic to the random access channel capacity consequently reducing the number of collisions and enhancing the probability of successful access. Also, our results show that although the enhancement of P_s can increase the access delay, it does not have an important impact on H2H traffic, which

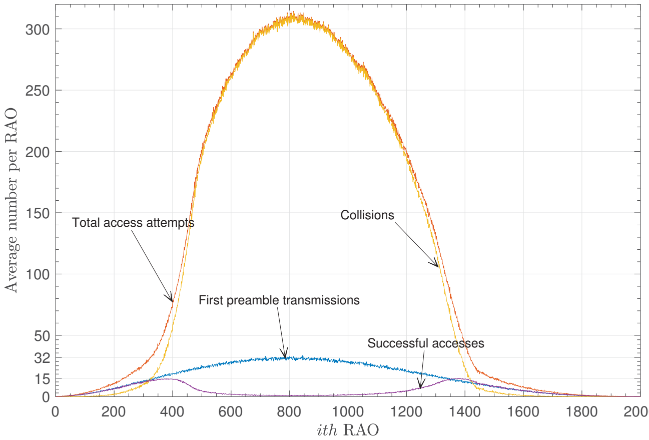

Figure 4. Temporal distribution of UE arrivals (first preamble transmissions), total preamble transmissions, collisions, and successful accesses per RAO, no access control.

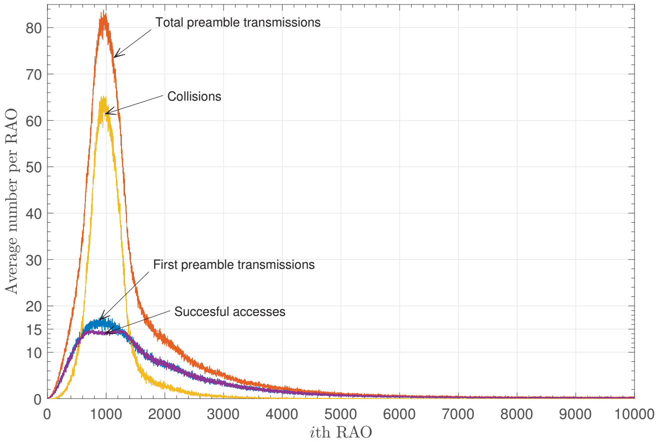

Figure 5. Temporal distribution of UE arrivals (first preamble transmissions), total preamble transmissions, collisions, and successful accesses per RAO when static ACB(0.5,4s) is implemented.

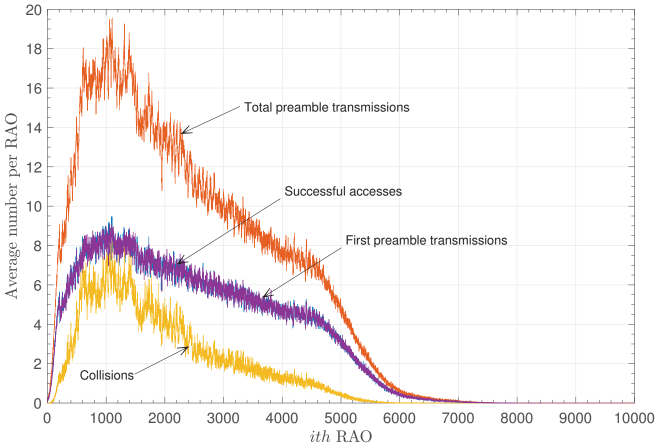

Figure 6. Temporal distribution of UE arrivals (first preamble transmissions), total preamble transmissions, collisions, and successful accesses per RAO when RL-based ACB is implemented.

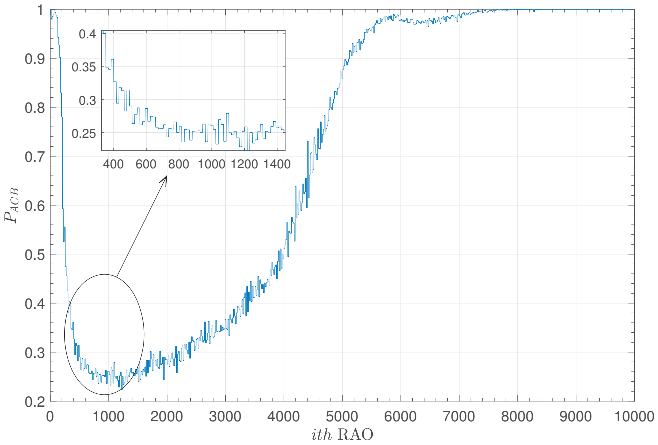

Figure 7. Adaptation of P_{ACB} as a function of time using Q-learning

Table III
KPIs Obtained when no ACB is Implemented, when static ACB(0.5,4s) is Implemented, and when Our RL-Based ACB is Implemented - Massive M2M + H2H Scenario

Key Performance Indicator		No ACB		ACB(0.5,4s)		Learning-based ACB	
		M2M	H2H	M2M	H2H	M2M	H2H
Success probability (%)	P_s	30.86	60.22	97.12	99.60	99.99	100
Number of preamble transmissions, K	$\mathbb{E}[K]$	3.46	2.38	2.49	1.56	1.85	1.62
	K_{95}	1.75	6.71	1.52	2.61	1.25	2.79
	K_{50}	1.10	1.22	1.04	0.00	1	0.00
	K_{10}	0.00	0.00	0.00	0.00	0.00	0.00
Access delay, D [ms]	$\mathbb{E}[D]$	67.94	45.61	4 155.7	3511.9	7757.6	3463.9
	D_{95}	182.25	145.13	15 839.0	13 649.0	20 924	15 164
	D_{50}	46.95	30.74	2 955	59.66	6 544	45.00
	D_{10}	15.00	15.00	17.00	15.00	17.00	15.00

is a necessary condition for the implementation of massive M2M communications. The Q-learning algorithm is aimed to reduce collisions, and therefore it has a slight impact on access delay. In the future, we intend to implement a version that focuses on optimizing this KPI while at the same time evaluating the impact that other parameters of Q-learning have over the performance.

Acknowledgment

This work has been supported in part by the Ministry of Economy and Competitiveness of Spain under Grants TIN2013-47272-C2-1-R and TEC2015-71932-REDT. The research of L. Tello-Oquendo was supported in part by Programa de Ayudas de Investigación y Desarrollo (PAID) of the Universitat Politècnica de València. The research of D. Pacheco-Paramo was supported by Universidad Sergio Arboleda, P.I. Tecnologías para la inclusión social y la competitividad económica. O.E.6.

References

[1] Ericsson. (2016, Nov.) Ericsson mobility report. [Online]. Available: https://www.ericsson.com/mobility-report

[2] Cisco. (2017, Feb.) Cisco visual networking index (VNI): Global mobile data traffic forecast update, 2016-2021. [Online]. Available: http://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/mobile-white-paper-c11-520862.html

[3] 3GPP, *TS 23.682, Architecture enhancements to facilitate communications with packet data networks and applications*, Mar 2016.

[4] F. Ghavimi and H.-H. Chen, "M2M Communications in 3GPP LTE/LTE-A Networks: Architectures, Service Requirements, Challenges, and Applications," *IEEE Commun. Surveys Tuts.*, vol. 17, no. 2, pp. 525–549, May 2015.

[5] A. Lo, Y. Law, and M. Jacobsson, "A cellular-centric service architecture for machine-to-machine (M2M) communications," *IEEE Wireless Commun. Mag.*, vol. 20, no. 5, pp. 143–151, 2013.

[6] M. S. Shafiq, L. Ji, A. X. Liu, J. Pang, A. Venkataraman, and J. Wang, "A First Look at Cellular Network Performance during Crowded Events," in *ACM SIGMETRICS/international conference on Measurement and modeling of computer systems*, June 2013.

[7] M. S. Shafiq, J. Erman, L. Ji, A. Liu, J. Pang, and J. Wang, "Understanding the Impact of Network Dynamics on Mobile Video User Engagement," in *ACM SIGMETRICS/international conference on Measurement and modeling of computer systems*, June 2014.

[8] H. Zang and J. Bolot, "Mining call and mobility data to improve paging efficiency in cellular networks," in *Proceedings of the 13th annual ACM international conference on Mobile computing and networking*, September 2007.

[9] 3GPP, *TS 36.211, Physical Channels and Modulation*, Dec 2014.

[10] ——, *TS 36.331, Radio Resource Control (RRC), Protocol specification*, Sep 2017.

[11] D. C. Chu, "Polyphase codes with good periodic correlation properties," *IEEE Trans. Inf. Theory*, vol. 18, 1972.

[12] L. Ferdouse, A. Anpalagan, and S. Misra, "Congestion and overload control techniques in massive M2M systems: a survey," *Trans. Emerg. Telecommun. Technol.*, vol. 25, no. 3, pp. 1–17, Mar 2015.

[13] A. Laya, L. Alonso, and J. Alonso-Zarate, "Is the Random Access Channel of LTE and LTE-A Suitable for M2M Communications? A Survey of Alternatives," *IEEE Commun. Surveys Tuts.*, vol. 16, no. 1, pp. 4–16, Jan 2014.

[14] I. Leyva-Mayorga, L. Tello-Oquendo, V. Pla, J. Martinez-Bauset, and V. Casares-Giner, "Performance analysis of access class barring for handling massive M2M traffic in LTE-A networks," in *Proc. IEEE International Conference on Communications (ICC)*, May 2016, pp. 1–6.

[15] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. MIT press Cambridge, 1998, vol. 1, no. 1.

[16] A. Lo, Y.-W. Law, M. Jacobsson, and M. Kucharzak, "Enhanced lte-advanced random-access mechanism for massive machine-to-machine (m2m) communications." 2011.

[17] R.-H. Hwang, C.-F. Huang, H.-W. Lin, and J.-J. Wu, "Uplink access control for machine-type communications in lte-a networks." vol. 20, Nov 2016.

[18] 3GPP, *TS 22.011, V15.1.0, Service Accessibility*, June 2017.

[19] M. Tavana, V. Shah-Mansouri, and V. W. S. Wong, "Congestion control for bursty M2M traffic in LTE networks," in *Proc. IEEE International Conference on Communications (ICC)*, Jun 2015, pp. 5815–5820.

[20] S. Duan, V. Shah-Mansouri, Z. Wang, and V. W. S. Wong, "D-ACB: Adaptive Congestion Control Algorithm for Bursty M2M Traffic in LTE Networks," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 12, pp. 9847–9861, 2016.

[21] L. M. Bello, P. Mitchell, and D. Grace, "Application of Q-Learning for RACH Access to Support M2M Traffic over a Cellular Network," in *Proc. 20th European Wireless Conference*, May 2014.

[22] J. Moon and Y. Lim, "A Reinforcement Learning Approach to Access Management in Wireless Cellular Networks," *Wireless Communications and Mobile Computing*, May 2017.

[23] 3GPP, *TS 36.321, Medium Access Control (MAC) Protocol Specification*, Sept 2012.

[24] A. Biral, M. Centenaro, A. Zanella, L. Vangelista, and M. Zorzi, "The challenges of M2M massive access in wireless cellular networks," *Digit. Commun. Netw.*, vol. 1, no. 1, pp. 1–19, 2015.

[25] 3GPP, *TS 36.213, Physical layer procedures*, Dec 2014.

[26] ——, *TR 36.912, Feasibility study for Further Advancements for E-UTRA*, Apr 2011.

[27] ——, *TR 37.868, Study on RAN Improvements for Machine Type Communications*, Sept 2011.

[28] T. M. Lin, C. H. Lee, J. P. Cheng, and W. T. Chen, "PRADA: Prioritized random access with dynamic access barring for MTC in 3GPP LTE-A networks," *IEEE Trans. Veh. Technol.*, vol. 63, no. 5, pp. 2467–2472, 2014.

[29] O. Arouk and A. Ksentini, "General Model for RACH Procedure Performance Analysis," *IEEE Commun. Lett.*, vol. 20, no. 2, pp. 372–375, Feb 2016.

[30] Z. Zhang, H. Chao, W. Wang, and X. Li, "Performance Analysis and UE-Side Improvement of Extended Access Barring for Machine Type Communications in LTE," in *Proc. IEEE Vehicular Technology Conference (VTC Spring)*, May 2014, pp. 1–5.

[31] R. G. Cheng, J. Chen, D. W. Chen, and C. H. Wei, "Modeling and analysis of an extended access barring algorithm for machine-type communications in LTE-A Networks," *IEEE Trans. Wireless Commun.*, vol. 14, no. 6, pp. 2956–2968, 2015.

[32] Telecomitalia. (2016, Nov.) Telecom italia: Big data challenge. [Online]. Available: http://www.telecomitalia.com/tit/en/innovazione/archivio/big-data-challenge-2015.html

[33] Nokia, "Mobile Broadband solutions for Mass Events," Nokia, Tech. Rep., 2014.