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 machine-to-machine (M2M) connectivity offers numerous advantages regarding coverage, deployment costs, security and management, among others. Nevertheless, having a large number of M2M devices activated simultaneously is difficult to tackle at the evolved Node B, and it causes complications in the connection establishment. The random access channel (RACH) in LTE-A is adequate for handling human-to-human (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 users contending for access. However, it is still not clear how to dynamically adapt its parameters, especially in highly changing scenarios with bursty traffic as it can occur when M2M communications are involved. We propose a dynamic algorithm based on reinforcement learning 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 key performance indicators.**

***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 essential technology for the upcoming generation of wireless systems due to its capacity to provide connectivity to 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 (10^{18} 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. Furthermore, features like widely deployed infrastructure, global connectivity, high quality of service (QoS), well-developed charging and security solutions, among others can be exploited in M2M applications [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 the QoS in 3G networks [6], [7], or location management optimization [8], among others.

In LTE-A, when a user equipment (UE) wants 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 [9], [10]. The evolved Node B (eNB) has a number of preambles (signatures generated by Zadoff-Chu sequences due to their good correlation properties [9], [11]) available for initial access to the network. The UEs transmit a preamble for attempting the first access; further details are explained in Section III-A.

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

There is a trade-off 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 to the traffic intensity is critical, but the 3GPP does not specify any particular algorithm for that purpose. In this work, we propose a reinforcement learning (RL) approach to tune dynamically the ACB barring rate; concretely, we use Q-learning, a well-known RL technique [15].

Our main contributions are summarized as follows.

- A Q-learning algorithm [15] is designed to dynamically and autonomously tune the ACB barring rate such that it can rapidly react to the traffic changes using local information available at the eNB.
- Our proposed scheme is intelligent and it conforms with 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 studies dealing with the optimization of the ACB scheme in LTE/LTE-A networks using either static or adaptive approaches. However, most of these studies require considerable modifications of 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 generates signaling load. In [17], a dynamic mechanism for access control in LTE-A is proposed to reduce the impact that massive M2M communications have on H2H traffic. Also, in this work they differentiate M2M traffic, allowing prioritization. However, this approach modifies ACB so that it can 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 this number is not known a priori, any mechanism that aims at optimizing the ACB has to develop an estimation of this value. In [19], a dynamic scheme for ACB is proposed. It is based on 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 traffic type was considered. Also in [20], an optimal value of the P_{ACB} parameter is obtained in an ideal case, i.e., assuming the eNB has all the information about the system. Some heuristics which resemble this optimal solution are provided as well; one of them changes the parameter P_{ACB} and the other changes both P_{ACB} and the number of preambles that can be acknowledged. 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 RL scheme is performed only on the M2M UEs to allocate the random access slot on which they should transmit for avoiding collisions. 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 at adapting 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 and the Access Class Barring in Section III-A and Section III-B, respectively.

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. Before initiating the random access procedure, the UEs must first obtain some 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*).

Two uplink channels are required for the access attempts, namely, the physical random access channel (PRACH) for preamble transmission and the physical uplink shared channel (PUSCH) for additional signaling data. 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. 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 [9], [10].

As mentioned above, 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

Once the UE has acquired the basic configuration parameters, 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] 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 [26] and with most of the literature [14], [19], [27]–[30].

Random access response—RAR—(*Msg2*): The eNB computes an identifier for each successfully decoded preamble,

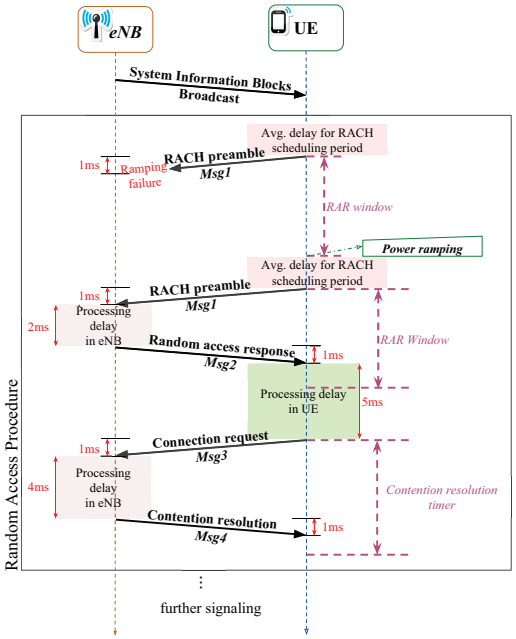

Figure 1. LTE-A contention-based random access procedure.

ID = 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 this 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 (*preambleTransMax* notified by the eNB through the SIB2 [10]) without success, the network is declared unavailable by the UE, an access problem is indicated to upper layers, and the random access procedure is terminated.

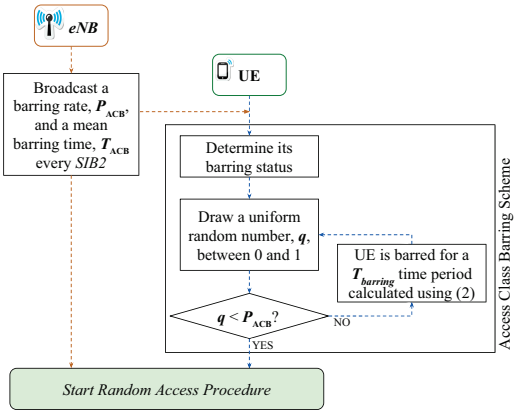

Figure 2. Access class barring scheme.

Backoff procedure: According to the LTE-A standard [23], if the random access attempt of a UE fails, regardless of the cause, the UE has to perform a backoff procedure before starting the random access process all over again. 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 failed the access 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 UEs. For doing so, all UEs are classified into 16 groups defined as access classes (ACs) 0 to 15. The AC number is stored in UE's SIM/USIM. 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 random access 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 random access 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

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

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.

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. This 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 taking a given action a being on a given state s and then continuing indefinitely by taking actions optimally. In the following, we introduce the arguments and parameters that define $Q(s,a)$.

Let $\mathcal{A} = \{1, 2, .., 16\}$ be the set of actions that change P_{ACB} to one of its possible values defined in Section III-B. When the chosen action is $a = 1$, then $P_{\text{ACB}} = 0.05$, and the rest of the values are mapped sequentially. The ACB mechanism is turned off when $P_{\text{ACB}} = 1$ (i.e., $a = 16$). Due to the characteristics of ACB, changes on P_{ACB} can only be received by UEs through SIB2 messages that are broadcast every T_{SIB2}. Hence, the Q-learning actions that change P_{ACB} will only be taken before the transmission of a SIB2. Following the specifications [10], throughout this work we will use a value of $T_{\text{SIB2}} = 16$ RAOs (i.e., 80 ms). Let s be the state defined as $s = \left(\overline{N_{PT}}, CV_{N_{PT}}, \Delta N_{PT}, P_{\text{ACB}}\right)$, where $\overline{N_{PT}} = \frac{1}{T_{\text{SIB2}}}\left(\sum_{k=1}^{T_{\text{SIB2}}} N_{PT_k}\right)$ is the mean number of preamble transmissions per RAO that the eNB detected in a whole T_{SIB2} (actually a truncated version as explained below), $CV_{N_{PT}} = \left[\frac{1}{T_{\text{SIB2}}-1}\left(\sum_{k=1}^{T_{\text{SIB2}}} \mid N_{PT_k} - \overline{N_{PT}} \mid^2\right)\right]^{\frac{1}{2}} / \overline{N_{PT}}$ is the variation coefficient of N_{PT} for the same period, ΔN_{PT} is the difference between the mean number of preamble transmissions in the current period and in the previous one, and P_{ACB} is the ACB probability that affected UEs during this period.

Fig. 3 illustrates the above state definition. In time $t = n-1$, 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 T_{SIB2} will depend on this information. At time $t = n$, 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} can be greater than $W_{\text{RAR}} \times N_{\text{RAR}}$, 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, in our experiments we observed that even in low congested scenarios, it was very unlikely that $\overline{N_{PT}}$ would exceeded 30 (we omit these results due to the lack of space). Therefore, considering that these scenarios might be 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, likewise based on our observations, the coefficient of variation values

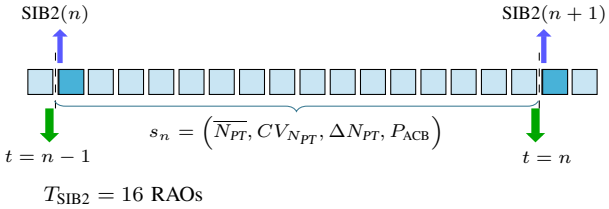

Figure 3. State definition and T_{SIB2}.

$CV_{N_{PT}} \in \{0, 0.2, 0.4, 0.6, 0.8\}$ were discretized to reduce the total number of states as follows. If $0 \leq CV_{N_{PT}} \leq 0.19$ over the corresponding T_{SIB2}, then 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 $\Delta N_{PT}(n) > 0$, then it will be set as 1; if $\Delta N_{PT}(n) = 0$, then it will be set as 2; if $\Delta N_{PT}(n) < 0$, then it will be set as 3. Finally, $P_{\text{ACB}}(n)$ is the barring factor that affected UEs during the considered period, that is, the value sent in SIB2(n).

The Q-value is updated according to the Q-function [15]:

$$Q(s,a) = Q(s,a) + \alpha \left[\mathcal{R} + \gamma \max_{a' \in \mathcal{A}} [Q(s',a')] - Q(s,a)\right]; \quad (3)$$

where α, $\mathcal{R}$, γ 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.
- $\mathcal{R}$ 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 (i.e., $P_{\text{ACB}} = 1$) when there is low occupation, and to decrease the P_{ACB} value as traffic grows to reduce congestion. According to our observations, we consider that for $N_{UL} = 15$ uplink grants, a value of $\overline{N_{PT}} \geq 10$ indicates congestion, and the rewards reflect this observation.
- 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 relevance of future rewards in the algorithm is low compared with current ones.

An ϵ-greedy approach is used in selecting an action. Let ϵ be the exploration probability, $0 \leq \epsilon \leq 1$. Then, with probability ϵ, the algorithm chooses equal-probably an action from the remaining feasible actions (exploration). With probability $1-\epsilon$, the algorithm will select the action with the highest $Q(s,a)$ value (exploitation). This is a trade-off between exploration and exploitation, where a higher ϵ will encourage more aggressive exploration for potentially better but yet-to-be-known action for a given state. In our experiments, the value of ϵ followed a linear function that went from 1 to 0. 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

Table I
SOME EXAMPLES OF REWARDS ASSOCIATED TO ACTIONS ON RL-ACB

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

Algorithm 1: RL-based ACB Scheme

Controller: Q-learning($\mathcal{S}, \mathcal{A}, \alpha, \mathcal{R}, \gamma, \epsilon$)
Input : $\mathcal{S}$ is the set of states, $\mathcal{A}$ is the set of actions, α is the learning rate, $\mathcal{R}$ is the reward, γ is the discount factor, ϵ is the exploration probability
Local : real array $\mathbf{Q}[s, a]$, previous state s, previous action a

repeat
  if *RAO*$(i) \mod T_{SIB2} = 0$ **then**
    select action a' from $\mathcal{A}$ based on ϵ;
    observe reward $\mathcal{R}(s, a', s')$ and state s';
    update $Q(s, a)$ by (3);
  else
  end
  $s = s'$
until $i = \max$ *RAO*;

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. The italian operator Telecom 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* [31]. This 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). This data is very useful to evaluate the temporal and geographical distributions of H2H traffic for a specific service (data, voice, SMS). According to [32], 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. Therefore, it is necessary to pre-process this data. Also in [32], it is stated that a base station (eNB) can support up to 55 EUTRAN radio access bearer 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 (30 000 M2M UEs). This M2M traffic follows a Beta(3,4) distribution over 10 seconds (2000 RAOs) as described in [26]. We measure the KPIs once the M2M UEs have completed their random access procedure.

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	$P_d = 1 - \frac{1}{e^k}$ [26]
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

In this study, we consider the typical PRACH configuration, *prach-ConfigIndex 6*, in conformance to the LTE-A specification for these kind of studies [23], [26], 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 per 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 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, in turn, increases the probability of preamble re-transmissions in the following RAOs, hence the probability of a successful preamble transmission is further reduced.

Fig. 5 shows 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

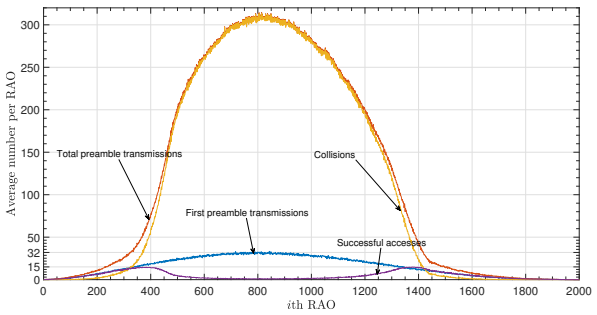

Figure 4. Temporal distribution of UE arrivals (first preamble transmissions), total preamble transmissions, collisions, and successful accesses per RAO, no access control implemented.

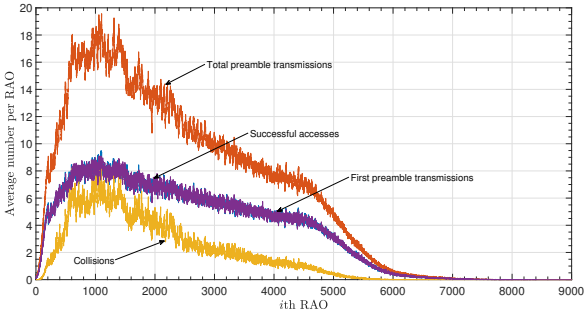

Figure 6. Temporal distribution of UE arrivals (first preamble transmissions), total preamble transmissions, collisions, and successful accesses per RAO when RL-based ACB is implemented.

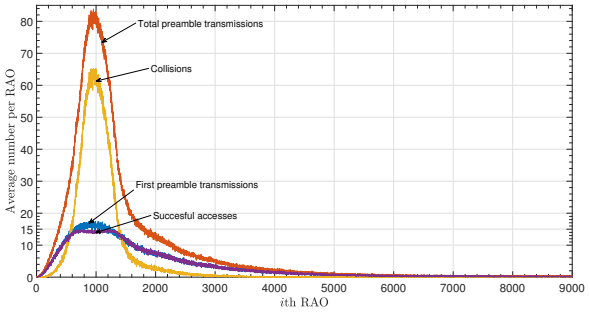

Figure 5. Temporal distribution of UE arrivals (first preamble transmissions), total preamble transmissions, collisions, and successful accesses per RAO when static ACB(0.5,4 s) is implemented.

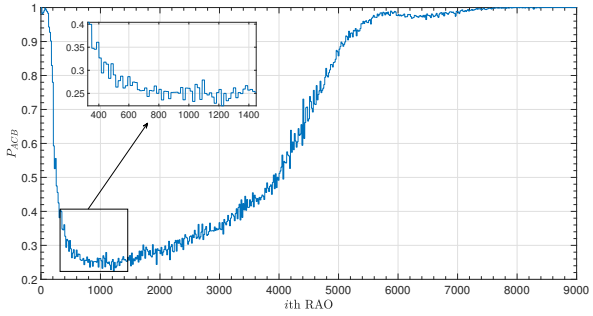

Figure 7. Adaptation of P_{ACB} as a function of time using Q-learning.

highly congested scenario given a requirement of $P_s \geq 0.95$ is achieved when $P_{\text{ACB}} = 0.5$ and $T_{\text{ACB}} = 4\,\text{s}$. 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 54 available preambles like this one [14], [27].

For the experiments associated with Q-learning, the algorithm was trained for one day (November 15); the parameter values used for this period were $\alpha = 0.15$, $\gamma = 0.7$, and a linear function from 1 to 0 for ϵ. We consider this training period significant since it represents around 6×10^5 epochs. Then, we tested the algorithm on November 16 on the cell with the highest occupation; we used 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 can be seen, the number of collisions was greatly reduced and it is consistently smaller than the number of successful transmissions. This is due to the fact that in our rewards 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 without access control, and to the LTE-A system with static ACB. More importantly, this reduction was achieved under dynamic conditions and by adapting P_{ACB} accordingly.

Fig. 7 showcases 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 two different access control schemes. 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 network without ACB suffers in terms of P_s and K. However, it has the smallest delay. On the other hand, our proposed ACB scheme based on Q-learning reaches the best P_s, with practically a 100% success. This is consistent with the results seen earlier on Fig. 6 and shows an improvement over the solution with static 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. Furthermore, our solution is able to reduce this KPI without considerably increasing the mean number of preamble transmissions for H2H traffic. This is important since one of the main requirements when introducing M2M communications into an LTE-A network is that it does not affect the preexisting H2H UEs. In fact, the mean access delay for H2H users is lower for the Q-learning scheme than in the solution with static ACB. However, as expected, there is a trade-off,

Table III
KPIs OBTAINED FOR LTE-A AND DIFFERENT ACB IMPLEMENTATIONS. MASSIVE M2M + H2H TRAFFIC

Key Performance Indicator		No ACB		ACB(0.5, 4 s)		RL-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	2.36	2.49	1.57	1.85	1.62
	K_{95}	8.58	6.71	6.31	2.63	6.17	2.71
	K_{50}	1.99	1.19	1.42	1.07	1.00	1.00
	K_{10}	1.00	1.00	1.00	1.00	1.00	1.00
Access delay, D [ms]	$\mathbb{E}[D]$	67.94	45.01	4 162.9	3512.5	7657.6	3463.9
	D_{95}	182.70	144.63	15 839.0	13 650.0	19 924.0	15 164.0
	D_{50}	47.10	30.14	2 955.0	59.90	6 534.0	45.00
	D_{10}	17.85	16.80	21.30	16.80	17.98	16.80

and this is reflected on an increment on the delay for M2M communications. This is expected since as it was shown in Fig. 6, the collisions were considerably reduced.

VI. CONCLUSIONS

We have proposed a dynamic mechanism based on reinforcement learning for tuning the ACB barring factor. We consider a hybrid scenario with both M2M and H2H communications. To provide a more realistic analysis of this type of scenarios, we used CDRs to model the H2H traffic. On the other hand, the M2M traffic is modeled according to the LTE-A specifications. Besides adapting the ACB barring rate to sudden changes in traffic intensity, the proposed solution adjusts this traffic to the random access channel capacity consequently reducing the number of collisions and enhancing the probability of successful access, P_s. Also, our results show that although the enhancement of P_s can increase the access delay, it does not have a significant impact on H2H traffic, which is a necessary condition for the implementation of massive M2M communications. The Q-learning algorithm is aimed at reducing collisions, and therefore it has a slight impact on access delay. In follow-up work, we will implement an algorithm 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. (2017, Nov.) Ericsson mobility report. [Online]. Available: https://www.ericsson.com/mobility-report

[2] Cisco. (2017, Mar.) 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*, Sep 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] L. Tello-Oquendo, I. Leyva-Mayorga, V. Pla, J. Martinez-Bauset, J. R. Vidal, V. Casares-Giner, and L. Guijarro, "Performance Analysis and Optimal Access Class Barring Parameter Configuration in LTE-A Networks with Massive M2M Traffic," *IEEE Trans. Veh. Technol.*, vol. PP, no. 99, pp. 1–1, 2017. DOI 10.1109/tvt.2017.2776868.

[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," *Personal and Ubiquitous Computing*, vol. 20, no. 6, pp. 851–862, 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 Trans. Veh. Technol.*, 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] ——, *TS 36.213, Physical layer procedures*, Dec 2014.

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

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

[27] 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.

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

[29] 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.

[30] 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.

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

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