

# Dynamic Access Class Barring Parameter Tuning in LTE-A Networks with Massive M2M Traffic

Luis Tello-Oquendo, José-Ramón Vidal, Vicent Pla, Luis Guijarro

► **To cite this version:**

Luis Tello-Oquendo, José-Ramón Vidal, Vicent Pla, Luis Guijarro. Dynamic Access Class Barring Parameter Tuning in LTE-A Networks with Massive M2M Traffic. 17th Annual Mediterranean Ad Hoc Networking Workshop (Med-Hoc-Net 2018), Jun 2018, Capri Island, Italy. pp.63-70. hal-01832531

**HAL Id: hal-01832531**

**<https://hal.inria.fr/hal-01832531>**

Submitted on 8 Jul 2018

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



# Dynamic Access Class Barring Parameter Tuning in LTE-A Networks with Massive M2M Traffic

Luis Tello-Oquendo, José-Ramón Vidal, Vicent Pla, and Luis Guijarro

Instituto ITACA. Universitat Politècnica de València, Valencia 46022, Spain

**Abstract**—Machine-to-machine (M2M) communication is one of the leading facilitators of the Internet of Things environment by offering ubiquitous applications and services. Using cellular networks for providing M2M connectivity brings several advantages such as extended coverage, security, robust management, and lower deployment costs; however, coexistence with a large number of M2M devices is still an essential challenge in LTE-A networks, in part due to the difficulty in allowing simultaneous access. Although the random access procedure in LTE-A is adequate for human-to-human (H2H) communications, when M2M communications are involved congestion control is required. One of the congestion control schemes suggested by the 3GPP is Access Class Barring (ACB), which can reduce the number of simultaneous users contending for access. However, it is still not clear how to adapt ACB parameters in bursty and heavy-loaded scenarios, such as those that appear in M2M communications. We propose a dynamic ACB scheme in which an estimate of the current number of M2M devices in backoff state is used to adjust in real-time the barring rate parameter. We evaluate the key performance indicators (KPIs) of dynamic ACB in several scenarios with different degrees of traffic load and compare them with those of a static ACB with optimal parameters. We show that the dynamic ACB outperforms the static one offering shorter access delay and higher successful access probability, while its impact on H2H communications KPIs is negligible. Besides, our proposed scheme conforms with the LTE-A specification so that it can be included as a viable solution.

**Index Terms**—Access class barring (ACB); cellular-systems; machine-to-machine communications; performance analysis; 5G.

## I. INTRODUCTION

INTERNET of Things (IoT) is emerging as one of the key transforming technologies to interconnect physical objects that interact with people, other physical objects, and systems to benefit society in unprecedented ways. It is predicted that the number of connected devices increases to 29 billion by 2022 [1], and the global mobile data traffic achieves 49 exabytes ( $10^{18}$  bytes) by 2021 [2]. Machine-to-machine (M2M) communication is one of the leading facilitators of the IoT environment by offering ubiquitous applications and services.

Unlike human-to-human (H2H) communication, distinct features of M2M traffic requires specialized and inter-operable communication technologies. Cellular networks are the natural choice to satisfy these requirements and handling a significant part of this emerging traffic due to their already existing infrastructures, extensive area coverage, and high-performance capabilities. Therefore, M2M communication over cellular

This research 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.

ISBN 978-3-903176-05-8 © 2018 IFIP

networks is no longer a matter of choice; there are already M2M subscriptions supported by cellular operators worldwide, enhanced techniques are currently under standardization by 3GPP, and M2M connectivity is driving the development of 5G cellular systems [3]–[5].

Handling the massive number of connections generated by a large number of M2M devices (UEs) is an important challenge in cellular networks that has received a significant amount of attention nowadays. Specifically, the signaling in the connection attempts and the limited resources of the physical channels in the random access procedure are the main factors to take into account. Furthermore, 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.

For avoiding that the above challenges become limitations that hinder the widespread horizons of M2M applications, the access class barring (ACB) scheme is included in the LTE-A radio resource control specification [6] 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.

There is a trade-off between relieving congestion and the key performance indicators (KPIs) of the network when the ACB is operating [7]. Thus, its parameters must be adjusted adequately. The proper tuning of ACB parameters according to the traffic intensity is critical; however, how to implement this scheme effectively and tuning their parameter configuration so that it can dynamically and autonomously adapt to the network load is not a trivial task and has not been specified. The dynamic adaptation of the barring rate is desirable since the implementation of a static ACB affects the access delay of every UE, even in cases of no congestion, when the scheme is not needed at all.

The aim of this paper is to provide an algorithm for the dynamic tuning of the barring rate according to the traffic load in the random access channel (RACH) and its capacity. Our solution uses a comprehensive view of the access attempts in the random access procedure and fits the incoming traffic to the capacity of the RACH by tuning the barring rate appropriately. Furthermore, it conforms with current system specifications allowing efficient congestion control and facilitating the coexistence of H2H and M2M traffic.

Specifically, the main contributions of this paper are the following:

- We devise both an estimator of the UEs that are in backoff state and a mechanism for fitting this traffic jointly with the new arrivals into the RACH capacity.

- We propose an algorithm for tuning the barring rate dynamically according to the traffic load based on the devised estimator. We closely follow the 3GPP directives [8]–[10] so that our proposed solution is aligned with the network specifications and can be successfully integrated in the system.
- Besides testing several traffic loads following the traffic model and configuration parameters for M2M communications suggested by the 3GPP [11], we evaluate the impact on the H2H traffic. We assess our proposed algorithm regarding the key performance indicators (KPIs) suggested by the 3GPP [11] for this kind of studies.

The rest of the paper is organized as follows. In Section II, a review of the literature regarding ACB is conducted. Then, the random access procedure in LTE-A and the ACB are presented in Section III-A and Section III-B, respectively. A detailed explanation of the proposed dynamic ACB jointly with its procedures for its proper functioning is presented in Section IV. The performance evaluation, selected traffic model, parameter configuration, KPIs, and our most relevant results are presented in Section V. Finally, our conclusions are presented in Section VI.

## II. RELATED WORK

There have been a number of research efforts devoted to optimizing ACB for handling massive M2M connection attempts on the RACH through either static or dynamic approaches. However, many of them offer complex procedures, use questionable assumptions for getting high performance, or do not conform with LTE-A specifications (e.g., without considering the updating period of notification information by the eNodeB) [12]–[14].

In [15], 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 3GPP specifications, the authors assume that a control-loop for congestion between the UEs and the eNodeB is available, which creates signaling overhead. In [13], a dynamic mechanism for access control in LTE-A is proposed, to reduce the impact that massive M2M communications can have on H2H traffic. Also, in this work the authors classify the traffic, allowing prioritization. However, this approach modifies ACB so that it can send different parameters for different classes, in a similar way to the extended access barring scheme. 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 at optimizing ACB has to develop an estimation of this value. In [16], a dynamic scheme for adjusting the ACB factor is proposed. It is based on the estimation of the contending UEs in the RACH using Kalman filtering. This solution enhances the overall performance but assuming full state information of the system. 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 [17], an optimal value of the barring rate parameter is obtained in an ideal case, i.e., assuming full state information about the system. Some heuristics which resemble this optimal solution are provided as well; one of them changes the barring rate and the other changes both the barring rate 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 random access opportunity, which is not consistent with the LTE-A specifications.

To successfully address the above limitations, we propose a dynamic ACB based on the estimation of UEs that are in backoff state and a mechanism for fitting this traffic jointly with the incoming one into the RACH capacity. Our solution tunes the barring rate according to the traffic load in real-time and deactivates itself when the traffic is below the RACH capacity. It conforms with current system specifications and constraints, offering effective access control.

## 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 ACB 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 the 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 [8].

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 eNodeB 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 ( $6 \times 180$  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 eNodeB; 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].

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 eNodeBs 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.

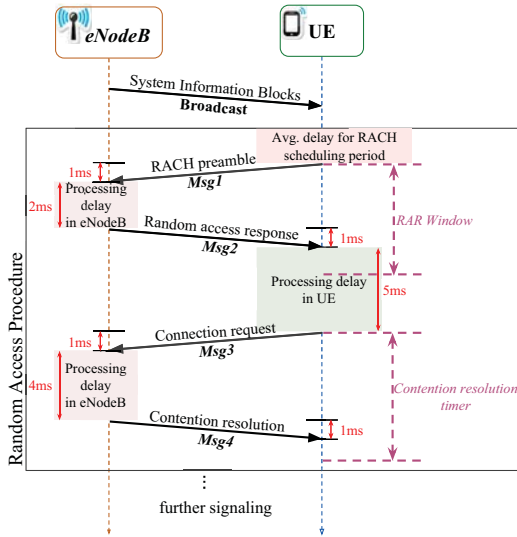


Figure 1. LTE-A contention-based random access procedure.

### 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 [8]–[10], [18] 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 eNodeB in the same RAO, using different preambles. The eNodeB 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 [11] and with most of the literature [7], [16], [19], [20].

**Random access response—RAR—( $Msg2$ ):** The eNodeB computes an identifier for each successfully decoded preamble,  $ID = f(\text{preamble}, \text{RAO})$ , 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 ( $B$ ), and the assignment of a temporary identifier.

Exactly two subframes after the preamble transmission has ended (this is the time needed by the eNodeB to process the received preambles), the UE begins to wait for a time window,  $W_{RAR}$ , to receive an uplink grant from the eNodeB 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 eNodeB through the SIB Type 2 (SIB2) [9]. 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 eNodeB using the reserved PUSCH resources; hybrid automatic repeat request (HARQ) is used to protect this message transmission.

**Contention Resolution ( $Msg4$ ):** The eNodeB transmits  $Msg4$  as an answer to  $Msg3$ . The eNodeB 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 value of  $B$  received with  $Msg2$ .

Note that each UE keeps track of its preamble transmissions. When a UE has transmitted a certain number of preambles ( $\text{preambleTransMax}$  notified by the eNodeB through the SIB2 [9]) 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.

**Backoff procedure:** According to the LTE-A standard [8], 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

$$T_{BO} = \mathcal{U}(0, B), \quad (1)$$

where  $\mathcal{U}(\cdot)$  stands for uniform distribution,  $B$  is the backoff indicator defined by the eNodeB, and its value ranges from 0 to 960 ms. The value of  $B$  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 parameter  $B$ .

### 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. 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 RACH, which might jeopardize the accomplishment of QoS objectives. For that purpose, 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 UEs may be assigned an AC between 0 and 9, and if a higher priority is needed, other classes may be used.

If ACB is not implemented, all ACs are allowed to access the RACH. When ACB is implemented, the eNodeB broadcasts (through SIB2) barring rates,  $P_{ACB} \in \{0.05, 0.1, \dots, 0.3, 0.4, \dots, 0.7, 0.75, 0.8, \dots, 0.95\}$ , and mean barring times,  $T_{ACB} \in \{4, 8, 16, \dots, 512\text{ s}\}$ , 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 by the eNodeB. For this, the UE generates a random number denoted by  $g$ ,  $g = \mathcal{U}(0, 1)$ . If  $g \leq P_{ACB}$ , the UE selects and transmits its preamble. Otherwise, the UE waits for a random time calculated as follows

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



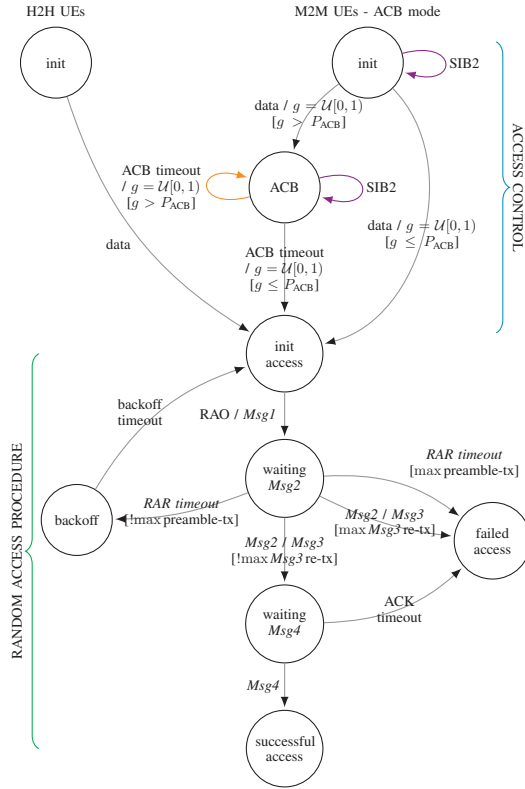


Figure 2. State transition diagram of the random access procedure when access control is implemented for M2M UEs.

Fig. 2 illustrates the state transition diagram of the ACB mechanism [6], [9] jointly with the random access procedure. Note that ACB is applied to the UEs before they perform the random access procedure explained in Section III-A. 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. DYNAMIC ACCESS CLASS BARRING FOR CONGESTION CONTROL

The ultimate goal of the ACB is to control the arrival of M2M UEs to the RACH so that the number of transmitted preambles stays below a certain threshold. To achieve this, the eNodeB must tune in real-time the  $P_{ACB}$  value accordingly to the traffic conditions. We propose to implement the  $P_{ACB}$  updating based on an estimate of the number of UEs that are in backoff state. In Section IV-A we derive analytically the estimation of the number of UEs in backoff state and in Section IV-B we describe the mechanism for tuning the  $P_{ACB}$  dynamically.

##### A. Estimation of the number of UEs in backoff state

At  $RAO(i)$ , let  $n_b(i)$  be the estimate number of UEs in backoff state,  $n_s(i)$  the number of UEs that accessed successfully,  $n_t(i)$  the number of UEs that transmitted a preamble and  $n_r(i)$  the number of UEs in backoff state that retransmitted. The estimated number of UEs in backoff state at  $RAO(i)$  (see Fig. 3) is given by

$$n_b(i) = n_b(i-1) - n_r(i) + n_t(i) - n_s(i). \quad (3)$$

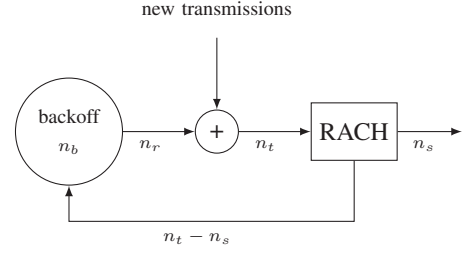


Figure 3. Variables and traffic flows used in the estimation of  $n_b$ .

UEs that fail the access, mainly because they reach the *preambleTransMax* value, are not considered in Eq. (3). Therefore, the estimation presented here is valid only when the success probability is high enough, which is a reasonable assumption when congestion control performs effectively, as we show in the following sections.

The eNodeB knows  $n_s(i)$  by observing the preambles that have been detected and decoded correctly. Since the number of uplink grants that the eNodeB can send in an RAO is limited to  $N_{UL}$  grants,  $n_s(i)$  is given by

$$n_s(i) = \min\{\text{preambles decoded}, N_{UL}\}. \quad (4)$$

The value of  $n_t(i)$  is unknown because, in our collision model, the preambles that collided (transmitted by more than one UE) are not decoded, and therefore the eNodeB does not know how many UEs transmitted them. Instead, we will use the estimated expectation of  $n_t$ , obtained from the number of used preambles  $n_u$  (preambles transmitted at least by one UE). Let  $Y_j \in \{0, 1\}$  be the random variable that denotes the transmission of preamble  $j$  given that the total number of transmissions is  $n_t$ . Then,  $Y_j = 0$  when preamble  $j$  has not been transmitted by any UE, and  $Y_j = 1$  otherwise. The probabilities of  $Y_j$  are

$$\begin{cases} \mathbb{P}\{Y_j = 0\} = \left(1 - \frac{1}{R}\right)^{n_t}, \\ \mathbb{P}\{Y_j = 1\} = 1 - \left(1 - \frac{1}{R}\right)^{n_t}, \end{cases} \quad (5)$$

where  $R$  is the number of available preambles in the system. The expected value of  $Y_j$  is given by

$$\mathbb{E}\{Y_j\} = 0 \cdot \mathbb{P}\{Y_j = 0\} + 1 \cdot \mathbb{P}\{Y_j = 1\} = 1 - \left(1 - \frac{1}{R}\right)^{n_t}. \quad (6)$$

Then, the number of used preambles,  $n_u$ , is

$$n_u = \sum_{j=0}^R Y_j, \quad (7)$$

and its expected value is

$$\mathbb{E}\{n_u\} = \mathbb{E}\left\{\sum_{j=0}^R Y_j\right\} = R \left[1 - \left(1 - \frac{1}{R}\right)^{n_t}\right]. \quad (8)$$

If we assume that  $\mathbb{E}\{n_u\}$  changes slowly, it can be estimated from a short term time average of  $n_u$ . Let  $\hat{n}_u(i)$  be an estimate of the time average of  $n_u$  obtained by exponential smoothing of  $n_u$  at  $RAO(i)$ . Then,

$$\mathbb{E}\{n_u\}(i) \approx \hat{n}_u(i) = \alpha \hat{n}_u(i-1) + (1 - \alpha) \hat{n}_u(i) \quad (9)$$

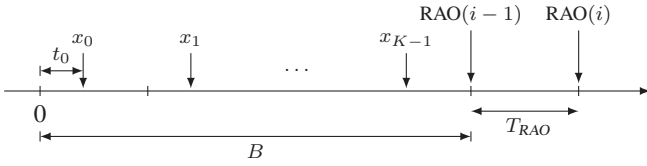


Figure 4. Backoff starting times.

with  $\alpha < 1$ , and from Eq. (8),  $n_t(i)$  is

$$n_t(i) = \frac{\log\left(1 - \frac{\mathbb{E}\{n_u\}(i)}{R}\right)}{\log\left(1 - \frac{1}{R}\right)}. \quad (10)$$

The value of  $n_r(i)$  is estimated from  $n_b(i-1)$ . Firstly, from  $n_b(i-1)$  we estimate the expected number of UEs that entered the backoff state during the period of length  $B$  (maximum backoff duration) previous to  $\text{RAO}(i-1)$ . Let  $n_A$  denote this number. Secondly, from  $n_A$  we estimate the expected number of UEs that entered the backoff state during the period of length  $B + T_{\text{RAO}}$  previous to  $\text{RAO}(i)$ , that we denote as  $n_B$ . Finally, from  $n_B$  we estimate the expected value of  $n_r(i)$ .

Let  $X$  be the random variable that represents the UE backoff start time in the interval  $[0, B]$ . The UEs enter the backoff state at a fixed time  $t_0$  after the transmission of a preamble, where  $t_0$  depends on the RACH configuration parameters. Besides, preambles are transmitted at a RAO subframe, so all values of  $X$  are grouped in subframe intervals every  $T_{\text{RAO}}$ . We approximate these values to the discrete set  $\{x_0 \dots x_{K-1}\}$ , being  $x_i = t_0 + i T_{\text{RAO}}$ , as shown in Fig. 4. We assume that at any time window  $B = K T_{\text{RAO}}$ , all backoff times have the same probability (i.e.,  $\mathbb{P}(X = x_i) = 1/K$ ).

Let  $T_{\text{BO}}$  be the random variable that represents the backoff duration, with uniform distribution given by Eq. (1). If we denote by  $p$  the probability that a UE starting backoff in the interval  $[0, B]$  is still in backoff at time  $B$ ,  $p$  is computed as

$$\begin{aligned} p &= \sum_{k=0}^{K-1} \mathbb{P}(X = x_k, x_k + T_{\text{BO}} > B) = \frac{1}{K} \sum_{k=0}^{K-1} \frac{x_k}{B} = \\ &= \frac{1}{K} \sum_{k=0}^{K-1} \frac{k T_{\text{RAO}} + t_0}{B} = \frac{1}{K} \frac{1}{B} K \frac{x_0 + x_{K-1}}{2} = \\ &= \frac{1}{2B} (t_0 + (K-1)T_{\text{RAO}} + t_0) = \\ &= \frac{1}{2} \left(1 - \frac{T_{\text{RAO}}}{B}\right) + \frac{t_0}{B}. \end{aligned} \quad (11)$$

The number of UEs in backoff at  $\text{RAO}(i-1)$  is a random variable that follows a binomial distribution with success probability  $p$ ,  $n_A$  attempts, and expected value  $p n_A$ . Note that we do not consider those UEs that ended the backoff and started a new one before  $\text{RAO}(i-1)$ . But, if the congestion control performs effectively, the probability of a second backoff will be low, and the error that is incurred by this approximation is negligible. If we take  $n_b(i-1)$  as the expected value of the number of UEs in backoff, we can estimate  $n_A$  from

$$n_b(i-1) = p n_A. \quad (12)$$

We also assume that users will start backoff at  $x_{K-1} + T_{\text{RAO}}$  with the same probability as in  $x_i, i \in \{0, \dots, K-1\}$ . Then,  $n_B$  can be estimated from

$$n_B = n_A + \frac{n_A}{K} = \left(1 + \frac{T_{\text{RAO}}}{B}\right) n_A. \quad (13)$$

Let  $Z$  be the discrete random variable that represents the UEs backoff start time during the interval  $[0, B + T_{\text{RAO}}]$ , with values  $\{x_0 \dots x_K\}$  and probabilities  $\mathbb{P}(Z = z_i) = 1/(K+1)$ . If we denote by  $q$  the probability that a UE starting backoff in the interval  $[0, B + T_{\text{RAO}}]$  retransmits in the interval  $[B, B + T_{\text{RAO}}]$ ,  $q$  is computed as

$$\begin{aligned} q &= \sum_{k=0}^K \mathbb{P}(Z = z_K, B < z_k + T_{\text{BO}} < B + T_{\text{RAO}}) = \\ &= \frac{1}{K+1} \mathbb{P}(T_{\text{BO}} > B - z_0) + \\ &\quad + \frac{1}{K+1} \sum_{k=1}^{K-1} \mathbb{P}(B - z_k < T_{\text{BO}} < B - z_k + T_{\text{RAO}}) + \\ &\quad + \frac{1}{K+1} \mathbb{P}(T_{\text{BO}} < B - z_K + T_{\text{RAO}}) = \\ &= \frac{1}{K+1} \left( \frac{t_0}{B} + \sum_{k=1}^{K-1} \frac{T}{B} + \frac{B + T_{\text{RAO}} - (kT_{\text{RAO}} + t_0)}{B} \right) = \\ &= \frac{1}{K+1} \left( (K-1) \frac{T_{\text{RAO}}}{B} + \frac{T_{\text{RAO}}}{B} \right) = \frac{T_{\text{RAO}}}{B + T_{\text{RAO}}}. \end{aligned} \quad (14)$$

The number of UEs in backoff that retransmit at  $\text{RAO}(i)$  is a random variable that follows a binomial distribution with success probability  $q$ ,  $n_B$  attempts, and expected value  $q n_B$ . If we take  $n_b(i)$  as the expected value of this variable, we can estimate  $n_r(i)$  as

$$n_r(i) = q n_B, \quad (15)$$

and, from Eqs. (11)–(15), it follows that

$$n_r(i) = \frac{2 T_{\text{RAO}}}{B - T_{\text{RAO}} + 2 t_0} n_b(i-1). \quad (16)$$

Finally,  $n_b(i)$  is computed by replacing Eqs. (16), (10), and (4) in Eq. (3).

Fig. 5 shows an example, obtained by discrete-event simulation, of the performance of this estimator. In this example, the estimator operates during a congestion episode induced by the M2M traffic benchmark described in Section V, with the UEs implementing ACB. As can be seen, the error in the estimation is minimal.

## B. Dynamic Barring Rate Tuning

In this section, we describe a mechanism for tuning the barring rate dynamically using the information provided by the estimation of the number of users in backoff state,  $n_b(i)$ . We assume that the eNodeB has no information about the number of new arrivals. Nevertheless, by estimating  $n_b(i)$ , the expected number of retransmissions,  $n_r(i)$ , is also estimated. From this, the unused capacity of the RACH can be evaluated and the incoming traffic can be modulated accordingly, that is, by computing  $P_{\text{ACB}}$  as a function of  $n_r(i)$ :  $P_{\text{ACB}} = f(n_r(i))$ .

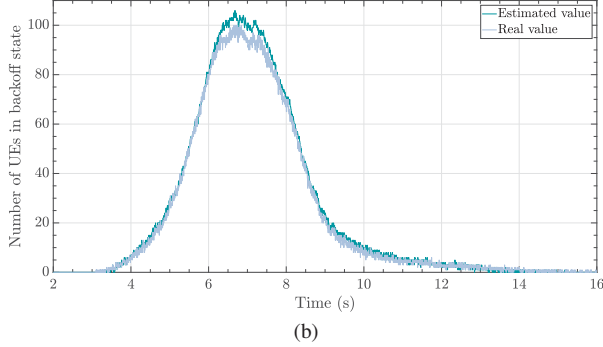
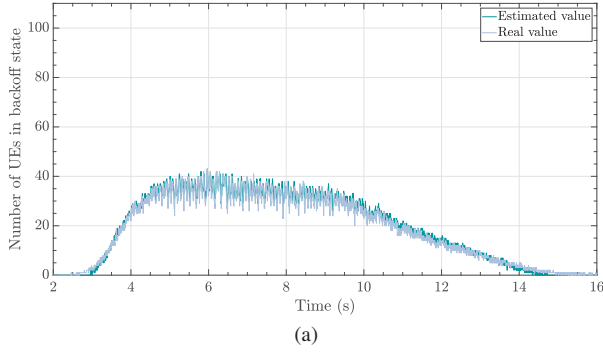


Figure 5. Performance of the estimator during a congestion episode. (a) Dynamic ACB. (b) Static ACB.

The number of access requests per RAO that maximizes the expected number of UEs that access the channel successfully is given as a function of the available preambles,  $R$ , by [19]

$$n_t^*(R) = [\log(R/(R-1))]^{-1}. \quad (17)$$

At  $\text{RAO}(i)$ , if  $n_r(i) \geq n_t^*(R)$ , all new arrivals should be barred, so  $f(n_r(i) \geq n_t^*(R)) = 0$ . On the other hand, if  $n_r(i) = 0$  there is no reason to reduce the incoming traffic, so  $f(0) = 1$ . Between these two boundary conditions,  $f(\cdot)$  should be a decreasing function. We have tested by simulation linear, quadratic, and cubic functions and we have found that, although all of them work quite well, the cubic function shown in Fig. 6 provides a better performance. Thus, we propose to compute  $P_{ACB}$  as

$$P_{ACB}(i) = f(n_r(i)) = \frac{2 n_r(i)^3}{n_t^*(R)^3} - \frac{3 n_r(i)^2}{n_t^*(R)^2} + 1. \quad (18)$$

The value of  $P_{ACB}$  is notified to the UEs by the eNodeB through the SIB2 message with a periodicity of  $T_{\text{SIB2}} = 80 \text{ ms} \equiv 16 \text{ RAOs}$ .

Finally, the dynamic ACB implementation can be summarized as

- Every  $\text{RAO}(i)$ ,
  - 1) Estimate  $\mathbb{E}\{n_u\}(i)$  with Eq. (9),
  - 2) Estimate  $n_t(i)$  with Eq. (10),
  - 3) Estimate  $n_r(i)$  with Eq. (16),
  - 4) Estimate  $n_b(i)$  with Eq. (4).
- Every  $T_{\text{SIB2}}$ ,
  - 1) Compute  $P_{ACB}(i)$  with Eq. (18),
  - 2) Update  $P_{ACB}(i)$  in SIB2.

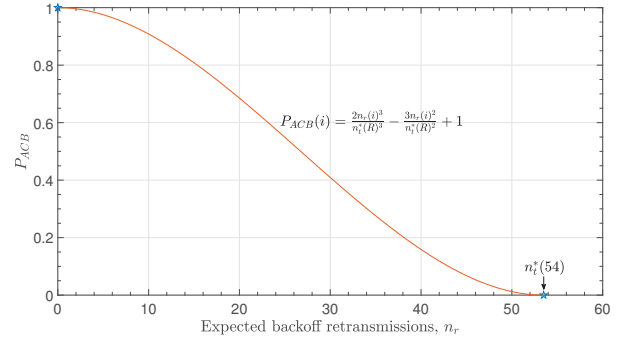


Figure 6. Computation of  $P_{ACB}$  from the expected number of retransmissions.

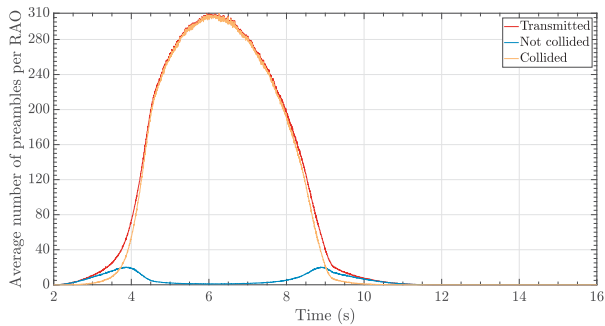
Table I  
RACH CONFIGURATION

Parameter	Setting
PRACH Configuration Index	$prach\text{-}ConfigIndex = 6$
Periodicity of RAOs	$T_{\text{RAO}} = 5 \text{ ms}$
Subframe length	1 ms
Available preambles for contention-based random access	$R = 54$
Maximum number of preamble transmissions	$preambleTransMax = 10$
RAR window size	$W_{\text{RAR}} = 5 \text{ subframes}$
Maximum number of uplink grants per subframe	$N_{\text{RAR}} = 3$
Maximum number of uplink grants per RAR window	$N_{\text{UL}} = W_{\text{RAR}} \times N_{\text{RAR}} = 15$
Preamble detection probability for the $k$ th preamble transmission	$P_d = 1 - \frac{1}{e^{kC}}$ [11]
Backoff Indicator	$B = 20 \text{ ms}$
Maximum number of $Msg3$ and $Msg4$ transmissions	5
Round-trip time (RTT) of $Msg3$	8 subframes
RTT of $Msg4$	5 subframes

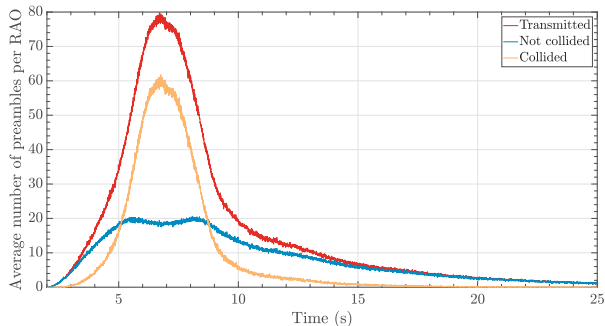
## V. PERFORMANCE EVALUATION

In this section, we evaluate the performance of the proposed dynamic ACB scheme with  $T_{ACB} = 4 \text{ s}$ , and compare it with the performance of a static ACB with optimal parameters ( $P_{ACB} = 0.5$  and  $T_{ACB} = 4 \text{ s}$ ). The parameters for static ACB are chosen based on a previous work [7] where we showed that the lowest access delay for a highly congested scenario given a requirement of  $P_s \geq 0.95$  is achieved when  $P_{ACB} = 0.5$  and  $T_{ACB} = 4 \text{ s}$ . We measure three KPIs, namely the probability to successfully complete the random access procedure,  $P_s$ ; the mean number of preamble transmissions needed by the UEs to successfully complete the random access procedure,  $W$ ; and the access delay (mean and percentiles),  $D$ . These KPIs are in conformance with the 3GPP directives [11] to assess the efficiency of the LTE-A random access procedure with M2M communications.

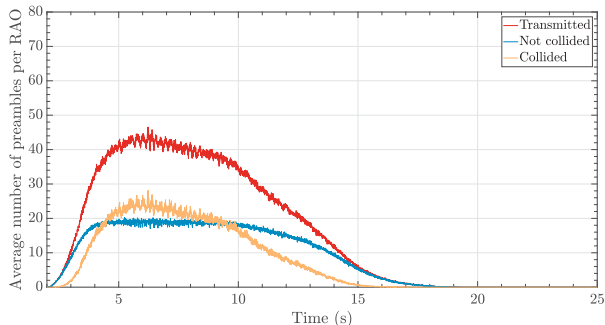
A single cell environment is assumed to evaluate the network performance. The system conveys traffic from both H2H and M2M UEs. The access requests of H2H UEs are distributed uniformly over time with a mean arrival rate of  $\lambda_H = 1$  arrivals/s. The M2M requests follow a  $Beta(3, 4)$  distribution over a period of 10 s, according to the traffic model 2 specified by the 3GPP in [11]. This traffic model can be seen as an extreme scenario in which a vast number of M2M UE arrivals occur in a highly synchronized manner (e.g., after an alarm that activates them).



(a)



(b)



(c)

Figure 7. Average number of preambles per RAO when H2H UEs and massive M2M UEs ( $N_M = 30\,000$ ) attempt to access. (a) No access control. (b) Static ACB (0.5, 4 s) implemented. (c) Dynamic ACB operating.

We developed a discrete-event simulator that fully reproduces the behavior of UEs, eNodeB, and RACH during the random access procedure described in Section III-A. We consider the typical PRACH configuration for these kind of studies, *prach-ConfigIndex* 6, in conformance to the LTE-A specification [8], [11], with the parameter values listed in Table I. The value of each KPIs is obtained as the mean of 100 simulation runs. Each simulation run uses a different random seed and ends when all the M2M UEs have completed their random access procedure.

Fig. 7a depicts the temporal distribution of preamble transmissions during a congestion episode of  $N_M = 30\,000$  M2M UEs and no congestion control. As can be seen, an access control mechanism is required for alleviating congestion. 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 retransmissions and reduces dramatically  $P_s$ .

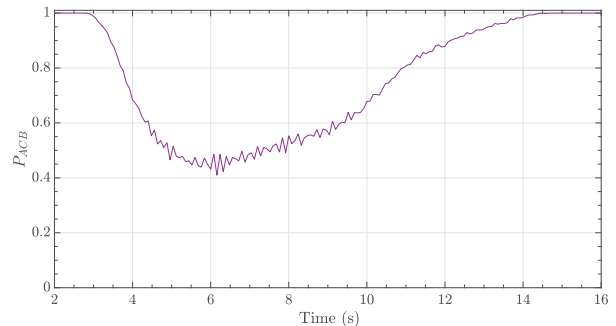


Figure 8. Evolution of  $P_{ACB}$  in dynamic ACB during a congestion episode.

Fig. 7b shows the above scenario for the static ACB with  $P_{ACB} = 0.5$  and  $T_{ACB} = 4$  s. Now, the total number of preamble transmissions is reduced while the number of not collided transmissions increases, what results in a higher probability of successful access. But the number of collisions is still high because it fails to maintain the preamble transmission rate close to the optimal number of access requests per RAO that maximizes the expected number of successful access,  $n_t^*(R)$ , which is approximately 53 with  $R = 54$  available preambles.

Fig. 7c illustrates the performance of our proposed dynamic ACB scheme in the same scenario. We see that the total number of preamble transmissions and the number of collided transmissions are both considerably reduced when compared to the static ACB. More importantly, this reduction is achieved under dynamic conditions and by adapting  $P_{ACB}$  accordingly. Note that most of the time the preamble transmissions rate is maintained close to the optimal value ( $n_t^*(R) \approx 53$ ) and the not collided preambles rate is maintained close to the RACH capacity (approx. 20 [7, Eq. (4)]), using the available resources efficiently.

Fig. 8 shows how  $P_{ACB}$  is tuned by the eNodeB with dynamic ACB operating during the congestion episode. As can be seen, before the congestion starts,  $P_{ACB} = 1$ ; it gradually decreases when the number of preamble transmissions rises, and grows again as the traffic diminishes, until it goes back to 1. Note that  $P_{ACB}$  changes dynamically with a granularity of  $T_{SIB2}$ .

Fig. 9a depicts the successful access probability,  $P_s$ , for both ACB types with different M2M traffic intensities ( $N_M$ ). As can be seen, in heavy-loaded scenarios ( $N_M > 26\,000$ ) the dynamic ACB provides a  $P_s \geq 0.99$ , clearly outperforming the static one. In terms of mean number of preamble transmissions needed to successfully complete the random access procedure, Fig. 9b shows that the dynamic ACB slightly increases this metric when ( $30\,000 > N_M$ ). Finally, Fig. 9c compares the mean access delay of both ACB types. We can see that the dynamic ACB decreases notably the mean access delay, even more in low-loaded scenarios, since our solution uses efficiently the available RACH resources.

In Table II, we can see several statistics for the three schemes: no control, static ACB, and dynamic ACB. For each of them, results for a heavy-loaded traffic scenario ( $N_M = 30\,000$ ) are shown. We show the KPIs defined at the beginning of Section V for both H2H and M2M UEs.

It is evident from the results that the network with no access control suffers in terms of  $P_s$  and  $W$ . However, the access delay is moderate. On the other hand, our proposed dynamic



## VI. CONCLUSION

We propose a dynamic ACB mechanism that adjusts the barring factor in real-time, based on the estimation of the number of UEs that are in backoff state. We have evaluated this solution by means of extensive discrete-event simulations, and compared its performance with that of a static ACB with optimal parameters. We conclude that, in a heavy-loaded traffic scenario, the dynamic ACB increases noticeably the probability of successful access while shorting the access delay, and the number of preamble transmissions needed to successfully complete the random access procedure. Moreover, we have checked that the dynamic ACB does not have a significant impact on H2H traffic performance. Our solution is conforming with current system specifications and allows efficient congestion control, facilitating the coexistence of H2H and M2M traffic.

## REFERENCES

- [1] Ericsson. (2017, June) 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.
- [3] Z. Dawy, W. Saad, A. Ghosh, J. G. Andrews, and E. Yaacoub, "Toward massive machine type cellular communications," *IEEE Wireless Communications*, vol. 24, no. 1, pp. 120–128, February 2017.
- [4] I. F. Akyildiz, S. Nie, S.-C. Lin, and M. Chandrasekaran, "5G roadmap: 10 key enabling technologies," *Computer Networks*, vol. 106, pp. 17–48, 2016.
- [5] 3GPP, *TS 22.368, Service Requirements for Machine-Type Communications*, Mar 2017.
- [6] —, *TS 22.011, V13.1.0, Service Accessibility*, Sep 2017.
- [7] 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. 67, no. 4, pp. 3505–3520, April 2018.
- [8] 3GPP, *TS 36.321, Medium Access Control (MAC) Protocol Specification*, Sep 2017.
- [9] —, *TS 36.331, Radio Resource Control (RRC), Protocol specification*, Sep 2017.
- [10] —, *TS 36.213, Physical layer procedures*, Dec 2014.
- [11] —, *TR 37.868, Study on RAN Improvements for Machine Type Communications*, Sep 2011.
- [12] H. Kim, S. s. Lee, and S. Lee, "Dynamic extended access barring for improved M2M communication in LTE-A networks," in *2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, Oct 2017, pp. 2742–2747.
- [13] 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.
- [14] C. M. Chou, C. Y. Huang, and C.-Y. Chiu, "Loading prediction and barring controls for machine type communication," in *2013 IEEE International Conference on Communications (ICC)*. IEEE, June 2013, pp. 5168–5172.
- [15] A. Lo, S. Member, Y. W. Law, and M. Jacobsson, "Enhanced LTE-Advanced Random-Access Mechanism for Massive Machine-to-Machine (M2M) Communications," *27th Meeting of Wireless World Research Forum*, pp. 1–7, 2011.
- [16] 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)*, June 2015, pp. 5815–5820.
- [17] 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.
- [18] 3GPP, *TR 36.912, Feasibility study for Further Advancements for E-UTRA*, Mar 2017.
- [19] 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.
- [20] O. Arouk and A. Ksentini, "General Model for RACH Procedure Performance Analysis," *IEEE Commun. Lett.*, vol. 20, no. 2, pp. 372–375, Feb 2016.

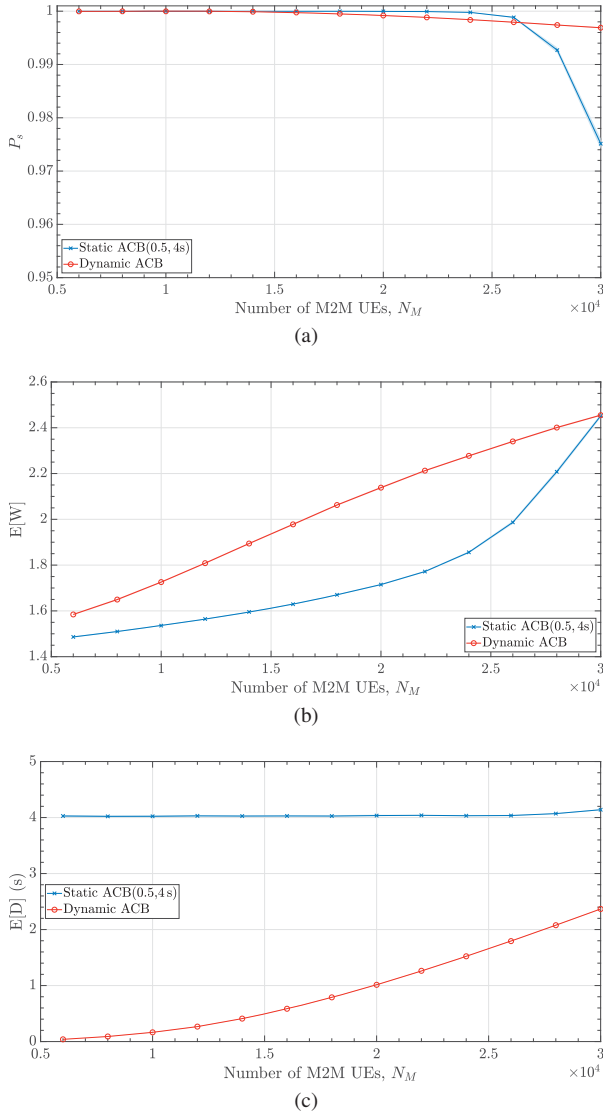


Figure 9. KPIs of the LTE-A network for static ACB and the proposed dynamic ACB. (a) Successful access probability. (b) Expected number of preamble transmissions per RAO. (c) Expected access delay.

Table II  
KPIs OBTAINED FOR LTE-A AND DIFFERENT ACB IMPLEMENTATIONS IN A HEAVY-LOADED TRAFFIC SCENARIO.

Key Performance Indicator		No ACB		ACB(0.5, 4s)		Dynamic ACB	
		H2H	M2M	H2H	M2M	H2H	M2M
Success probability	$P_s$	0.66	0.31	0.99	0.97	0.99	0.99
Number of preamble transmissions, $W$	$E[W]$	2.15	3.45	1.60	2.45	2.03	2.45
Access delay, $D$ [ms]	$E[D]$	46	71	32	4111	42	2367
	$D_{99}$	203	217	125	24676	151	9700
	$D_{95}$	141	186	67	15744	104	8242
	$D_{80}$	59	117	43	7865	56	4742
	$D_{10}$	17	18	16	22	17	21

ACB scheme shows an improvement over the solution with static ACB, providing a  $P_s \geq 0.99$  for both H2H and M2M communications while reducing considerably the M2M access delays without increasing the H2H access delays. 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.