Contention Resolution Queues for Massive Machine Type Communications in LTE

Andres Laya*, Luis Alonso† and Jesus Alonso-Zarate‡
*KTH Royal Institute of Technology, Sweden. e-mail: laya@kth.se
†Universitat Politècnica de Catalunya (UPC), Spain. e-mail: luisg@tsc.upc.edu
‡Centre Tecnològic de Telecomunicacions de Catalunya (CTTC), Spain. e-mail: jesus.alonso@cttc.es

***Abstract*—In this paper, we address the challenge of high device density performing simultaneous transmissions by proposing and evaluating a solution to efficiently handle the initial access contention for highly dense LTE networks. We present the implementation of a tree-splitting algorithm in the access procedure of LTE, which is capable to cope with high number of simultaneous arrivals. Based on simulations we show a feasible implementation capable to achieve, under certain network configuration conditions, up to 85% average access delay reduction and 40% reduction on the average energy consumption, while maintaining a consistently low blocking probability, regardless of the number of initial simultaneous access attempts.**

***Index Terms*—Machine Type Communications; Energy consumption; LTE-Advanced; Machine-to-Machine; Tree-splitting; Distributed Queuing; Random Access Procedure.**

I. Introduction

Machine Type Communication (MTC) impose challenges on cellular networks related to new traffic characteristics and number of devices in the networks [1]. The 3GPP has been actively describing and addressing many of these challenges. On the particular topic of contention resolution, a technical report highlights the need to design improvements for the access mechanisms of cellular systems to be able to handle tens of thousands of devices in a single cell [2], [3]. This report resulted in the standardization of Access Class Barring (ACB) scheme as part of the Release 8 and Extended Access Barring (EAB) in Release 11. The limitation of these solutions is that they are based on backoff periods that disperse access attempts. This has a negative impact on the energy consumption and the access delay for the devices. This is the limitation that we address in this paper.

Since LTE-A systems are based on an ALOHA-like scheme for the contention resolution, they are not stable when the channel occupation rate is high. This is presented in the queuing theory analysis in [4]. An increasing number of schemes can be found in the literature, which aim at overcoming these limitations, as covered in [5]. Nevertheless, most proposals fall sort on providing solutions that consider the balance between access delay, access probability rate and energy consumption.

In this paper, we consider a particular approach that can efficiently tackle the instability issue. The initial proposal on this line was presented by Campbell and Xu [6], consisting on a MAC protocol whose high performance is completely independent of the number of devices sharing a common channel. The proposal is known as Distributed Queuing (DQ) and it is fitted to the high density of devices to be found in MTC. Following the initial proposal, consequent studies have analyzed the performance of the DQ protocol [7], [8], demonstrating the stability of its performance and the near optimum behavior in terms of channel utilization, access delay, and energy consumption. However, the DQ principles cannot be directly applied to the LTE standard. For this reason, in this paper, we describe the modifications to use the DQ mechanisms to improve the contention-based Random Access (RA) procedure, used for initial association of uncoordinated devices in LTE-A. This work is an extension of the initial implementation presented in [9], which is limited in the use of access resources, as further explained in Section IV.

The organization of the document is the following: on Section II, the access mechanisms used in LTE-A standard are explained. In Section III, we provide an introduction to the DQ concept. The integration of DQ into the RA procedure is given in Section IV. In Section V-A, we present the system and simulation setup that we use to obtain the comparative results presented in Section V-B. Finally, the conclusions and further remarks are provided in Section VI.

II. Contention-based Access Mechanisms in LTE-A

In this section, the contention-based Random Access (RA) procedure is described. This procedure is triggered in the cases of initial network access, connection reestablishment, handover, synchronization for data transmission or reception, and for resource scheduling requests for new data transmissions [5], [10]. The contention-based RA procedure consists on a four-message handshake between the device (UE) and the eNodeB. This is presented in in Fig. 1, and the purpose for each of the four messages is described next.

Msg. 1, RA Preamble: it is a 6 bit signature that devices use when attempting an access (with a maximum of 64 possibilities). A device randomly selects one preamble from those available and transmits it on the Random Access CHannel (RACH). The RACH is formed by a periodic sequence of allocated time-frequency resources, usually referred to as RA slots. For the LTE Frequency Division Duplex (FDD) specification, the RA slot periodicity varies between 1 and 20 ms. [11]. Collisions occur if more than one device selects the same preamble and transmits it over the same RA slot.

Msg. 2, Random Access Response (RAR): if the eNodeB detects a preamble, it replies with the RAR. This

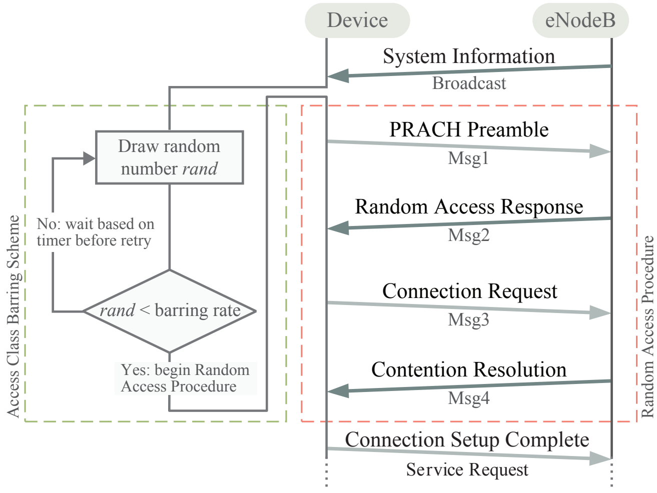


Fig. 1. Access Class Barring (ACB) scheme and contention-based Random Access (RA) Procedure in LTE-Advanced.

message is sent over the Physical Downlink Shared CHannel (PDSCH) and contains information related to the detected preamble, uplink timing alignment instructions and the resource grant to transmit Msg. 3. Optionally, the RAR can include a Backoff Indicator (BI). The RAR contains different subheaders, one for the BI information and additional subheaders to deliver feedback information to each preamble detected without collision. If a device receives a RAR without information for the preamble it used, it will perform a backoff time according to the BI parameter [12].

MSG. 3, CONNECTION REQUEST: after the initial uplink resource grant informed in Msg. 2, the device transmits a connection request to the eNodeB, conveying the establishment cause. In case of undetected preamble collision, more than one device got assigned the same uplink resource; the eNodeB will detect the Msg. 3 collision and it will not acknowledge this message; resulting on access failure for the devices.

MSG. 4, CONTENTION RESOLUTION: a device receiving Msg. 4 will have a successful access. If there is no successful reception, a new access attempt is scheduled.

The 3GPP included the ACB scheme in subsequent amendments to the standard to provide additional control mechanisms [2]. In this scheme, each device determines the barring status with the information provided from the serving network. If the ACB is active, the device draws a uniform random number between 0 and 1 when initiating connection establishment and compares with the barring rate established by the network. This will determine whether the device is barred or not, as shown in Fig. 1. The barring factor ranges from 0 to 95% and the barring time spans from 4 to 512 seconds [11].

III. DISTRIBUTED QUEUING FOR CONTENTION RESOLUTION

The Distributed Queuing (DQ) is based on a m-ary tree splitting algorithm with a simple set of rules to organize devices in virtual queues during an access procedure. When collisions are detected, the devices are split into groups for the subsequent transmissions, reducing the probability of collision due to simultaneous attempts. The distributed scheduling of the queues enables almost full channel utilization regardless of its capacity, the number of the transmitting devices, and the traffic pattern. The queues are distributed in the sense that each device uses internal counters to represent the queue length and the position of the device within the queue. The values of each counter are updated based on the network feedback. In this way, the devices can process their transmission turn.

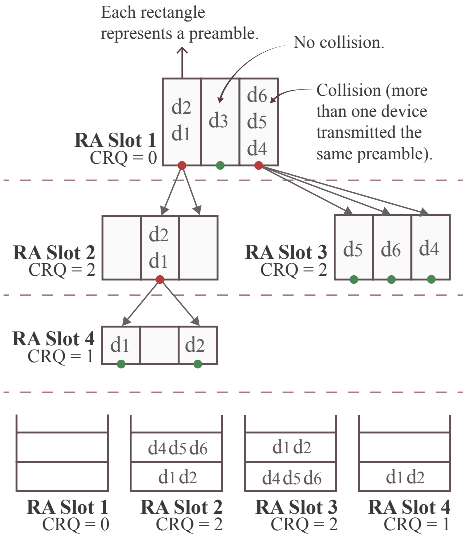


Fig. 2. Tree-splitting algorithm and CRQ behavior in the collision resolutions. For this example, 3 preambles are available on each RA slot. The content of each rectangle denotes the ID of devices that transmit the same preamble. On the lower part, a representation of the CRQ is depicted.

Fig. 2 depicts an example of the algorithm execution. In the first RA Slot, six devices request access. If more than one device selects the same preamble, there is a collision and a RA Slot is assigned exclusively that the set of devices. These devices enter a queue referred to as Collision Resolution Queue (CRQ). For each preamble collision there will be a different contention group and the CRQ length will increase by one. The access node (eNodeB) must provide feedback for the RA Slots status so each device can compute its position in the queue. This is achieved by means of two integer numbers, the RQ counter and the pRQ counter, as explained next.

The RQ counter is used to store the CRQ length and it is calculated as follows:

- If there have been collisions pending resolution ($RQ > 0$), reduce RQ by one to account for the resolution attempt of the devices at the head of the CRQ.
- Increase the value of RQ by one for each preamble with a collision state in the previous RA Slot.

The pRQ counter is used to keep the device's position in the CRQ. It is calculated in the following manner:

- If the device is waiting in the CRQ ($pRQ > 0$), it must first decrease its RQ and pRQ values by one and then increase the RQ by one for each preamble with a collision state in the previous RA Slot.
- If the device has transmitted an preamble on the previous RA Slot and collided, the device sets its pRQ value to point at RQ.

On the example in Fig. 2, at RA Slot 1, d1 and d2 collide with preamble 1 and enter in the first position in the CRQ; d3 succeeds with preamble 2; d4, d5 and d6 collide with preamble 3 and enter in the second position in the CRQ. At RA Slot 2, d1 and d2 contend since they are at the first position in the CRQ. d4, d5 and d6 will wait in the queue until the next RA Slot. d1 and d2 collide again, this time with preamble 2; this group enters at the end of the CRQ. At RA Slot 3, d4, d5 and d6 used a different preamble, so the three succeed and leave the CRQ. At RA Slot 4, d1 and d2 contend again and succeed.

IV. CRQ integration into the RA procedure

A. DQ-based RA procedure

In this section, we describe one approach to adapt the queuing mechanisms to the RA procedure. Traditional DQ systems employ minislots for preamble transmission. However, leveraging on the availability of orthogonal preambles in LTE, different preambles on the same RA Slot can be used instead. On this implementation, devices will select a specific RA Slot on the first attempt and it will only use subsequent repetitions of the same RA Slot on the following LTE frames if further retransmissions are needed.

Upon initial access, a device selects a RA Slot and waits for the corresponding Msg. 2 in order to get the current status of the CRQ. If there is an ongoing contention in the selected RA Slot ($RQ > 1$), new devices are not allowed to enter. Therefore, the device will not transmit in the next RA Slot and repeats this process until there are no further collisions.

If a free RA Slot is found, the device will send a preamble on the next occurrence of the RA Slot and it will wait for the corresponding Msg. 2. Three states must be provided on the Msg. 2 and the devices will do as follows:

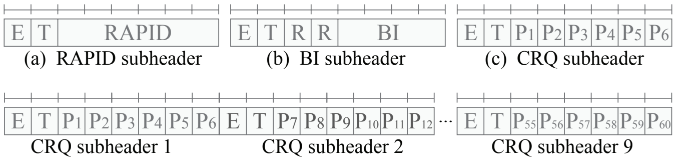
1) Empty state: no preamble was received. The device will increase by one the preamble retransmission counter and reenter the CRQ.
2) Collision state: a collision was detected. The device will increase by one the preamble retransmission counter and reenter the CRQ.
3) Success state: a preamble was received and no collision was detected. The device will decode the RAR and proceed to the transmission of Msg. 3.

B. DQ feedback implementation in Msg. 2

We have revisited the standard RAR feedback in order to accommodate the corresponding CRQ information. The RAR (Msg. 2) is a MAC PDU composed of a variable size header and zero or more RAR payloads. This header consists of one or more subheader of the following types [12]:

- RAPID Subheaher (Fig. 3(a)): RAPID stands for Random Access Preamble Identifier, which correspond to the preamble number. There will be one RAPID subheader for each successfully received preamble.
- BI Subheader (Fig. 3(b)): this subheader contains the BI parameter and there could only be one of these subheaders at the most per RAR.

The subheader's fields are shown in Fig. 3 and correspond to:



(a) RAPID subheader (b) BI subheader (c) CRQ subheader

CRQ subheader 1 CRQ subheader 2 CRQ subheader 9

Fig. 3. MAC PDU subheaders for Mag 2. (a) corresponds to the subheader related to a specific preamble. (b) corresponds to the Backoff Indication (BI) parameter. (c) corresponds to the subheader used to provide CRQ feedback. The lower part of the figure shows the concatenation of CRQ subheaders.

- E: extension bit, 1 means that an additional subheader is attached after.
- T: type bit. 1 for RAPID subheader, 0 for BI subheader.
- R: reserved bit.
- BI: Backoff Indicator parameter.
- RAPID: Random Access Preamble Identifier.

As initially presented in [9], we propose the inclusion of a CRQ subheader (Fig. 3(c)) to provide the CRQ feedback. Since there can only one BI subheader per RAR, appending subsequent subheaders with T = 1 provides the necessary distinction between the three types of subheaders. The specific fields of a CRQ subheader are P1 to P6, which correspond to the status of each preamble in the previous RA Slot, 1 for collision and 0 for no collision or no detection. Since each MAC PDU subheader is 8 bits long, the feedback for up to 6 preambles can be included in a single CRQ subheader. The work presented in [9] is limited to 6 preambles. In this paper, we extend the implementation for those cases in which the network use more than 6 preambles. In such cases, additional CRQ subheaders are consequently appended in the RAR, on the same manner that consequent RAPID subheaders can be appended in the standard procedure. The three states required on the feedback for each preamble are provided as follows:

1) Empty state: Px as zero and no associated RAPID subheader for this preamble.
2) Collision state: Px as one and no associated RAPID subheader for this preamble.
3) Success state: Px as zero and associated RAPID subheader for this preamble.

Following this implementation, we describe in the next section the system setup used to compare the standard RA procedure with a DQ-based RA procedure.

V. Performance evaluation

A. System Model and Definitions

To conduct simulations, the ns-3 modules validated in [5] for the LTE RA modules in FDD mode have been used. We assume an LTE network where devices are cell-synchronized and they received all configuration parameters related to the RA procedure. It is also assumed that the eNodeB will not be able to decode simultaneous transmission of the same preamble. We evaluate up to 1500 simultaneous access attempts using the simulation parameters in Table I, with this metrics:

1) *Average Access Delay*: time elapsed between the first preamble transmission and Msg. 4 reception, only successful accesses are considered for the calculation.

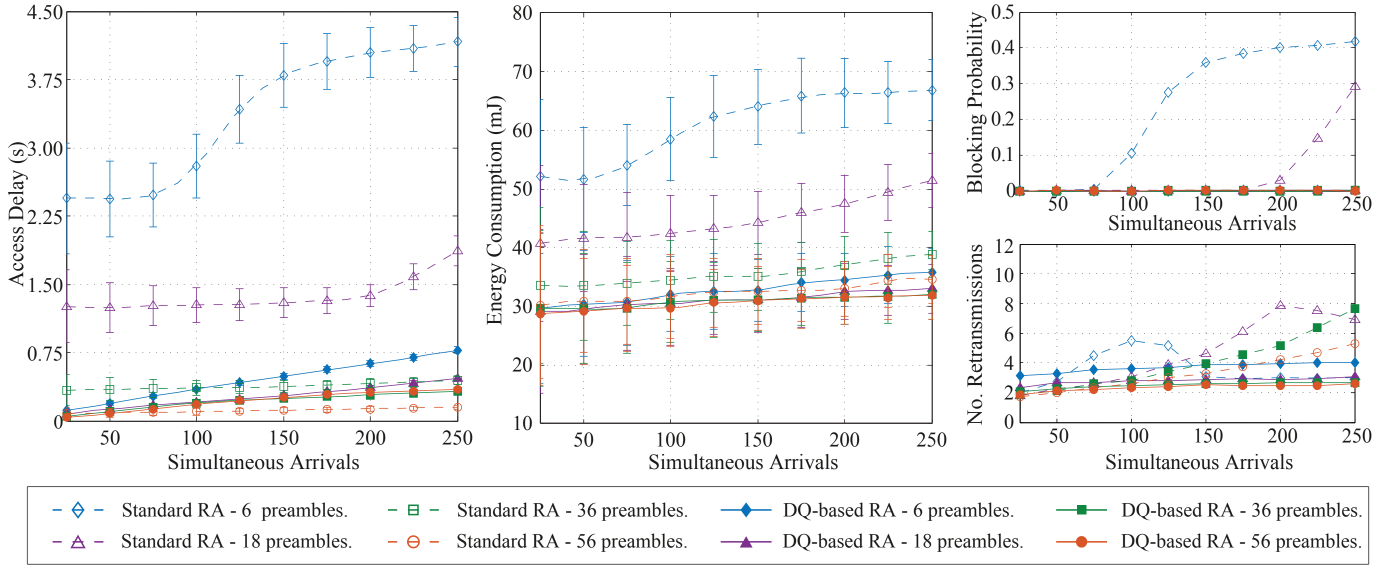


Fig. 4. Comparative between the standard RA procedure with ACB and the proposed DQ-based RA procedure, with up to 250 simultaneous arrivals.

2) *Blocking Probability*: the probability of a device reaching the maximum number of attempts and being unable to complete an access process.
3) *Average Energy Consumption*: the total energy spent until the access to the network has been granted, only successful accesses are considered for the average calculation.
4) *Average Number of Preamble Retransmissions*: the number of attempts executed before getting access.

If a device reaches the maximum number of preamble transmission without gaining access, it is blocked by network. Therefore, the time elapsed during the access attempts, the average number of preamble retransmissions and the energy consumed are not considered for the average calculation.

B. Comparative Results

In this section, the performance of the standard RA procedure is compared with the proposed DQ algorithm. The results are presented in two different sets. The first one for simultaneous arrivals for up to 250 devices. The second set is for simultaneous arrivals from 250 to up to 1500 devices.

TABLE I
SIMULATION PARAMETERS

Parameter	Simulated Values				Unit
No. Available preambles	56	36	18	6	int.
Barring Factor[a]	80	60	40	40	%
Barring Time[a]	2				s
PRACH Configuration Index[b]	3				int.
Backoff Indicator[b]	480				ms
Max. Preamble retransmissions[b]	20				int.
RAR Window Size[a]	5				ms
Contention Resolution Timer[a]	48				ms
Power consumption values[c]					
Transmission	500				mW
Active Period (Reception mode)	150				mW
Accurate clock (Idle mode)	10				mW

[a] All possible values available in 3GPP TS 36.331 [11].
[b] All possible values available in 3GPP TS 36.321 [12].
[c] Values taken from the description given in [13], assuming that the power consumption on transmission mode is equal to the radiated power.

In Fig. 4, the results for the first set can be appreciated. The limitations of the standard RA procedure are evident when the number of available preambles is low. This holds for the cases of 6 and 18 preambles. The performance shows a direct negative impact on the access delay and energy consumption. Moreover, it can be appreciated how the standard procedure is not capable to cope with an increasing number of simultaneous arrivals and the blocking probability reaches unacceptable rates; after 200 simultaneous arrivals for 18 preambles and after 75 simultaneous arrivals for 6 preambles. The increase of the blocking probability can only be prevented on the current standard by increasing the access delay; either by increasing the BI, increasing the barring time or decreasing the barring rate. The CRQ implementation shows an effective improvement for these cases. If we compare the performance of both procedure using 6 preambles, the DQ implementation achieves at least 85% average access delay reduction and 40% reduction on the average energy consumption, while maintaining a consistently low blocking probability, regardless of the number of initial simultaneous access attempts.

Contrarily, the standard RA procedure is capable to handle the contention efficiently for the case of 56, with the lowest access delay. Nevertheless, there is a 10% higher energy consumption when compared to the DQ-based RA, which is due to the higher number of average preamble retransmissions required by the devices under the standard RA procedure.

The comparison results for simultaneous arrivals ranging from 250 to 1500 are shown in Fig. 5. For all cases, it can be appreciated how the standard RA procedure is unable to cope with the high number of arrivals, leading to unacceptable blocking probability values, even for the case of 56 preambles.

For the cases where less preambles are available, the barring factor is lower (Table I). This generates higher dispersion of subsequent attempts, reducing the collision probability. The effect of a higher barring factor can be seen on the case of 56

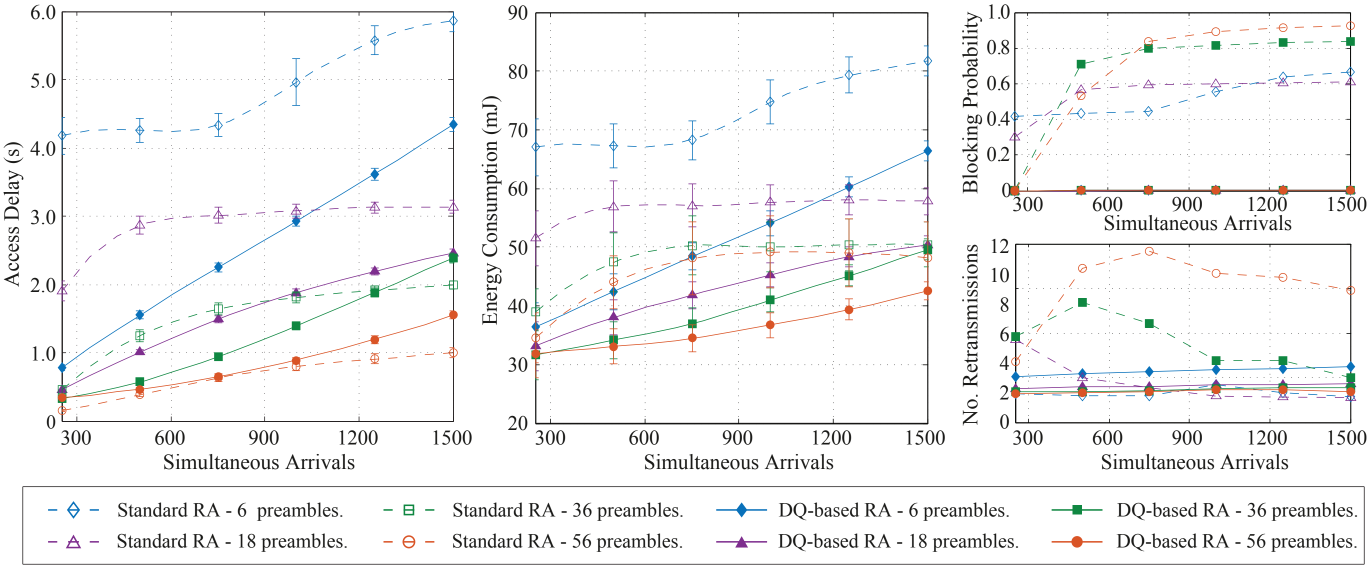


Fig. 5. Comparative between the standard RA procedure with ACB and the proposed DQ-based RA procedure, with up to 1500 simultaneous arrivals.

preambles, which has a higher blocking probability compared to the case of 6 preambles after 500 simultaneous arrivals. Reducing the barring factor could have the benefit of reducing the blocking probability, but the access delay and the energy consumption will increase accordingly.

The benefits of the DQ-based RA procedure are evident in the second set. Even with the increase in the access delay and the energy consumption, the blocking probability show how the distributed scheduling enables almost full channel utilization regardless of the number of transmitting devices.

As a clear comparison, the performance of the standard RA procedure with 6 preambles at 250 simultaneous arrivals is comparable with the performance of the DQ-based RA procedure with 6 preambles at 1500 simultaneous arrivals; in terms of the access delay and the energy consumption.

At this point, the cases in which this solution is beneficial have been presented. Therefore, future work on this area should consider the scheduling of uplink data transmissions following the distributed queues mechanism after the access to the network has been granted. Extending the gains in latency and reduction of the energy consumption on MTC devices.

VI. Final Assessment and Conclusions

In this work, we present an overview of the RA procedure in order to explain how the current standard can only accommodate high number of devices by increasing the backoff parameters for preamble retransmissions, resulting in increased access delay and energy consumption on the device side.

According to these limitation, we present an alternative procedure based on a tree-splitting algorithm and a distributed queue which can reduce the average access delay, while reducing the energy consumption and maintaining a low blocking probability for an increasing amount of simultaneous access attempts. The solution can be implemented with simple modifications to the standard and could reduce the energy consumption and the access delay under most of the evaluated conditions, but it is particularly relevant for the extreme arrival cases expected in future MTC scenarios.

Acknowledgment

This work has been partially funded by Wireless@KTH, the EIT ICT Labs project EXAM, ADVANTAGE (FP7-607774) and the Catalan Government under grant (2014-SGR-1551).

References

[1] 3GPP TR 22.868 V8.0.0, “Study on Facilitating Machine to Machine Communication in 3GPP Systems,” March 2007.

[2] 3GPP TR 37.868 V11.0.0, “Study on RAN Improvements for Machine-type Communications,” September 2011.

[3] 3GPP TSG RAN WG2 #69bis R2-102296, “RACH intensity of Time Controlled Devices,” Beijing, China, April 2010.

[4] B. Yang, G. Zhu, W. Wu, and Y. Gao, “M2M access performance in LTE-A system,” *Transactions on Emerging Telecommunications Technologies*, vol. 25, no. 1, pp. 3–10, 2014.

[5] 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,” *Communications Surveys Tutorials, IEEE*, vol. 16, no. 1, pp. 4–16, First 2014.

[6] W. Xu and G. Campbell, “A Near Perfect Stable Random Access Protocol for a Broadcast Channle,” in *IEEE Proc. ICC 92*, vol. 1, 1992, pp. 370–374.

[7] L. Alonso, R. Agustí, and O. Sallent, “A Near-Optimum MAC Protocol Based on the Distributed Queueing Random Access Protocol (DQRAP) for a CDMA Mobile Communication System,” in *IEEE Journal on Selected Areas in Comm*, vol. 18, no. 9, Sep. 2000, pp. 1701–1718.

[8] J. Alonso-Zarate, C. Verikoukis, E. Kartsakli, A. Cateura, and L. Alonso, “A near-optimum cross-layered distributed queuing protocol for wireless LAN,” *Wireless Communications, IEEE*, vol. 15, no. 1, pp. 48–55, February 2008.

[9] A. Laya, L. Alonso, and J. Alonso-Zarate, “Efficient Contention Resolution in Highly Dense LTE Networks for Machine Type Communications,” in *GLOBECOM, 2015 IEEE*, Dec 2015.

[10] Sesia, S. and Baker, M. and Toufik, I., *LTE - The UMTS Long Term Evolution: From Theory to Practice*. Wiley, 2011, pp. 421–456.

[11] 3GPP TS 36.331 V10.5.0, “Evolved Universal Terrestrial Radio Access (E-UTRA); Radio Resource Control (RRC),” March 2012.

[12] 3GPP TS 36.321 V9.3.0, “Evolved Universal Terrestrial Radio Access (E-UTRA); Medium Access Control (MAC),” June 2010.

[13] T. Tirronen, A. Larmo, J. Sachs, B. Lindoff, and N. Wiberg, “Machine-to-Machine Communication with Long-Term Evolution with reduced device energy consumption,” *Transactions on Emerging Telecommunications Technologies*, vol. 24, no. 4, pp. 413–426, 2013.