Multiple Access Class Barring factors Algorithm for M2M communications in LTE-Advanced Networks

Meriam Bouzouita
University of Rennes, Rennes, France
SUP'COM, Ariana, Tunisia
mariem.bouzouita@supcom.tn

Yassine Hadjadj-Aoul
University of Rennes, Rennes, France
yhadjadj@irisa.fr

Nawel Zangar
Faculty of science of Tunis, University of el Manar, Tunisia
SUP'COM, Tunisia
nawel.zangar@insat.rnu.tn

Gerardo Rubino
INRIA, Rennes
Gerardo.Rubino@inria.fr

Sami Tabbane
SUP'COM, Tunisia
Sami.Tabbane@insat.rnu.tn

ABSTRACT

The forecast dramatic growth, of the number of Machine-to-Machine (M2M) communications, challenges the traditional networks of Mobile Network Operators (MNO). In fact, a large number of devices may attempt simultaneously to access the base station, which may result in severe congestions at the random-access channel (RACH) level. To alleviate such congestion while regulating the M2M devices' opportunities to transmit, the Access Class Barring (ACB) process was proposed. In this article, we propose a novel implementation of the ACB mechanism in the context of multiple M2M traffic classes. Based on a scheduling algorithm, we have applied a PID controller to adjust dynamically multiple ACB factors related to each class category, guaranteeing a number of devices around an optimal value that maximizes the Random Access (RA) success probability. The obtained results demonstrate the efficiency of the proposed mechanism by increasing the success probability and minimizing radio resources' underutilization with respect to each class priority.

Keywords

M2M; MTC; Access Class Barring; Congestion; Random Access.

1. INTRODUCTION

Machine Type Communications (MTC) or Machine-to-Machine (M2M) communications are nowadays gaining a huge interest from the stakeholders, and particularly the Mobile Network Operators (MNO), and their customers. In fact, M2M communications are seen as one of the most important opportunities to face the revenue's cuts for mobile operators while providing a plethora of services to the customers. These services can be declined in a wide range of automated applications covering a large number of domains [4][10].

The huge number of M2M devices, which may attempt, at the same time, to access the base station, may result in severe congestions at the random-access channel (RACH) level [7]. In fact, a large number of devices may be triggered simultaneously and attempt to perform the Random Access (RA) in order to request for uplink radio resources. This congestion is even more aggravated when considering the class of event-driven communications, in which a large number of devices is activated during a very short period of time. These devices contend for a limited number of resources, called preambles. Indeed, if two or more MTC equipments choose the same preamble, the Evolved Node B (eNB) will be unable to identify the initiator of the RA and a collision will happen [2]. This may reduce the success access probability and may result in a performance degradation for MNO.

In this paper, our main concern is to design an efficient mechanism to maximize the wireless resources' utilization while guaranteeing the access priorities that may exist between the different class of applications. Another important concern consists in protecting the M2M event-driven communications by prioritizing their access to the channel while adapting rapidly the ACB factors to absorb this type of traffic. To achieve this objective, we proposed both: (1) a Proportional Integral Derivative (PID) controller to make the number of M2M devices attempting the access procedure converge to the optimal one, and (2) a scheduling algorithm to ensure a weighted proportional fairness among M2M devices of the different classes except the class of event-driven devices, which is prioritized.

The remainder of this article is organized as follows. Section 2 is dedicated to the description of the proposed mechanism. Section 3 portrays the simulation setup and discusses the obtained results. Finally, the paper concludes in Section 4 with a summary recapping the main advantages and achievements of the proposed scheme.

2. MULTIPLE ACCESS CLASS BARRING ALGORITHM

Many ACB-based algorithms were proposed in the literature [1]. However, most of them do not consider multi-class M2M devices. Thus, the obtained ACB factor is applied to

MSWiM'15 November 02-06, 2015, Cancun, Mexico.

all classes of M2M devices, which might be harmful for MNO networks. The originality of our approach resides, first, in considering devices belonging to different classes. Second, in opposition to existing works, we propose to find out the optimal number of devices and devising, then, an efficient controller to make the number of devices attempting the RA converge to the optimal one.

2.1 System Model

In this paper, we consider the classical architecture described in [4]. We propose, in the following, to model the RA process, as described in [2]. Our model for M2M devices' random access with multiple ACB factors (i.e. one factor per class) is influenced by the single-class system model that we proposed in [6]. The model is a fluid one: the involved quantities, the whole numbers, are seen here as real quantities. The parameters used in the proposed system model are listed below:

- $x_{1,i}(t)$: the number of backlogged MTC devices from class i at time t, where $i \in \{1, 2, \ldots, k\}$. The constant k represents the number of considered classes.
- $x_2(t)$: the number of MTC devices that pass the ACB check and wait to start RA attempt at time t.
- $x_3(t)$: the number of MTC devices that succeed RA procedure at time t.
- λ_i: the arrival rate of MTC devices from class $i \in \{1, 2, \ldots, k\}$. Different traffic patterns will be considered in the following, depending on the type of M2M application.
- $\theta_{1,i}(x_{1,i})$: the rate of ACB failure for class $i \in \{1, 2, \ldots, k\}$.
- $\theta_2(x_2)$: the rate of RA failures (i.e. collision and retransmission).
- $\mu(x_3)$: the rate of MTC departure after performing the RA successfully.
- $p_i(x_{1,i})$: the ACB factor for class $i \in \{1, 2, \ldots, k\}$.
- N: indicates the total number of radio resources (preambles) available during one time slot. It is a constant value.
- $R_s(x_2)$: denotes the number of MTC devices that transmitted their preambles successfully.

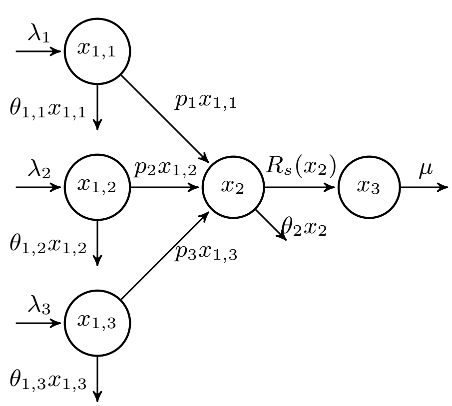

Figure 1: System description for $k = 3$

Now we are ready to describe the evolution of the states $x_{1,i}$ with $i \in \{1, 2, \ldots, k\}$, x_2 and x_3 based on the model described in Fig. 1. Let first define $\mathbb{C} = \{1, 2, \ldots, k\}$ as the set of classes. The system's dynamics is described by the following system of differential equations:

$$\begin{cases} \dfrac{\mathrm{d}x_{1,i}}{\mathrm{d}t} = \lambda_i - p_i x_{1,i} - \theta_{1,i} x_{1,i}, & \text{for all } i \in \mathbb{C}, \\ \dfrac{\mathrm{d}x_2}{\mathrm{d}t} = \displaystyle\sum_{i=1}^{k} p_i x_{1,i} - R_s(x_2) - \theta_2 x_2, \\ \dfrac{\mathrm{d}x_3}{\mathrm{d}t} = R_s(x_2) - \mu x_3, \end{cases} \tag{1}$$

with the constraints that for all $i \in \mathbb{C}$, $x_{1,i}$, x_2 and x_3 should be nonnegative.

The function $R_s(x_2)$ is represented by the expected number of MTC devices succeeding in the access process. This number represents the number of preambles with only one device.

Let's define P_s as the probability that a given preamble is chosen by one MTC device (i.e. probability of success). To that purpose, we suppose that there are N available preambles in each RA opportunity and x_2 MTC devices contending for these resources. This is a typical "balls into bins' problem", in which P_s is given by:

$$P_s = \frac{x_2}{N}\left(1 - \frac{1}{N}\right)^{x_2 - 1}. \tag{2}$$

For a large N, this can be approximated by:

$$P_s = \frac{x_2}{N}\,\mathrm{e}^{-\dfrac{x_2 - 1}{N}}. \tag{3}$$

Therefore,

$$R_s(x_2) = x_2 e^{-\dfrac{x_2 - 1}{N}}. \tag{4}$$

The collision probability P_c can be derived using (3) and the idle probability (i.e. no user chooses a given preamble):

$$P_c = 1 - \frac{x_2}{N}\,\mathrm{e}^{-\dfrac{x_2 - 1}{N}} - e^{-\dfrac{x_2}{N}}. \tag{5}$$

2.2 Optimal number of M2M devices

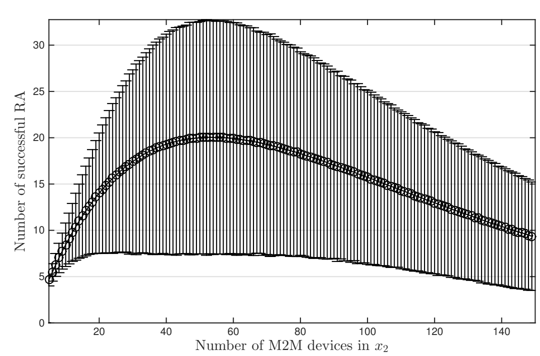

Figure 2: Successful RA

The main idea, in this section, is to derive an optimal number of contending M2M devices x_2 in a way to be used by the proposed controller as an objective to achieve (i.e. targeted number of M2M devices).

All devices that have passed successfully the ACB test, will contend for the same radio resources (N) as a way to access the network. The optimal value of devices, performing RA at the same time (i.e. x_2^{ref}) and maximizing the success probability, is obtained based on Monte-Carlo simulations.

For the simulations, we vary the number of M2M devices between 1 and 150 devices. We evaluated the average and the variance on the number of successful RA. To validate the obtained results, many seeds were tested and the results, obtained in Fig. 2, were similar. The results, depicted in Fig. 2, show that the maximum number of successful RA is obtained when the number of M2M devices in x_2 is equal to $(N-3)$. This number will be adopted as the optimal value (i.e. the targeted value) to generate the appropriate ACB factor in our scheduling algorithm.

2.3 PID feedback control on the access probability

In the proposed approach, the dynamic adjustment of the ACB factor (i.e. access probability) is achieved using a discrete Proportional Integral Derivative (PID) regulator [5]. The main idea, behind, is to make the total number of MTC devices, contending for the access, converges to the targeted value, which is defined in the previous subsection.

The discrete PID regulator can be described by the following equation [5]:

$$P_{acb}(n) = K_p e(n) + K_i \sum_{k=0}^{n} e(k) + K_d(e(n) - e(n-1)) \quad (6)$$

where n, P_{acb}, e, K_p, K_i and K_d represent respectively the instant, the controller output, the difference between the measured value and the set point value (i.e. the targeted value), the proportional gain, the integral gain and the derivative gain. In order to get the ideal response of the system, we considered the Ziegler-Nichols method [5] for the tuning of PID parameters.

As P_{acb} is a probability, a saturation block[1] is added to bind the values within the interval $[0,1]$. The probability calculated in (6) is general and do not concern a particular class of traffic. Thus, the probability P_{acb}^{i} for each class of service $i \in \mathbb{C}$ is derived from this value, as described in the next section. These values are broadcasted through a signaling channel to the different classes of M2M applications.

2.4 Proposed algorithm

In this subsection, we describe in details the proposed algorithm, which is illustrated in the figure below, to compute multiple ACB factors for the different classes of applications.

After a phase of initialization (at phase 1), the eNB estimates the error (i.e. e) to be used thereafter in the PID regulator (at phase 2). The error reflects the difference between the obtained value of the number of contending devices x_2 at step n and the targeted value x_2^{ref}. Therefore, the eNB estimates the number of successful ACB tests $\hat{x}_2$ by estimating the states $x_{1,i}$, for all $i \in \mathbb{C}$ (i.e. $\hat{x}_{1,i}$, for all $i \in \mathbb{C}$). These estimates can be easily obtained by a per-class counting of the number of devices that have passed successfully the ACB test and by averaging these obtained values using an Exponentially Weighted Moving Average (EWMA).

After that (at phase 3), the eNB uses the PID controller to adjust the ACB factor (i.e. P_{acb}) enabling to converge to-

[1]A component imposing higher and lower bounds.

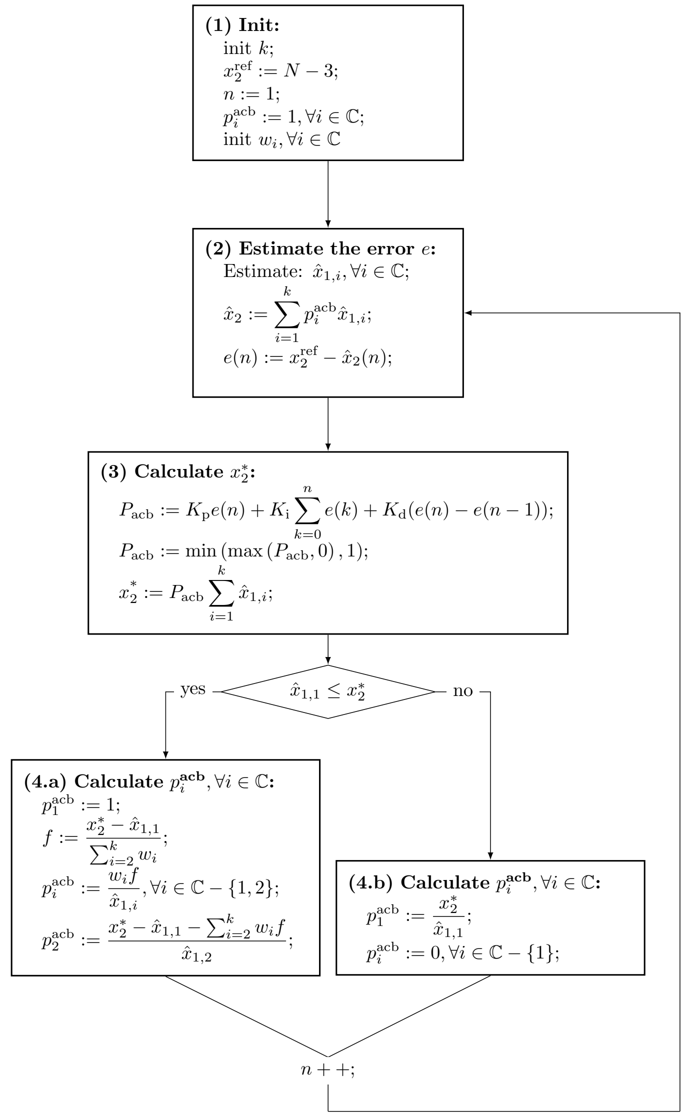

Figure 3: Scheduling Algorithm

wards the targeted number of contending devices (i.e. x_2^{ref}). As the output of the PID controller should be a probability, in spite of considering P_{acb}, we consider only values within the interval $[0,1]$ by applying the: $\min(\max(P_{acb}, 0), 1)$. The computation of the blocking probability P_{acb} allows generating an optimal number of equipments that should pass successfully the ACB process (i.e. x_2^*) based on the estimated $\hat{x}_{1,i}$, for all $i \in \mathbb{C}$ (see phase 3).

In the proposed algorithm, we considered the first class as the most prioritized by using an absolute priority access, as it is the case for some M2M applications such as emergency services. The access probabilities of the devices belonging to other classes are balanced using a weight factor w_i, for all $i \in \mathbb{C}$. Depending on the availability of preambles for the first class, we pass by phase (4.a) or (4.b). Phase (4.b) is executed when there are not enough preambles to grant the access to all the devices of the first class (i.e. x_2^* is smaller than $\hat{x}_{1,1}$). In this case the blocking probability for class 1 is calculated to have an optimal number of contending devices for this class $p^1_{acb} = \frac{x^*_2}{\hat{x}_{1,1}}$. The devices from the other classes are blocked. When there are enough preambles to grant the access for all the devices in class 1, phase (4.a) is executed. In this case, p^1_{acb} is equal to one (i.e. all the devices from the class are accepted), and the access is shared fairly between the other classes depending on the weight of each class (i.e. weighted fairness).

Once the ACB factors generation process is finished, it is broadcasted through a signaling channel to all the equipments, which should update their access probability and start ACB check. Then, the operation is repeated from phase 2, after incrementing the step variable n.

Note that the broadcasting of the access probabilities is repeated for each frame, which represents a delay of 10ms. Other delays might be considered for a less important accuracy and reactivity. The considered delay is short but represents a negligible overhead.

3. PERFORMANCE EVALUATION

3.1 Simulation parameters

Having described the details of the proposed algorithm to calculate a per-class ACB factor for heterogeneous M2M devices, we direct now our focus on evaluating its performance using the "Network Simulator (ns3)" environment [8]. The proposed model supports an unlimited and configurable number of types of M2M applications. However, for the simulations, we considered only the following classes:

- *Emergency and prioritized applications (class 1)*: this category of applications must be processed with the highest rate of successful accesses. Emergency applications' arrival follows a Beta-based traffic model [3].
- *Applications for remote control and surveillance (class 2)*: this category represents M2M application with a good level of priority and continuous data transmission [9]. The model of arrival of corresponds here to a uniform process.
- *Smart-grid-related applications (class 3)*: in this category a reasonably large amount of data is periodically transferred to eNB and, thus, causing RAN overload. Such applications have a very low priority of access with delay-tolerance and can be rejected in case of congestion. The periodic arrival of smart-grid related devices is also modeled with a uniform-based process.

The parameters' settings are listed in Table 1. The duration of the simulation corresponds to the distribution period of the traffic of most critical traffic. More details on the traffic patterns can be found in [3].

3.2 Simulation results

To validate the proposed model defined in Figure 1, we present, in this subsection, the analytical values of success and collision probabilities against the ones obtained using simulation. Then, we will give the number of successful ACB tests compared with the targeted value (i.e. optimal value as found in section 2.2) to demonstrate the effectiveness of our proposal. Finally, we show the efficiency of the proposed solution in prioritizing the different M2M applications.

Figures 4 and 5 show the simulation results obtained respectively for the success probability and the collision probability against those obtained theoretically. It can be seen

Table 1: Simulation parameters

Parameters	Values
Simulation duration	10s
Total number of preambles	54
Cell bandwidth	5MHz
Max. # of preamble retransmissions	10
ac-BarringTime	4s
Total number of MTC devices	10000
# of devices in classes (1,2,3)	(1000,3000,6000)
Beta distribution time	10s
Beta function parameters	$\alpha = 3; \beta = 4$

that the simulation results match very well the theoretical values, which allow validating the proposal model.

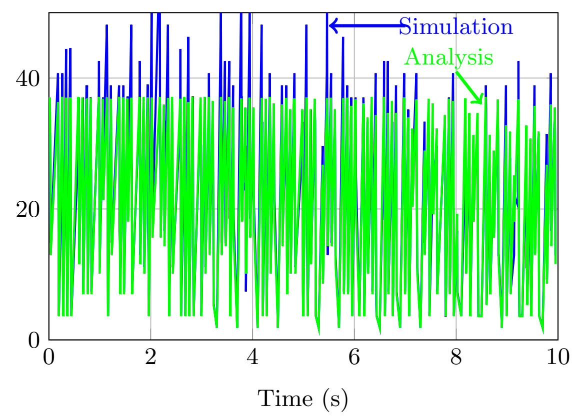

Figure 4: Success Probability (%)

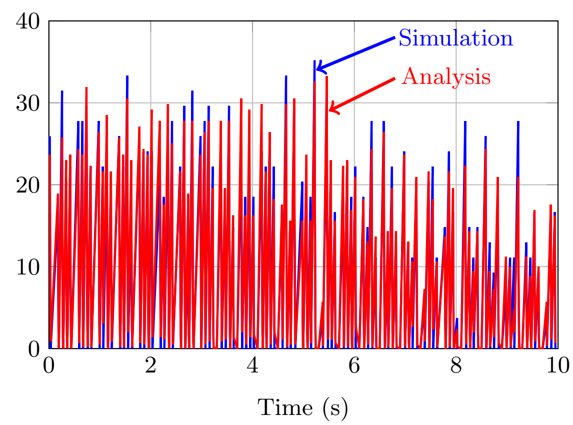

Figure 5: Collision Probability (%)

Figure 6 illustrates the instantaneous and the average (i.e. EWMA) number of successful ACB tests. We can easily see that even if the instantaneous values oscillate between 30 and 70, the average values remain very close to the targeted value (i.e. 51), which is the objective of our mechanism. This, clearly, demonstrates the effectiveness of the PID controller as it helps in regulating the ACB factor dynamically according to the congestion level and the M2M application classes.

Note that values smaller than the target may lead to resources' underutilization, while values bigger than the target

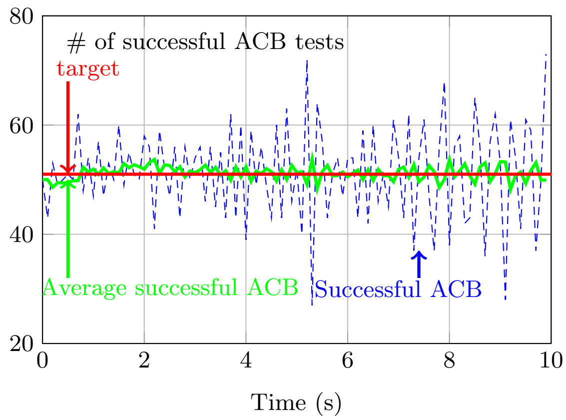

Figure 6: Total successful ACB tests vs Setpoint

may lead to excessive collisions, which also lead to resources' underutilization.

To see the efficiency of the proposed algorithm in prioritizing different classes of M2M applications, let see the results obtained in Fig. 7. Figure 7 depicts the cumulative number of successful ACB tests for each class of M2M applications in figures (B) compared with the cumulative number of arrivals for each class in figures (A). It can be easily seen that the number of arrivals for class 1 is nearly equal to the number of successful ACB tests, which is in a complete conformance with the absolute constraint requirements for this category of traffic.

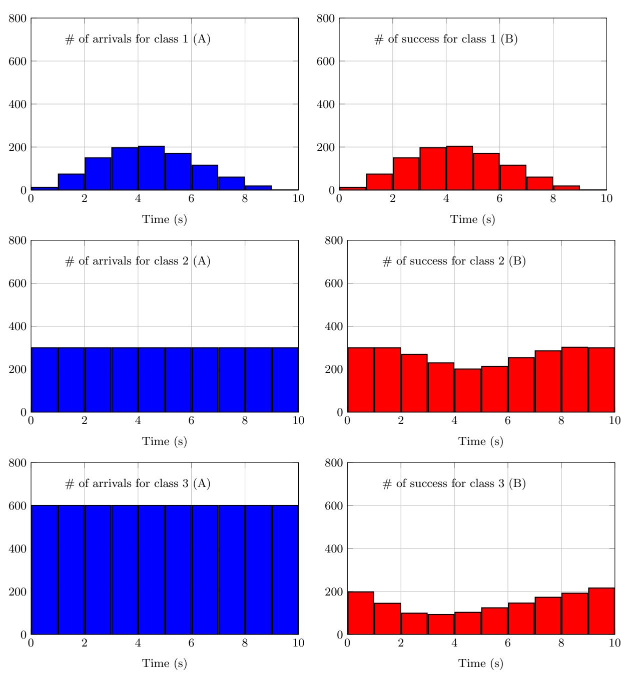

Figure 7: (A) Cumulative number of arrivals per second, (B) Cumulative number of successful ACB tests per second

Another important observation, in Fig. 7, is that when the congestion is at its maximum (between seconds 2 and 8) the traffic of class 2 suffers from some degradation as some of the traffic is blocked. However, the blocking probability of this traffic class, which has a weight bigger than class 3, is low. This has direct consequences on the large number of successful ACB tests for this class. On the other hand, the less prioritized class (i.e. class 3) suffer from the beginning from blocked traffic. The phenomenon is exacerbated when the congestion is at its peak, as it can be seen in Fig. 7. In fact, given that devices of class 3 have the lowest priority, they will be blocked in case of congestion, i.e. when arrivals of class 1 and class 2 are more important. Consequently, it results in reducing collision probability as it is observed in figure 5.

4. CONCLUSION

In this paper, the RAN overload issue caused by MTC in LTE-Advanced networks has been addressed. We have proposed a novel mechanism as a way to alleviate RAN congestion by efficiently managing the M2M devices' random accesses. Using Monte-Carlo simulations, we find out the optimal number of the MTC devices that should compete for the random access to maximize the number of devices succeeding in the ACB procedure. After that, in order to regulate adaptively the ACB factor guaranteeing a total number of devices around the targeted value, we used a discrete PID controller. Then, we applied a scheduling algorithm as a way to schedule different M2M traffic classes. Simulation results show that the proposed mechanism can accurately predict congestion situations while significantly reducing the collision probability.

5. REFERENCES

[1] 3GPP. Evolved universal terrestrial radio access (e-utra); radio resource control (rrc). Technical report, TS 36.331 V10.2.0, 06 2011.

[2] 3GPP. Medium access control (mac) protocol specification. Technical report, TS 36.321 V10.2.0, 06 2011.

[3] 3GPP. Study on ran improvements for machine-type communications (release 11). Technical report, TR 37.868 V11.0.0, 09 2011.

[4] 3GPP. System improvements for machine-type communications, technical specification group services and system aspects (release 11). Technical report, TR 23.888 V1.6.0, 2011.

[5] K. J. Åström and T. Hägglund. Advanced PID control. ISA-The Instrumentation, Systems, and Automation Society, Research Triangle Park, NC, 2006.

[6] M. Bouzouita, Y. Hadjadj-Aoul, N. Zangar, S. Tabbane, and C. Viho. A random access model for M2M communications in LTE-advanced mobile networks. In Modeling and Simulation of Computer Networks and Systems, pages 577 – 599. Morgan Kaufmann, 2015.

[7] A. Ksentini, Y. Hadjadj-Aoul, and T. Taleb. Cellular-based machine-to-machine: overload control. Network, IEEE, 26(6):54–60, November 2012.

[8] ns3. http://www.nsnam.org/, 2015.

[9] I. Petiz, P. Salvador, and A. N. Nogueira. Characterization and modeling of m2m video surveillance traffic. In IARIA Int. Conf. on Advances in Future Internet - AFIN, August 2012.

[10] G. Wu, S. Talwar, K. Johnsson, N. Himayat, and N. D. Johnson. M2M: from mobile to embedded internet. IEEE Communications Magazine, 49(4):36–43, 2011.