Recursive Pseudo-Bayesian Access Class Barring for M2M Communications in LTE Systems

Hu Jin, *Member, IEEE*, Waqas Tariq Toor, Bang Chul Jung, *Senior Member, IEEE*, and Jun-Bae Seo

***Abstract*—Commercial long-term evolution (LTE) systems adopt an access class barring (ACB) mechanism in the initial random access procedure with multiple preambles in order to accommodate bursty traffic arrivals of machine-type communications. In this paper, we propose two *Bayesian* ACB algorithms that estimate the number of active machine devices based only on the number of *idle* preambles in each slot. In the commercial LTE systems, eNodeB cannot instantaneously distinguish if a particular preamble is sent from a single device (i.e., success) or multiple devices (i.e., collision). However, the idle preambles can be instantaneously detected at the base station (BS) in each slot. Numerical results show that the proposed algorithms yield quite similar performance with the ideal ACB algorithm, assuming that the exact number of active devices is known to the eNodeB.**

***Index Terms*—Access class barring (ACB), Bayesian estimation, internet-of-things (IoTs), machine-type communication (MTC), massive random access.**

I. Introduction

Machine-to-machine (M2M) commucation, also called machine type communication (MTC), is considered as an enabling technology for internet-of-things (IoTs), and it is expected that a staggering number of machine devices request easy and fast uplink access to networks in a near-real-time fashion. In accomplishing M2M communications in long-term evolution (LTE) systems [1], when an active MTC device wishes to transmit data over uplink, it randomly chooses one of a pool of random access preambles (RAPs) and transmits it through the random access channel (RACH). If a large number of MTC devices try to access eNodeB (evolved Node B), i.e., base station (BS), almost at the same time, then congestion may occur in the RACH.

As previous work for controlling the congestion in M2M communications, a random access (RA) technique was proposed by exploiting the timing alignment information in order to distinguish various MTC devices [2]. Furthermore, in [3], it was proposed to increase the number of RAPs by reducing their cyclic shift size, which results in reducing collisions. In order to provide access quality-of-service (QoS) for MTC devices, [2], [3] adopted access class barring (ACB) scheme [4]. The ACB scheme for bursty traffic sources was investigated in [5], where a heuristic algorithm controls ACB factor dynamically. In [6], the minimum service time for all MTC devices to complete their access was studied under the assumption that the number of active MTC devices is exactly *known* to eNodeB. In [7], relaxing the ideal assumption of [6], a maximum likelihood estimator on the number of active MTC devices was proposed by observing the number of successfully transmitted RAPs and that of RAPs not transmitted at each slot. To improve the performance futher, in [8], an ACB scheme was proposed by exploiting both the timing advance information [2] and the dynamic ACB factor [6], [7]. In [9], a dynamic allocation of the RAPs for the MTC devices was proposed with the dynamic ACB scheme.

In [5]–[9], the total population size of MTC devices, which are activated in a certain activation period, is assumed to be exactly known to eNodeB, and the collisions are assumed to be recognized instantaneously at the eNodeB. In practice, however, the exact number of MTCs that may be activated in a certain period in a cell may not be known. Furthermore, if multiple MTCs transmit the same RAP, i.e., RAP collisions occur, it is not possible for eNodeB to know instantaneously which RAPs are transmitted by more than one MTCs. In fact, the eNodeB in commercial LTE systems can recognize the collisions some time slots after it receives RAPs [4].

In order to overcome the two drawbacks in the previous work, this paper proposes two ACB algorithms, called pseudo-Bayesian ACB (PB-ACB) and enhanced PB-ACB (EPB-ACB), where ACB factor is dynamically adjusted in accordance with an estimation on the number of backlogged MTC devices in a recursive Bayesian way. Taking into consideration non-zero processing delay of the commercial LTE systems, we assume that eNodeB cannot identify instantaneously which RAPs are transmitted successfully or not. Under the limitation of such non-zero processing delay of detecting RAPs, one additional striking feature of rendering our algorithms more practically worthy is that our estimation is based only on observing the number of RAPs *not transmitted*. In practice, it is much easier to detect RAPs not transmitted in comparison with the previous work of performing estimation with idle (not transmitted), success and/or collided RAP observations [5]–[9]. Finally, the difference between PB-ACB and EPB-ACB is that the latter takes into account bursty arrivals of MTC devices, while the former does not.

II. System Model

This section introduces the RA procedure of LTE system and the traffic model of MTC devices. In eNodeB, RACH appears every x (msec) for $1 \leq x \leq 20$ [10], which depends on the serving area of the eNodeB and its traffic load. We assume that one RACH appears every 1 (msec), say every slot, and that ACB factor $p \in [0, 1]$ is broadcasted at each slot to accurately control the congestion. The contention-based RA procedure consists of the four steps explained below so that radio resource called time-frequency resource block (RB) is eventually allocated to an MTC device after the four steps are successfully completed.

1) *Step 1:* An activated MTC device for a packet transmission generates a random real number in [0, 1]. If it is less than ACB factor p, the MTC device randomly selects one among M RAPs and transmits it to RACH; otherwise, its access is barred. *Thus, our first research interest is to find an optimal ACB factor that maximizes the number of success RAPs in each slot, which will*

Manuscript received April 11, 2016; revised January 1, 2017; accepted February 22, 2017. Date of publication March 10, 2017; date of current version September 15, 2017. This work was supported in part by the National Research Foundation of Korea (NRF) Grant funded by the Korea Government (MSIP) (No. 2015R1C1A1A01053857) and in part by the Basic Science Research Program through the NRF funded by the Ministry of Science, ICT & Future Planning (MSIP) (NRF-2016R1A2B4014834). The review of this paper was coordinated by Dr. F. Gunnarsson. *(Corresponding author: Bang Chul Jung.)*

H. Jin and W. T. Toor are with the Department of Electronics and Communication Engineering, Hanyang University, Ansan 15588, South Korea (e-mail: hjin@hanyang.ac.kr; waqastoor@hanyang.ac.kr).

B. C. Jung is with the Department of Electronics Engineering, Chungnam National University, Daejeon 34134, South Korea (e-mail: bcjung@cnu.ac.kr).

J.-B. Seo is with the Department of Electrical Engineering, Indian Institute of Technology Delhi, New Delhi 110016, India (e-mail: jbseo@iitd.ac.in).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TVT.2017.2681206

Algorithm 1: Pseudo-Bayesian ACB Algorithm (PB-ACB).

1. Initialize $\nu_0 = M$ and $p_0 = 1/M$.
2. **if** the number of idle preambles is r **then**
3. $\Delta\nu = 0.582 \cdot M - 1.582 \cdot r$
4. $\nu_{t-1} = \nu_{t-1} + \Delta\nu$ (Correct the previous estimation)
5. $\nu_t = \nu_{t-1} + a_t - c_t$ (New estimation)
6. **end if**
7. $\nu_t = \max(M, \nu_t)$ and $p_t = \min\left(1, \frac{M}{\nu_t}\right)$

Algorithm 2: Enhanced PB-ACB Algorithm (EPB-ACB).

1. Initialize $\nu_0 = M$, $p_0 = 1/M$, and $k_0 = 0$.
2. **if** the number idle preambles is r **then**
3. $\Delta\nu = 0.582 \cdot M - 1.582 \cdot r$
4. $\nu_{t-1} = \nu_{t-1} + \Delta\nu$
   (Correction step for bursty arrivals)
5. **if** $\Delta\nu > 0$ **then**
6. $k_t = k_{t-1} + 1$ and $\nu_{t-1} = \nu_{t-1} + k_t \cdot \Delta\nu$
7. **else**
8. $k_t = 0$ and $\nu_{t-1} = \nu_{t-1}$
9. **end if**
10. $\nu_t = \nu_{t-1} - c_t$
11. **end if**
12. $\nu_t = \max(M, \nu_t)$ and $p_t = \min\left(1, \frac{M}{\nu_t}\right)$.

be discussed in Section III-B. Note we assume that it takes one slot, say τ_1, for the active MTC device to transmit its RAP.

2) *Step 2:* The eNodeB detects which RAPs are transmitted and then broadcasts RA response messages corresponding to the RAPs detected. It should be noted that the eNodeB cannot distinguish whether an RAP is transmitted from a single MTC (i.e., success) or more than one MTCs (i.e., collision) in this step such that the decision on success or collision cannot be made; however, it was assumed to be possible in [5]–[9]. Generally, the time duration between Step 1 and Step 2 takes 2 to 12 (msec), which is denoted by τ_2. *Our goal at this step is to estimate the number of active MTC devices based only on the number of RAPs not transmitted.* While the decision on success or collision should be postponed, it is possible to decide which RAPs are transmitted or not, which is a byproduct.

3) *Step 3:* Only the MTC devices that have received the RA response messages are allowed to transmit the eNodeB a connection request message to the RB specified in the RA response message and initiate a contention resolution (CR) timer. In this step, the eNodeB is now able to decide collisions since different MTC devices shall transmit the connection request messages to the same RB. The time duration between Step 2 and Step 3 is equal to 5 (msec), which is denoted by τ_3. Thus, the RAP declared a successful transmission at this step is in fact the one that has been transmitted $\tau_1 + \tau_2 + \tau_3$ slots before.

4) *Step 4:* The eNodeB transmits CR messages to the MTC devices whose messages sent in Step 3 are successfully decoded. If an activated MTC device receives the CR message designated to it from the eNodeB, it completes the RA. Otherwise, it regards its RA attempt as a collision, and then tries to access again based on a predefined backoff time. The typical values of the backoff time are 8, 16, 32, and 64 (msec). Throughout this paper, activated MTC devices and those in retransmissions are called *backlogged.*

As MTC device traffic model, we assume that there are a total of N MTC devices and M RAPs available. It is notable that our proposed algorithms do not need to know the population size N. We further assume that the MTC devices are activated at time $x \in (0, T_A)$ whose probability density function (PDF) $f_X(x)$ is Beta distribution with parameters $\alpha = 3$ and $\beta = 4$ in [6]:

$$f_X(x) = \frac{x^{\alpha-1}(T_A - x)^{\beta-1}}{T_A^{\alpha+\beta-1} B(\alpha, \beta)} \tag{1}$$

where $B(\alpha, \beta)$ denotes the Beta function $\int_0^1 x^{\alpha-1}(1-x)^{\beta-1}dx$. If T_A consists of I_A slots, the expected number of newly activated MTC devices in the ith slot is given by $\lambda_i = N\int_{t_{i-1}}^{t_i} f_X(x)dx$ for $i = 1, 2, \cdots, I_A$. In addition to the Beta distribution, we also consider the uniform distribution in $(0, T_A)$ for the activation of the MTC devices [4].

III. Pseudo-Bayesian Access Class Barring

We introduce the overall procedure of PB-ACB and EPB-ACB algorithms first in Section III-A. Section III-B and C show the detailed derivations on an optimal ACB factor and estimation on backlogged MTC devices used in the algorithms, respectively.

A. Proposed Algorithms

Let p_t and ν_t denote the ACB factor and the estimated mean number of backlogged MTC devices in an eNodeB, respectively, where the subscript t denotes a slot index. In each slot, the eNodeB observes the number of RAPs not transmitted, denoted by r, among a total of M RAPs.

The underlying mechanism of the two proposed algorithms is first to estimate the mean number of backlogged MTC devices, ν_{t-1}, based on the *a priori* distribution, denoted by $\mathbb{P}(n, \nu_{t-1})$, for n backlogged MTC devices and then broadcast the ACB factor. After backlogged MTC devices react, the eNodeB observes r RAPs not transmitted and construct the *a posteriori* distribution, denoted by $\mathbb{P}(n|r, \nu_{t-1})$, such that its mean can be newly estimated, i.e., $\nu'_{t-1} = \mathbb{E}[n|r, \nu_{t-1}] = \sum_{n=0}^{\infty} n\mathbb{P}(n|r, \nu_{t-1})$. The eNodeB makes a correcting offset for ν_{t-1}, denoted by $\Delta\nu = \nu'_{t-1} - \nu_{t-1}$ (positive or negative). Thus, at each slot, with r RAPs not transmitted, the eNodeB can get the correcting offset $\Delta\nu$ in the third line of Algorithm 1 and makes a correction on the previous estimation, ν_{t-1}. Furthermore, if $\Delta\nu_t$ is a positive value, the eNodeB interprets it as an increase in the number of backlogged MTC devices due to new arrivals. Thus, newly activated MTC devices are estimated as

$$a_t = \max(0, \Delta\nu_t). \tag{2}$$

Moreover, at each slot, as introduced in Step 3, the eNodeB declares the number of successfully transmitted connection request messages, denoted by c_t in the fifth line of PB-ACB. Note again that c_t does not indicate the number of successfully transmitted RAPs at time t, but those RAPs transmitted $\tau_1 + \tau_2 + \tau_3$ slots before t while being declared successful transmissions at time t based on the connection request message. Finally, the ACB factor p_t is broadcasted for the next slot.

Let us introduce the difference between PB-ACB and EPB-ACB algorithms. Suppose that M be very large. Then, sufficient statistical information may be conveyed via r such that PB-ACB algorithm can estimate accurately the number of backlogged MTC devices. However, if M is small, they may not work well. For an extreme example, suppose $M = 1$. Then, the eNodeB can only observe either $r = 0$ or $r = 1$. If the RAP is busy ($r = 0$), then $\Delta\nu = 0.582$. However, for bursty arrivals of MTC devices, such an increment on ν_{t-1} may not be sufficient to keep track of the number of backlogged MTC devices. Hence, in order to take into account such bursty arrivals we need a boosting factor for $\Delta\nu$. The EPB-ACB algorithm introduces the boosting factor, k_t: when a positive $\Delta\nu$ is observed over some consecutive slots, k_t keeps track of it with $k_t = k_{t-1} + 1$. Then, ν_{t-1} is corrected as $\nu_{t-1} = \nu_{t-1} + k_t \cdot \Delta\nu$ such that ν_{t-1} could reflect bursty increase in newly activated MTC devices.

B. Optimal ACB Factor

This section derives an optimal ACB factor, which is used as p_t in Step 7 of PB-ACB and Step 12 of EPB-ACB algorithm. It shall be obtained as maximizing the average number of successful RAP transmissions over the estimated mean number of backlogged MTC devices. To do this, it is needed to estimate the mean number of backlogged MTC devices. However, it is not known in advance how many active MTC devices are; we presume that the number of backlogged MTC devices follows a Poisson distribution with mean ν as the *a priori* distribution as

$$\mathbb{P}(n,\nu) = \frac{\nu^n}{n!}e^{-\nu} \tag{3}$$

where $\mathbb{E}[n] = \nu$.

For a backlogged MTC device to transmit an RAP successfully, say RAP k, other backlogged MTC devices should not choose the RAP. This probability, denoted by $P_S(S_k)$, can be expressed as

$$P_S(S_k) = \sum_{n=0}^{\infty}\sum_{m=0}^{n} P_S(S_k|m)P_A(m|n)\mathbb{P}(n,\nu) \tag{4}$$

where $P_S(S_k|m)$ and $P_A(m|n)$ denote respectively the probability that RAP k is selected by a single MTC device given that m MTC devices pass the ACB check, and the probability that m MTC devices pass the ACB check given n backlogged MTC devices, each of which is obtained as

$$P_S(S_k|m) = \binom{m}{1}\frac{1}{M}\left(1-\frac{1}{M}\right)^{m-1} \tag{5}$$

and

$$P_A(m|n) = \binom{n}{m}p^m(1-p)^{n-m}. \tag{6}$$

After plugging (5) and (6) into (4), we can get $P(S_k)$ as

$$P_S(S_k) = \frac{p\nu}{M}e^{-\frac{p\nu}{M}}. \tag{7}$$

The average number of backlogged MTC devices that successfully access RAP k is obtained as

$$\mathbb{E}[S_k] = 1\cdot P_S(S_k) + 0\cdot(1-P_S(S_k)) = \frac{p\nu}{M}e^{-\frac{p\nu}{M}}. \tag{8}$$

Thus, the average number of backlogged MTC devices that succeed in RAP transmissions over the total M available RAPs is obtained by

$$\mathbb{E}\left[\sum_{i=1}^{M} S_k\right] = \sum_{i=1}^{M}\mathbb{E}[S_k] = p\nu e^{-\frac{p\nu}{M}} \tag{9}$$

where the symmetry of the RAPs are applied. After taking the first derivative of (9) with respect to p, we have the optimal ACB factor p^* i.e.,

$$p^* = \min(1, M/\nu). \tag{10}$$

C. Update Rule for ν

We are now in a position to find $\mathbb{P}(n|r,\nu_{t-1})$, from which $\mathbb{E}[n|r,\nu_{t-1}]$ is obtained; the observation that r RAPs have not been transmitted enables us to estimate the conditional probability of n backlogged MTC devices given r RAPs not transmitted in a Bayesian manner. To carry on, let us denote by $P_R(n,r)$ and $P_R(r)$ respectively the joint probability that r RAPs are not transmitted while there are n backlogged MTC devices, and the probability that r RAPs are not transmitted, i.e., $P_R(r) = \sum_{n=0}^{\infty} P_R(n,r)$. We can get $\mathbb{P}(n|r,\nu_{t-1})$ as

$$\mathbb{P}(n|r,\nu_{t-1}) \Rightarrow \mathbb{P}(n|r,\nu) = \frac{P_R(n,r)}{P_R(r)} \tag{11}$$

where although $P_R(n,r)$ and $P_R(r)$ depend on ν, we suppress it for notational simplicity.

First, we can write $P_R(n,r)$ as

$$P_R(n,r) = \left\{\sum_{m=0}^{n} P_I(r|m)P_A(m|n)\right\}\cdot\mathbb{P}(n,\nu). \tag{12}$$

Note that $P_A(m|n)$ is given in (6), whereas $P_I(r|m)$ is the probability that r RAPs are not chosen (such that they are not transmitted) amongst M RAPs after m backlogged MTC devices pass the ACB check and choose some RAPs. To find $P_I(r|m)$, let $P_e(E_k)$ denote the probability that RAP k for $k \in \{1,2,\ldots,M\}$ is not chosen, which is obtained as $P_e(E_k) = (1-1/M)^m$. Then, the probability that *particular* r RAPs, $(k_1, k_2, \cdots, k_r)$, are not chosen is obtained as

$$P_e(\underbrace{E_{k_1}\cap\cdots\cap E_{k_r}}_{r\text{ events}}) = \left(1-\frac{r}{M}\right)^m. \tag{13}$$

Additionally, let us denote by $S_r(M)$ the probability that *any* r RAPs are found not chosen out of M RAPs. Since the number of ways to choose r RAPs randomly amongst M RAPs is $\binom{M}{r}$, $S_r(M)$ is obtained by

$$S_r(M) = \binom{M}{r}\left(1-\frac{r}{M}\right)^m. \tag{14}$$

We then write $P_I(r|m)$ as

$$P_I(r|m) = S_r(M)(1-P_e(\underbrace{E_{k_1}\cup\cdots\cup E_{k_{M-r}}}_{M-r\text{ events}})) \tag{15}$$

where $P_e(E_{k_1}\cup\cdots\cup E_{k_{M-r}})$ denotes the probability that at least one RAP is not chosen given the remaining $M-r$ RAPs. Therefore, its complement means that all the $M-r$ RAPs must be chosen. Based on the inclusion-exclusion principle, $P(E_{k_1}\cup\cdots\cup E_{k_{M-r}})$ is found by

$$P_e(E_{k_1}\cup\cdots\cup E_{k_{M-r}}) = \sum_{i=1}^{M-r}(-1)^{i+1}S_i(M-r). \tag{16}$$

Substituting (17) into (16) yields

$$\begin{aligned} P_I(r|m) &= \binom{M}{r}\left(1-\frac{r}{M}\right)^m\left(1-\sum_{i=1}^{M-r}(-1)^{i+1}S_i(M-r)\right) \\ &= \binom{M}{r}\sum_{i=0}^{M-r}(-1)^i\binom{M-r}{i}\left(1-\frac{r+i}{M}\right)^m. \end{aligned} \tag{17}$$

Accordingly, by applying (12) and (17), $P_R(r)$ in (11) is obtained as (18) as shown at the bottom of the next page.

Finally, after a lengthy manipulation, we can get

$$\mathbb{E}[n|r] = \frac{\sum_{n=0}^{\infty} nP_R(r,n)}{P_R(r)}$$
$$= \nu\left[(1-p) + p\left(1-\frac{r}{M}\right)\left(1-e^{-\frac{p\nu}{M}}\right)^{-1}\right] \tag{19}$$

where the numerator $\sum_{n=0}^{\infty} nP_R(r,n)$ is obtained as (20), shown at the bottom of the page. Under the assumption that n backlogged MTC devices employ the optimal ACB factor p^*, we can get $\Delta\nu$ as

$$\Delta\nu = \mathbb{E}[n|r,\nu_{t-1}] - \nu_{t-1} \approx E[n|r] - \nu$$
$$= \nu p^*\left(e^{-\frac{p^*\nu}{M}} - \frac{r}{M}\right)\left(1-e^{-\frac{p^*\nu}{M}}\right)^{-1}\Bigg|_{p^*=\frac{M}{\nu}}$$
$$= 0.582\cdot M - 1.582\cdot r. \tag{21}$$

IV. Numerical Results

Extensive computer simulations are performed to illustrate the performance of the proposed algorithms. We assume that the RA slots is equal to $\tau_1 = 1$ (msec) [10], and that the eNodeB is able to recognize the success of a certain RAP transmission $\tau_1+\tau_2+\tau_3 = 7$ slots after receiving the RAP. Additionally, the CR timer is set to 8 (msec), and the MTC devices in collisions shall reattempt a RAP transmission after 15 slots.

Note that the existing algorithms in [9] and [7] are assumed to exactly know N, T_A, M, the number of idle RAPs and the number of success RAPs (or the number of collided RAPs) in each slot instantaneously. On the other hand, our proposed algorithms only utilize the information of M and the number of idle RAPs. Since the number of success RAPs is not available in practice, it is not possible to fairly compare our algorithms with them. Instead, the proposed algorithms are compared to an ideal ACB scheme [9], where the eNodeB knows the number of backlogged MTC devices exactly so that it can broadcast the optimal ACB factor every slot. The proposed PB-ACB and EPB-ACB algorithms can be considered as a realization of the concept of the

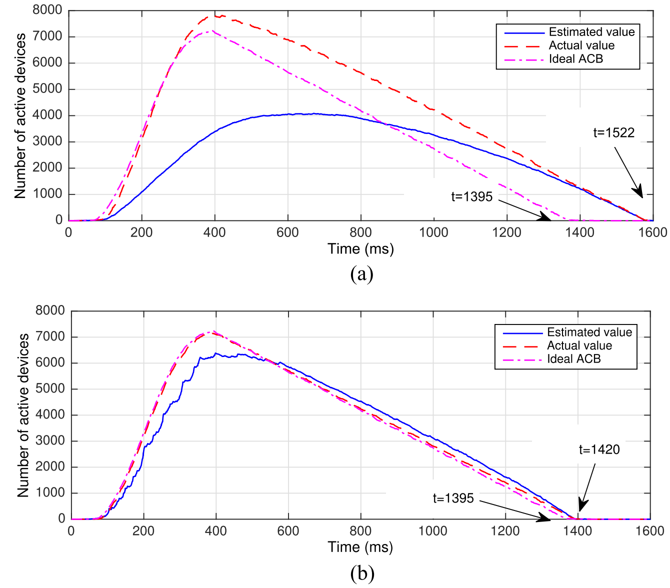
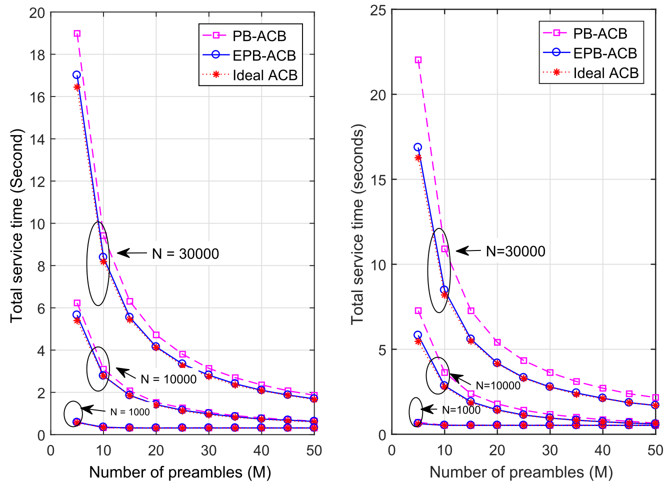


Fig. 1. Number of active MTC devices over time when $N = 10000$ and $M = 20$. (a) PB-ACB. (b) EPB-ACB.

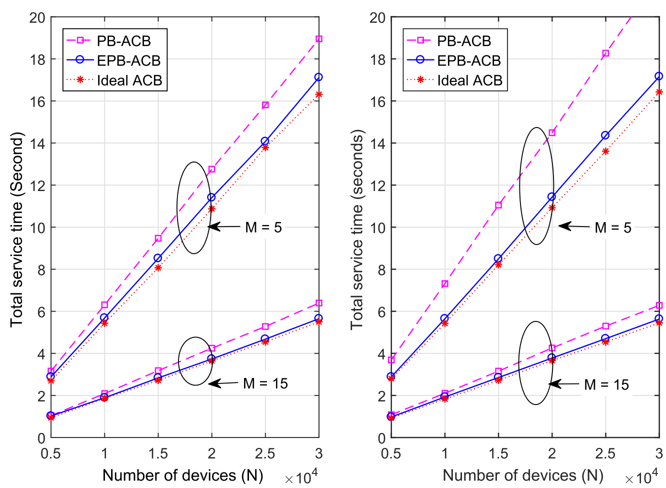
$$P_R(r) = \sum_{n=0}^{\infty}\sum_{m=0}^{n}\sum_{i=0}^{M-r}(-1)^i\binom{M}{r}\binom{M-r}{i}\left(1-\frac{r+i}{M}\right)^m\binom{n}{m}p^m(1-p)^{n-m}\frac{\nu^n}{n!}e^{-\nu}$$
$$= \sum_{n=0}^{\infty}\sum_{i=0}^{M-r}(-1)^i\binom{M}{r}\binom{M-r}{i}\left(1-\frac{r+i}{M}\right)^m\sum_{m=0}^{n}\binom{n}{m}\left(\left(1-\frac{r+i}{M}\right)p\right)^m(1-p)^{n-m}\frac{\nu^n}{n!}e^{-\nu} \tag{18}$$
$$= \binom{M}{r}\sum_{i=0}^{M-r}(-1)^i\binom{M-r}{i}e^{-\frac{(r+i)p\nu}{M}} = \binom{M}{r}e^{-\frac{p\nu}{M}\cdot r}\sum_{i=0}^{M-r}\binom{M-r}{i}\left(-e^{-\frac{p\nu}{M}}\right)^i = \binom{M}{r}e^{-\frac{p\nu}{M}\cdot r}\left(1-e^{-\frac{p\nu}{M}}\right)^{M-r}.$$

$$\sum_{n=0}^{\infty} nP_R(r,n) = \sum_{n=0}^{\infty}\sum_{m=0}^{n}\sum_{i=0}^{M-r} n\cdot(-1)^i\binom{M}{r}\binom{M-r}{i}\left(1-\frac{r+i}{M}\right)^m\binom{n}{m}p^m(1-p)^{n-m}\frac{\nu^n}{n!}e^{-\nu}$$
$$= \sum_{i=0}^{M-r}(-1)^i\binom{M}{r}\binom{M-r}{i}\left(\frac{(M-(r+i)p)\nu}{M}\right)\sum_{n=0}^{\infty}\left(\frac{(M-(r+i)p)\nu}{M}\right)^{n-1}\frac{1}{(n-1)!}e^{-\nu}$$
$$= \binom{M}{r}e^{-\frac{rp\nu}{M}}\sum_{i=0}^{M-r}(-1)^i\binom{M-r}{i}\left(\frac{(M-rp)\nu}{M} - \frac{ip\nu}{M}\right)e^{-\frac{ip\nu}{M}}$$
$$= \binom{M}{r}e^{-\frac{rp\nu}{M}}\left[\frac{(M-rp)\nu}{M}\sum_{i=0}^{M-r}\binom{M-r}{i}\left(-e^{-\frac{p\nu}{M}}\right)^i - \frac{p\nu}{M}\sum_{i=0}^{M-r} i\binom{M-r}{i}\left(-e^{-\frac{p\nu}{M}}\right)^i\right]$$
$$= \binom{M}{r}e^{-\frac{p\nu}{M}\cdot r}\left[1-e^{-\frac{p\nu}{M}}\right]^{M-r}\nu\left[(1-p)+p\left(1-\frac{r}{M}\right)\left(1-e^{-\frac{p\nu}{M}}\right)^{-1}\right]. \tag{20}$$



(a) (b)

Fig. 2. Total service time for varying the number of RAPs when $T_A =$ 500 ms. (a) Beta. (b) Uniform.



(a) (b)

Fig. 3. Total service time for varying the population size when $T_A = 500$ ms. (a) Beta. (b) Uniform.

ideal ACB algorithm by estimating the number of active MTC devices in each slot. If our proposed algorithms yield similar performances to the ideal ACB algorithm, then we can confirm the effectiveness of the proposed algorithms. The performance of the ideal ACB algorithm was considered as the upper bound in [9] and [7].

Fig. 1 shows the number of backlogged MTC devices over time, when $N = 10000$ and $M = 20$. The activation time T_A is set to 500 ms and the distribution of activation pattern has the Beta distribution. In Fig. 1, t denotes the total service time taken for all the MTC devices to make a successful RAP transmission. When we observe the actual and estimated number of backlogged MTC devices, there exists a certain gap between them. Such a gap seems large especially in bursty (congestion) regime with PB-ACB algorithm, while the estimation made by EPB-ACB algorithm keeps track of the actual value quite well.

Figs. 2 and 3 show the total service time when both the number of RAPs, M, and the population size of MTC devices, N, vary; the activation time T_A is still set to 500 ms and the activation patter has the Beta and Uniform distributions. It is found that both the PB-ACB and EPB-ACB algorithms show the performance close to the ideal ACB algorithm, while the total service time decreases as M increases; EPB-ACB algorithm yields a better performance than PB-ACB algorithm in all cases. In Fig. 3, the performance gab between PB-ACB algorithm and the ideal ACB algorithm becomes smaller as M increases up to $M = 15$, while EPB-ACB algorithm outperforms PB-ACB algorithm and show almost the performance of the ideal ACB algorithm.

V. Conclusion

In this paper, two ACB algorithms were proposed to efficiently resolve the congestion problem in large-scale M2M communications in LTE systems. Different from the conventional ACB algorithms, the proposed algorithms utilize information on the number of *idle* RAPs, which is available in practical systems. Extensive simulations show that the proposed algorithms result in a quite close performance to the ideal ACB algorithm that assumes perfect information on the number of active MTC devices.

References

[1] M. Hasan, E. Hossain, and D. Niyato, “Random access for machine-to-machine communication in LTE-advanced networks: Issues and approaches,” *IEEE Commun. Mag.*, vol. 51, no. 6, pp. 86–93, Jun. 2013.

[2] K. S. Ko, M. J. Kim, K. Y. Bae, D. K. Sung, J. H. Kim, and J. Y. Ahn, “A novel random access for fixed-location machine-to-machine communications in OFDMA based systems,” *IEEE Commun. Lett.*, vol. 16, no. 9, pp. 1428–1431, Sep. 2012.

[3] H. S. Jang, S. M. Kim, K. S. Ko, J. Cha, and D. K. Sung, “Spatial group based random access for M2M communications,” *IEEE Commun. Lett.*, vol. 18, no. 6, pp. 961–964, Jun. 2014.

[4] Third-Generation Partnership Project, “Study on RAN improvements for machine-type communications,” 3GPP, TR 37.868 V11.0.0, Oct. 2011.

[5] H. Wu, C. Zhu, R. J. La, X. Liu, and Y. Zhang, “FASA: Accelerated S-ALOHA using access history for event-driven M2M communications,” *IEEE/ACM Trans. Netw.*, vol. 21, no. 6, pp. 1904–1917, Dec. 2013.

[6] S. Duan, V. Shah-Mansouri, and V. W. S. Wong, “Dynamic access class barring for M2M communications in LTE networks,” in *Proc. IEEE GLOBECOM*, Dec. 2012, pp. 4747–4752.

[7] M. Tavana, V. Shah-Mansouri, and V. W. S. Wong, “Congestion control for bursty M2M traffic in LTE networks,” in *Proc. IEEE Int. Conf. Commun.*, Jun. 2015, pp. 5815–5820.

[8] Z. Wang and V. W. S. Wong, “Optimal access class barring for stationary machine type communication devices with timing advance information,” *IEEE Trans. Wireless Commun.*, vol. 14, no. 10, pp. 5374–5387, Oct. 2015.

[9] S. Duan, V. Shah-Mansouri, Z. Wang, and V. 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, Dec. 2016.

[10] A. Laya, L. Alonso, and J. Alonso-Zarate, “Is the random access channel of LTE and LTE-A suitable for M2M communications? A survey of alternatives,” *IEEE Commun. Surveys Tut.*, vol. 16, no. 1, pp. 4–16, Jan. 2014.