Fast Retrial for Low-Latency Connectivity in MTC with Two Different Types of Devices

Jinho Choi

Abstract—In this paper, we consider co-existing two different types of devices in machine-type communication (MTC), namely type-1 and type-2 devices, where type-1 devices need short access delay for low-latency requirements, while type-2 devices are delay-tolerant. For short access delay, we study the use of fast retrial in preamble transmissions when a group of preambles is divided into two subsets to support two different types of devices. Stability conditions are derived using Foster-Lyapunov criteria in terms of arrival rates, the number of preambles, and the number of type-1 devices. We also propose an adaptive algorithm that dynamically decides the minimum number of preambles for type-1 devices under stability conditions.

Index Terms—MTC; Random Access Delay; Fast Retrial; Stability

I. INTRODUCTION

Machine-type communication (MTC) plays a key role in supporting various Internet-of-Things (IoT) applications within cellular systems [1] [2]. Since it is expected to support a large number of devices with sporadic traffic in MTC, random access is considered. In particular, slotted ALOHA (which is known to originally stand for Additive Links On-line Hawaii Area, but is now used to mean its random access scheme) is widely studied for MTC [3].

In MTC, for random access, an active device (that has data to send) is to transmit a preamble that is randomly selected from a pool of preambles, which is shared by all devices. Since the size of the preamble pool is finite, there exist preamble collision, which happens if multiple devices choose the same preamble, and the performance depends on the size of the preamble pool. Thus, as in [4], it is considered to adaptively adjust the size of preamble pool depending on the devices’ activity, while access class barring (ACB) is usually used for access control with a fixed size of preamble pool in MTC [5].

There can be multiple types of devices with different requirements. In this paper, we consider two different types of devices, namely type-1 and type-2 devices (type-1 devices are delay-sensitive, while type-2 devices are delay-tolerant), that co-exist in a system and share a pool of preambles. The main contribution of the paper is two-fold: i) to support type-1 devices, fast retrial [6] is applied to preamble transmissions for short access delay (without reserving preambles [7]) and a sufficient condition for the stability (i.e., condition for a finite access delay) is derived; ii) an adaptive algorithm to stabilize fast retrial for type-1 devices is proposed. Note that fast retrial is also applied to MTC in [8], where access control is employed to stabilize fast retrial. On the other hand, in this paper, access control (which may not be applicable to delay-sensitive devices to meet their requirements) is not used, but dynamic resource allocation is considered. Furthermore, unlike [9] [10] [11], the size of preamble pool for delay-sensitive devices is dynamically adjusted without knowing the traffic intensity (i.e., arrival rate) of type-1 devices. Thus, the proposed adaptive algorithm can be used when the traffic intensity of type-1 devices is unknown or varying.

II. SYSTEM MODEL

In this section, we consider a random access system that consists of one base station (BS) and two different types of devices that share a group of preambles.

A. Two Different Types of Devices

Throughout the paper, we consider the case that two different types of devices, namely types-1 and 2, co-exist. Type-1 devices need short access delay for low-latency requirements, while type-2 devices are delay-tolerant. As in [11], the number of type-1 devices is usually much smaller than that of type-2 devices, while type-1 devices need more resources to meet short access delay requirements.

A handshaking process as in [1] [2] is considered with a group of preambles to support two different types of devices. In particular, we assume that a group of preambles is divided into two subsets to support two different types of devices as illustrated in Fig. 1. Denote by \( L_1 \) and \( L_2 \) the pools of preambles for type-1 and 2 devices, respectively. Let \( L_i = |\mathcal{L}_i|, i = 1, 2 \). Furthermore, let \( L = L_1 + L_2 \), which is the total number of preambles. It is assumed that the BS can adaptively decide the sizes of the preamble pools. Thus, \( L_1 \in \{1, \ldots, L - 1\} \), while \( L_2 = L - L_1 \).

With a finite size of pool, since the BS may not be able to detect some transmitted preambles due to preamble collisions, it is required for a device to re-transmit preambles until a preamble is successfully received by the BS according to a re-transmission strategy, which results in the access delay. Thus, for type-1 devices, it is important to shorten the access delay for low-latency requirements.

To support type-1 devices with short access delay, various approaches can be considered. To this end, in [9], with two different types of devices, namely delay-sensitive and delay-tolerant devices (which might be equivalent to type-1 and type-2 devices, respectively, in this paper), \( L_1 \) can be dynamically adjusted to minimize access delay for delay-sensitive devices. In this paper, as mentioned earlier, \( L_1 \) is also to be dynamically adjusted for short access delay. In addition,
we consider fast retrial [6] for type-1 devices with their low-latency requirements. 

Note that in [9], fast retrial is implicitly employed for preamble (re-)transmissions. However, no stability is studied, while we will derive stability conditions in Section III.

B. Fast Retrial for Preamble Transmissions

For short access delay, an active type-1 device experiencing preamble collision can immediately re-transmit another randomly selected preamble in $L_1$ in the next time slot without waiting for any back-off time based on fast retrial [6].

In Fig. 2, we illustrate fast retrial with $L_1 = 4$ preambles. At slot $t$, suppose that devices 1 and 3 transmit preamble 1, which results in preamble collision. At the next time slot, i.e., slot $t+1$, the two devices re-transmit randomly selected preambles (preamble 2 for device 1 and preamble 4 for device 3), while a new active device, i.e., device 2, transmits preamble 1. In this case, all the devices can successfully transmit preambles. This shows that immediate re-transmissions by fast retrial may not lead to successive preamble collision, and shorten the access delay (due to no back-off time).

At the $n$th type-1 device, the state of queue is updated as follows:

$$q_n(t + 1) = (q_n(t) + a_n(t) - s_n(t))^+,$$  \hspace{1cm} \text{for } n = 1, \ldots, N_1, \hspace{1cm} (1)$$

where $q_n(t)$, $a_n(t)$, and $s_n(t)$ are the length of queue, the number of new arrivals (of access request), and the number of successful preamble transmissions of the $n$th type-1 device at slot $t$, respectively. Here, $(x)^+ = \max\{0, x\}$. Note that a type-1 device becomes active if its queue is not empty, i.e., $q_n(t) > 0$, with fast retrial.

We assume that $a_n(t)$ is independent and identically distributed (iid) with a finite mean as follows:

$$\lambda_n = \mathbb{E}[a_n(t)] < \infty,$$ \hspace{1cm} (2)$$

where $\lambda_n$ is the average arrival rate and $\mathbb{E}[\cdot]$ represents the statistical expectation.

Since each active device randomly chooses a preamble from the pool, it can be shown that

$$s_n(t) = \begin{cases} 1, & \text{w.p. } p_n(t) \\ 0, & \text{w.p. } 1 - p_n(t) \end{cases}$$ \hspace{1cm} (3)$$

where $p_n(t) = \left(1 - \frac{1}{L_1}\right)^{K_1(t) - 1}$ is the conditional probability of no preamble collision or successful preamble transmission when there are $K_1(t)$ active type-1 devices. Furthermore, it can be shown that

$$K_1(t) = |\{n | q_n(t) > 0\}| \leq N_1,$$ \hspace{1cm} (4)$$

because a type-1 device becomes active when its queue is not empty as mentioned earlier. As a result, $q(t) = [q_1(t) \ldots q_{N_1}(t)]^T$ is a Markov chain, where $q(t) \in \mathbb{Z}_0^{N_1}$. Here, $\mathbb{Z}_0 = \{0, 1, \ldots\}$, i.e., $\mathbb{Z}_0$ represents the set of non-negative integers. Since a large $q_n(t)$ means a long queuing delay for a type-1 device, it is necessary to avoid. To find the conditions for stable queues, we can consider Foster-Lyapunov criteria [12] [13].

Lemma 1: If

$$\frac{1}{N_1} \sum_{n=1}^{N_1} \lambda_n < \left(1 - \frac{1}{L_1}\right)^{N_1 - 1},$$ \hspace{1cm} (5)$$

$q(t)$ is positive recurrent.

Proof: Let $V(q(t)) = \sum_{n=1}^{N_1} q_n(t)$ be a Lyapunov function. Consider the drift that is defined as

$$D(q) = \mathbb{E}[V(q(t + 1)) - V(q(t))] | q(t) = q].$$ \hspace{1cm} (6)$$

Let $Q_1 = \{q | q_n \geq 1, \hspace{0.2cm} n = 1, \ldots, N_1\}$, and $\bar{Q}_1 = \mathbb{Z}_0^{N_1} \setminus Q_1$ be the complement of $Q_1$. Thus, for any $q$, we have $q \in Q_1 \cup \bar{Q}_1$. Note that $\bar{Q}_1$ is a finite set. For any $q \in Q_1$, we have

$$p_n(t) = \left(1 - \frac{1}{L_1}\right)^{|q(t)||_0 - 1} = \left(1 - \frac{1}{L_1}\right)^{N_1 - 1},$$ \hspace{1cm} (7)$$

where $||\cdot||_p$ denotes the $p$-norm. Thus, for any $q \in Q_1$, it can be shown that

$$D(q) = \sum_n \mathbb{E}[a_n(t) - s_n(t)]$$

$$= \sum_n \lambda_n - p_n(t) = \sum_n \lambda_n - \left(1 - \frac{1}{L_1}\right)^{N_1 - 1}. \hspace{1cm} (8)$$

III. STABILITY AND ADAPTIVE ALGORITHM

In this section, we will find stability conditions for type-1 devices based on Foster-Lyapunov criteria [12] [13]. In addition, an adaptive algorithm to decide $L_1$ is derived.

A. Stability

Denote by $N_1$ the number of type-1 devices. In addition, let $K_1(t)$ denote the number of active type-1 devices that send preambles at slot $t$.
Thus, according to (5), we have
\[ D(q) \leq -\epsilon, \quad q \in Q_1, \tag{9} \]
where \( \epsilon > 0. \)

For \( q \in Q(N_1) \), there is at least one empty queue (i.e., \( q_n(t) = 0 \)). For the case of empty queue, we have \( q_n(t+1) = a_n(t) - s_n(t) \). Thus, it can be shown that
\[
\mathbb{E}[q_n(t+1) - q_n(t) \mid q_n(t) = q_n] = \lambda_n (1 - p_n(t)) + \mathbb{E}[(a_n(t) - s_n(t))^+ | p_n(t)] \\
\leq \lambda_n (1 - p_n(t)) + \mathbb{E}[a_n(t)] p_n(t) = \lambda_n,
\]
which results in
\[
D(q) \leq \sum_n \lambda_n, \quad q \in Q_1.
\]

According to [12, Proposition D.1], (9) and (11) imply that \( q(t) \) is a positive recurrent Markov chain.

In (5), the right-hand side (RHS) term is the probability of no preamble collision under full loading (i.e., all \( N_1 \) type-1 devices transmit randomly selected preambles), which is the minimum probability of successful preamble transmission or the minimum departure rate. Thus, for a stable system, it implies that the average arrival rate on the left-hand side (LHS) has to be lower than or equal to the minimum departure rate.

Let \( \lambda_{max} \) be the maximum mean arrival rate for all type-1 devices so that \( \lambda_n \leq \lambda_{max} \). Then, from (5), it can be shown that
\[
\lambda_{max} < \left(1 - \frac{1}{L_1}\right)^{N_1-1} \leq \exp^{-\frac{N_1-1}{L_1}}.
\]
Clearly, from this, with \( \lambda_{max} \leq 1 \), it follows that
\[
N_1 \leq \bar{N}_1 \triangleq 1 + L \ln \frac{1}{\lambda_{max}}, \tag{13}
\]
where \( \bar{N}_1 \) is the maximum number of type-1 devices with stable queues or a finite access delay with fast retrial.

For a stable system of type-1 devices, it is important to decide the key parameters according to (13). Clearly, the number of type-1 devices has to be less or equal to \( \bar{N}_1 \). In addition, \( \lambda_{max} \) can be broadcast to all the type-1 devices so that their arrival rates cannot be greater than \( \lambda_{max} \).

B. Adaptive Algorithm for \( L_1 \)

Suppose that \( \lambda_n \leq \lambda_{max} \). Then, \( L_1 \) can be smaller than \( L \) with a stable system of type-1 devices so that \( L_2 = L - L_1 \) preambles can be assigned to type-2 devices. Thus, the minimum \( L \) that satisfies (5) has to be found. Unfortunately, since the \( \lambda_n \)'s may not be known to the BS and furthermore the arrival rate of each device can be time-varying, the BS needs to estimate \( \Lambda = \sum_n \lambda_n \) to find the minimum \( L \). To this end, we can consider an adaptive algorithm with an estimate of \( \Lambda \).

For convenience, let \( z = 1 - \frac{1}{L_1} \in [0,1) \) and consider the following function:
\[
f(z) = \frac{1}{N_1} (\Lambda z - z^{N_1})
\]
Clearly, it can be shown that \( f(z) \) is a concave function of \( z \) and its derivative becomes
\[
\frac{df(z)}{dz} = \frac{\Lambda}{N_1} - z^{N_1-1}.
\]
As a result, \( f(z) \) has the unique maximum and the solution is
\[
z^* = \arg\max_{0 \leq z < 1} f(z),
\]
which can be found by setting its derivative to zero. With \( z^* \), it can be readily shown that \( L_1^* = \frac{1}{1 - z^*} \) satisfies the equality in (5).

Recall that \( K_1(t) \) is the instantaneous number of active type-1 devices at slot \( t \). Provided that the queues are stable, the total mean departure rate has to be equal to the sum of new arrival rate and backlogged rate (which is the number of type-1 devices with collided preambles per slot). Let \( \Lambda_d \) and \( \Lambda_b \) denote the total means of departure and backlogged rates, respectively. Then, we have
\[
\Lambda_d = \Lambda + \Lambda_b.
\]

It can be shown that
\[
\Lambda_b = \mathbb{E} \left[ K_1(t) \left(1 - \left(1 - \frac{1}{L_1}\right)^{K_1(t)-1}\right)\right].
\]

With a sufficiently large \( N_1 \), we consider the following Poisson approximation for \( K_1(t) \):
\[
K_1(t) \sim \text{Pois}(0, \Lambda_d).
\]

Then, it follows that
\[
\Lambda_b = \Lambda_d - \Lambda_d e^{-\frac{\Lambda_d}{L_1}}
\]
Substituting (19) into (17), we have \( \Lambda = \Lambda_d e^{-\frac{\Lambda_d}{L_1}} \). Thus, an estimate of \( \Lambda \) is given by
\[
\hat{\Lambda} = K_1(t) e^{-\frac{\Lambda_d(t)}{L_1}},
\]
which leads to the following stochastic gradient ascent algorithm to find \( z^* \):
\[
z(t+1) = z(t) + \mu \frac{df(z)}{dz} \bigg|_{\Lambda=\hat{\Lambda},z=z(t)} \\
= z(t) + \frac{\mu}{N_1} \left(K_1(t) e^{-\frac{\Lambda_d(t)}{L_1}} - N_1 z(t)^{N_1-1}\right)
\]
where \( \mu \) is the step-size. Here,
\[
L_1(t) = \lceil\max\{1, \frac{1}{1 - z(t)}\}\rceil.
\]

IV. SIMULATION RESULTS

In this section, we present simulation results to see the performance of type-1 devices in terms of queue length. For simplicity, we assume that \( \lambda_n = \lambda \) for all \( n \).

In Figs. 3 (a) and (b), the average queue length, \( \mathbb{E}[q_n(t)] \), is shown as functions of \( \lambda \) (with \( N_1 = 30 \) and \( L_1 = 20 \)) and \( N_1 \) (with \( \lambda = 0.2 \) and \( L_1 = 20 \)), respectively. Since the access delay increases with queue length, we can see that \( \lambda \) is to be lower than its maximum, \( \left(1 - \frac{1}{L_1}\right)^{N_1-1} \), when \( L_1 \) and \( N_1 \) are fixed (as in Fig. 3 (a)) or \( N_1 \) is to be smaller than its maximum, \( 1 + L_1 \ln \frac{1}{\lambda} \) (as in Fig. 3 (b)) for stable systems.

1It can be easily shown that the second derivative of \( f(z) \) is greater than or equal to 0.
In this paper, we applied fast retrial to preamble transmissions for type-1 devices that require short access delay when two different types of devices co-exist. For stable fast retrial, stability conditions have been derived using Foster-Lyapunov criteria. In addition, an adaptive algorithm to decide the size of preamble pool was derived. To guarantee a certain access delay, access control can also be used together with the adaptation of the size of preamble pool, which might be a further research topic to be studied in the future.

V. CONCLUDING REMARKS

In this paper, we applied fast retrial to preamble transmissions for type-1 devices that require short access delay when two different types of devices co-exist. For stable fast retrial,