



Article

# Dynamic RACH Partition for Massive Access of Differentiated M2M Services

Qinghe Du <sup>1,2</sup>, Wanyu Li <sup>1</sup>, Lingjia Liu <sup>3</sup>, Pinyi Ren <sup>1,\*</sup>, Yichen Wang <sup>1</sup> and Li Sun <sup>1</sup>

<sup>1</sup> Department of Information and Communications Engineering, School of Electronic and Information Engineering, Xi'an Jiaotong University, Xi'an 710049, China; duqinghe@mail.xjtu.edu.cn (Q.D.); liwanyu@stu.xjtu.edu.cn (W.L.); wangyichen0819@mail.xjtu.edu.cn (Y.W.); lisun@mail.xjtu.edu.cn (L.S.)

<sup>2</sup> Shaanxi Smart Networks and Ubiquitous Access Research Center, Xi'an 710049, China

<sup>3</sup> Department of Electrical Engineering and Computer Science, the University of Kansas, Lawrence, KS 66045, USA; lingjialiu@ittc.ku.edu

\* Correspondence: pyren@mail.xjtu.edu.cn; Tel.: +86-29-8266-4427

Academic Editors: Yunchuan Sun, Antonio Jara and Shengling Wang

Received: 31 December 2015; Accepted: 17 March 2016; Published: 30 March 2016

**Abstract:** In machine-to-machine (M2M) networks, a key challenge is to overcome the overload problem caused by random access requests from massive machine-type communication (MTC) devices. When differentiated services coexist, such as delay-sensitive and delay-tolerant services, the problem becomes more complicated and challenging. This is because delay-sensitive services often use more aggressive policies, and thus, delay-tolerant services get much fewer chances to access the network. To conquer the problem, we propose an efficient mechanism for massive access control over differentiated M2M services, including delay-sensitive and delay-tolerant services. Specifically, based on the traffic loads of the two types of services, the proposed scheme dynamically partitions and allocates the random access channel (RACH) resource to each type of services. The RACH partition strategy is thoroughly optimized to increase the access performances of M2M networks. Analyses and simulation demonstrate the effectiveness of our design. The proposed scheme can outperform the baseline access class barring (ACB) scheme, which ignores service types in access control, in terms of access success probability and the average access delay.

**Keywords:** M2M communications; overload control; preamble partition; access success probability; average access delay

## 1. Introduction

With the rapid development of cellular networks, the M2M communications, known as an indispensable part of Internet of Things (IoT) [1,2], have received more and more research attention. Generally, M2M communications provide a flexible solution for many scenarios in IoT. On the one hand, M2M in LTE, relying on cellular networks as the infrastructure, provides the long-distance access option for IoT, such that better coverage can be achieved. On the other hand, M2M communications may also offer the possibility for short-range applications.

In general, a M2M communication system includes a large number of machine-type communication (MTC) devices that can communicate without human intervention [3–9]. However, when the number of MTC devices trying to transmit data to the eNB (eNodeB) is considerably large within a very short period of time, the radio access network (RAN) overload issue will arise accordingly. In such a condition, the network congestion [4,5] inevitably increases delays, causes packet loss and even leads to service interruption. To alleviate network congestion caused by RAN overload, several schemes [6] were proposed and studied, among which the access class barring (ACB) [7] scheme is currently regarded as a simple and popular solution in M2M networks. The key idea of the ACB

scheme can be summarized as follows. Under the ACB scheme, the eNB broadcasts two parameters to all MTC devices. Based on the two parameters, each MTC device performs random access barring for itself. Particularly, each MTC generates a random number and compares the number with the threshold broadcast by the eNB. If the number is smaller than the threshold, it proceeds with the access attempt to the network; otherwise, it will back off for a random time period before attempting to access again. The ACB scheme is simple to implement, and thus, it receives wide attention.

Generally speaking, the ACB scheme has been shown to be beneficial when MTC devices can tolerate long access delays due to frequent collisions during random access. However, when there also exist delay-sensitive services, the ACB scheme might not work efficiently. In the literature, the extended access barring (EAB) scheme [8] takes the delay-sensitive devices into account. The basic idea of EAB can be summarized as: as long as EAB is activated in the case of network congestion, the delay-sensitive devices are enabled to send access requests while the delay-tolerant ones are disabled from doing so. This approach actually sacrifices the performance for delay-tolerant services. However, it is worth noting that in realistic M2M networks, the number of delay-sensitive services is far less than that of delay-tolerant ones. Clearly, benefiting delay-sensitive services too much would severely degrade the performances of delay-tolerant services and use the resource in a highly-inefficient way.

To address the aforementioned problems, this paper proposes an efficient scheme that performs dynamic allocation of the random access channel (RACH) resources to clustered MTC devices with differentiated delay requirements. Specifically, the proposed scheme can adjust the preamble partition ratio between the two given clusters. The original ACB scheme is further applied to each cluster for access attempts. The novelty and contribution of our work lie in the optimized partition solution, as well as the theoretical analyses. Simulation results show that compared to the baseline ACB scheme, the proposed scheme obtains significant improvement in access success probability and also achieves performance improvement in reducing access delay.

The remainder of this paper is organized as follows. Section 2 reviews the related work on existing research efforts and candidate solutions on M2M congestion control. Section 3 introduces the system framework and M2M traffic model. Section 4 proposes the dynamic RACH partition scheme in detail. Section 5 analyzes the performance for our proposed scheme. Then, Section 6 presents the simulation evaluations and compares our scheme with the baseline ACB scheme. Finally, the paper concludes with Section 7.

## 2. Related Work

Notably, the M2M communication system is a large-scale network with diverse applications and a massive number of interconnected machines. There are mainly two standards bodies pushing the standardization process of M2M communications: the Third Generation Partnership Project (3GPP) and the European Telecommunications Standards Institute (ETSI), which specified their respective M2M communication architectures. ETSI defined the service-oriented M2M network architecture that comprises the device-and-gateway domain and the network domain, but without the underpinning of particular wireless technologies [10]. Furthermore, the ETSI-M2M does not indicate the specifications for M2M area networks and details for the access and core networks. The 3GPP-M2M focuses on enhancing the cellular wireless networks. Consequently, the typical smartM2M and oneM2M architectures specified by ETSI are inclined to provide M2M services independent of the underlying networks. That is to say, the access overload issue is less serious in the ETSI-M2M architecture, although the management will be hard. In contrast, 3GPP categorizes MTC devices as a special type of cellular users with a low rate and priority, and the MTC devices need to connect to the base station for data transmission. Therefore, the centralized control is ready. However, when the population of MTC devices gets large, the congestion problem could be extremely severe. Since the applications of M2M networks typically require security and privacy, the centralized architecture still attracts the major attention. Consequently, in this paper, our efforts will be dedicated to M2M under the 3GPP architecture.

It is worth mentioning that in the early stage of proposals for MTC, direct connections between MTC devices, as well as multi-hop transmissions across MTC devices were also suggested [11]. However, these types of connections also often come with quality-of-service (QoS) requirements, such as delay requirements [12,13]. Then, these functions have been gradually carried out by the device-to-device (D2D) communications with cooperation and cognition capability [12–15], where adjacent devices can connect directly by reusing the cellular users' spectrum in either an underlay [12,16], an overlay [17] or a hybrid style [13]. Multi-hop communications over D2D networks has also been attracting research attention [16,18].

Recently, comprehensive studies have been launched over the last few years to explore the RAN overload issue for M2M communications [19]. As suggested in [20], a huge volume of signaling and data flow will be yielded, easily causing severe congestions in random access network. Similar to wireless sensor networks (WSN) [21], massive M2M devices with burst data within a short period of time may also produce massive accesses [22], which result in radio access networks' (RAN) overload issue. As M2M continues to burgeon rapidly, it is worth researching RAN overload control towards future ubiquitous IoT.

There are several research outcomes of solving the congestion problem [23]. Among all available solutions, ACB is recognized as an effective yet simple mechanism to regulate access in LTE/LTE-A networks [24,25]. Sixteen classes are defined, and several of them are reserved for high-priority cases. However, the access schemes for coexisting services with different priorities have not been specified. Towards this issue, several schemes have been proposed to adapt system parameters to the varying M2M network statuses. The work in [6] discussed several methods for modified ACB approaches: extended access barring (EAB), dynamic access barring (DAB) and cooperative ACB. The EAB scheme deals with differentiated services, which are divided into clusters with respective ACB parameters. However, different clusters still share the same access resources. The DAB scheme focuses on the dynamic adjustment of controlling parameters for ACB, while not addressing the differentiated services. We will show later that the partition of access preamble resources for differentiated services is the key to optimize the access performances, which have not been thoroughly studied in the literature.

Amokrane *et al.* [26] proposed a mechanism for congestion control in M2M networks. In this paper, congestion concerns both the radio access network and the mobile core network. The core idea of this work is to mitigate the MME/S-GW overload by rejecting the MTC traffic at the radio access network [26]. This work can reduce the amount of signaling for MTC devices and can satisfy the desired resource utilization ratio in the core network. Lien *et al.* [11] comprehensively discussed ubiquitous massive access via 3GPP M2M communications. This work proposed an effective solution to provide QoS guarantees to facilitate M2M applications with hard timing constraints. Lo *et al.* [27] proposed a self-optimizing overload control (SOOC) scheme that can configure RA resources according to the load condition. A typical feature of the SOOC scheme is that it can collect useful information via overload monitoring. Then, it can dynamically adjust the RA resources. However, this work did not present simulation or experiment results for performance evaluations. Aside from the above work, congestion and overload control in M2M communications was also actively discussed in [28]. Some applications based on M2M communications were described in [29,30].

The work in [31] proposed a prioritized random access for alleviating RAN overload, which takes advantage of the joint design combining the dynamic access barring and virtual resource allocation. Yang *et al.* [32] proposed a backoff adjustment scheme, which can improve the performances for low congestion levels. A code-expanded RA mechanism was developed by [33], where RA slots can be assembled in groups. This approach can increase the amount of contention resource at the cost of extra energy.

As elaborated above, extensive research has been dedicated to the RAN-level contention mechanism. In summary, the essence of RAN overload control mechanisms includes: dispersing the load of random access to different time slots, barring the random access behaviors and tuning system parameters based on MTC access traffic [34]. Existing research outcomes on the ACB mechanism

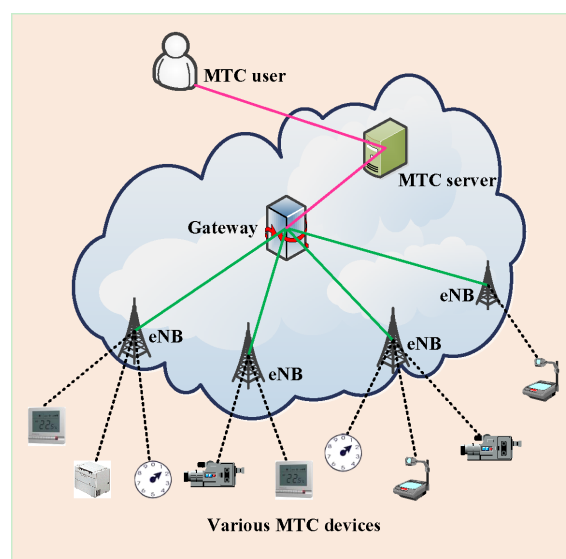
indicate that the ACB scheme is effective mainly for delay-tolerant devices. While for the EAB mechanism, although it takes delay-sensitivity into consideration and divides all devices into two groups, it cannot make full use of the preambles well. This is because it ignores the fact that the number of delay-sensitive services is far less than delay-tolerant ones in M2M networks. Moreover, it does not perform dynamic adjustment of preamble allocation to the two groups, and thus, it is hard to further enhance the performance. Note that in this paper, we use the term “group” and “cluster” for MTC devices exchangeably for presentation convenience.

### 3. System Model

This section firstly describes a typical M2M traffic model defined in 3GPP. Then, it briefly reviews the random access (RA) procedures in LTE-A. At last, the categories for M2M devices are specifically addressed.

#### 3.1. System Architecture

The network architecture of M2M communications in 3GPP is depicted in Figure 1, in which an MTC user can control various MTC devices via the MTC server. In the M2M network, various MTC devices access the eNB over wireless links. The eNB is responsible for collecting the data from MTC devices and forwards these data to the backhaul network and/or the Internet via a gateway for diverse applications. Notably, the M2M gateway ensures that M2M devices can interconnect with access networks. Clearly, the massive accesses of MTC devices to the eNB over wireless links form the major bottleneck and/or congestion in M2M networks, which, therefore, are the main focus of the work conducted in this paper. The work mainly concentrates on the multiple accesses by a large amount of MTC devices to the eNB. In order to help characterize the system architecture of the proposed scheme, here the motivation scenario in the case of the co-existence of delay-sensitive and delay-tolerant devices is specifically illustrated. In the scenario, delay-sensitive and delay-tolerant devices respectively correspond to one MTC application. The urban London scenario [31] is considered, and the delay-tolerant devices are from the application of smart meters. The delay-sensitive devices from the hospital e-care application are from the hospice beds from one of the largest hospitals in London. Obviously, the number of smart meters is quite huge, while the hospital e-care is very rare. When two types of services simultaneously attempt to access the network, the RAN overload issue appears. Meanwhile, the design of the resource allocation to realize the access control becomes very important, which is the work conducted in this paper.

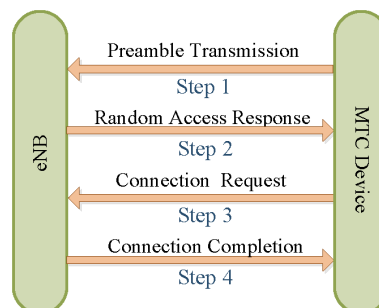


**Figure 1.** M2M communications in LTE-Advanced cellular networks towards IoT.

### 3.2. Random Access Procedures

For MTC devices, we consider the typical random access procedures defined in LTE protocols [35], where all MTC devices compete for the available wireless resources. It is worth noting that in M2M networks, the contention-based random access procedure usually applies for delay-tolerant access requests. However, contention-free opportunistic access is also supported by pre-allocating particular resources, which are for delay-constrained requests. Because the amount of devices sending request with strict delay demands is quite small compared to delay-tolerant ones and the total number of such MTC devices is often huge, here we focus on the typical contention-based random access procedure, which consists of four steps [35] between MTC devices and the eNB. Moreover, the resource pre-allocation approach is very inflexible, and thus, not able to provide the requests with fine-grained delay requirements. Next, we elaborate on the detailed random access procedures as follows.

When an MTC device attempts to access the network, it needs to send out an access request over the random access channel (RACH), which is comprised of several random access slots, which are used for the transmission of access requests. The length of the RA slot depends on the value of the configuration index. Relying on the existing protocols, the configuration index is valued as six, which means that in the RACH, there is an access opportunity every five milliseconds. In other words, the RACH finishes configuration every five milliseconds. Furthermore, there are in total 64 orthogonal available preambles. Only 54 of them are available for contention-based access, while the remaining 10 preambles are reserved for contention-free access. An access request is completed only if the four steps [31] are successfully finished, as shown in Figure 2 and Algorithm 1. However, note that the massive access attempts by sending the preambles occur in Step 1, which cause the majority of collisions. Thus, in this paper, we mainly concentrate on Step 1 to design the access control schemes.



**Figure 2.** Random access procedures between the M2M device and eNodeB (eNB) in LTE.

---

#### Algorithm 1 : Four steps of random access.

---

1: **Step 1:** *Preamble transmission*

Once an MTC device launches an access request to the RACH, it firstly selects a preamble of the RACH to transmit an access request. Under this condition, if two or more devices select the same preamble during the same slot, such that eNB is unable to decode any of the preambles, a collision occurs.

2: **Step 2:** *Random access response (RAR)*

For each successfully-decoded preamble, the eNB computes an identifier and then transmits an RAR to the UE devices.

3: **Step 3:** *Connection request*

The M2M device transmits a connection request message with a UE identifier to the eNB.

4: **Step 4:** *Contention completion*

Upon reception of a connection request in Step 3, the eNB transmits a contention resolution message as an answer to Step 3. Therefore, a device that does not receive Step 4 indicates a failure in the contention completion and requests a new access attempt.

---

Before proceeding further, we need to explain the fundamental mechanism of the ACB mechanism and to define the corresponding parameters used in this paper. The ACB scheme [11] was proposed for random access control of MTC devices. Particularly, the eNB broadcasts two parameters: an access barring factor, denoted by  $ac\_barringfactor$ , and an access barring time, denoted by  $ac\_barringtime$ . Each MTC device attempting to access the network generates a uniformly-distributed random variable  $q$ ,  $0 \leq q \leq 1$ . If  $q \leq ac\_barringfactor$ , the MTC device continues with the random access procedures. Otherwise, it is barred for a random time duration based on  $ac\_barringtime$  by using Equations (24) and (25) to retry the access.

### 3.3. Categories of M2M Devices

According to the comprehensive research [36], in the real scene where delay-sensitive and delay-tolerant devices coexist, although the occurrence of delay-sensitive devices, such as the hospital e-care, is very rare, the delay-sensitive devices are strict with delay, and they need instant processing. For the application in hospital e-care, the maximum tolerant delay is five milliseconds, since in a real system, the data become useless after that. On the other hand, delay-tolerant devices, such as smart grids [11], can tolerate several seconds or even minutes, and a great majority of devices is subject to this category.

## 4. Resource Partition Scheme for M2M Networks

The proposed scheme solves the RAN overload problem in the scenarios of the co-existence of delay-sensitive and delay-tolerant devices. In general, the proposed scheme comprehensively takes the resource allocation and the access control into account for the scenarios where two types of services co-exist in M2M networks. Specifically, the proposed architecture is composed of several main moves: firstly, in order to effectively provide quality of service (QoS) for two types of MTC devices mentioned above, we consider that MTC devices are classified into two clusters on the basis of delay requirements. Secondly, the vital conceptual design is achieved by dynamically adjusting the preamble partition between two given clusters. Thirdly, devices from two clusters adopt the ACB mechanism to access the network, respectively.

According to the proposed principle discussed above, as depicted in Figure 3, this paper presents the implementations at length afterwards.

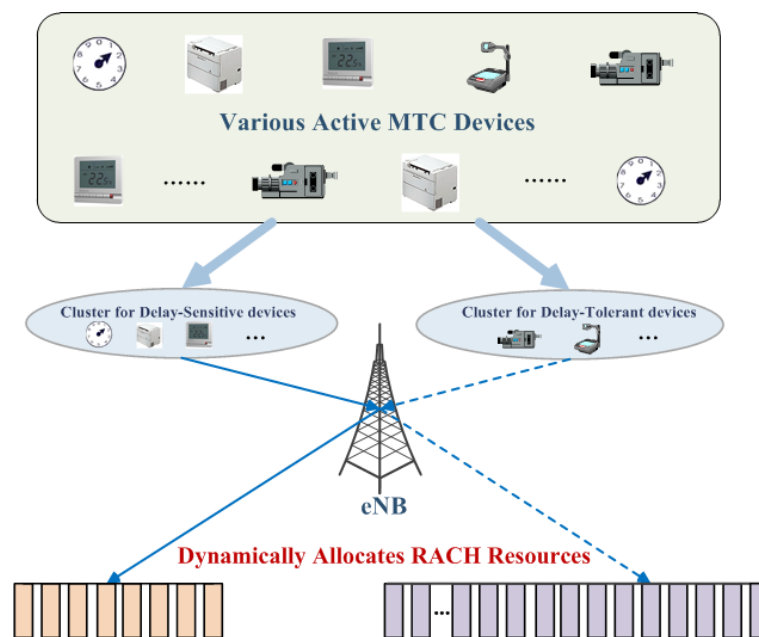


Figure 3. The diagram of the conceptual design.

#### 4.1. Clustered Structure

Due to the lower incidence of delay-sensitive services [37], the delay-sensitive devices utilize the preamble resources occasionally, which results in smaller traffic loads comparing to the delay-tolerant ones. Considering this, we divide those devices that are attempting to access the network into two clusters according to their requirements of delay. In the actual implementation of our scheme, we pre-categorize the attempting devices into two clusters. In other words, once the total number of devices is given, the devices would be automatically divided into two clusters.

#### 4.2. Dynamic Adjustment of Preamble Partition

After dividing the attempting devices into two clusters, the eNB dynamically determines the RA preamble partition between two clusters before accessing the network. We propose a feasible scheme to obtain the preamble partition. To formulate such a random access problem, we pre-define the variables as follows.

In a certain RA slot, we denote  $A_s$  as the number of delay-sensitive devices,  $A_n$  as the number of delay-tolerant ones and  $A_i$  as the total number of the active MTC devices. Namely,  $A_s + A_n = A_i$ . Then, we define the preamble partition as:

$$\beta = \frac{M_s}{M_n} \quad (1)$$

where  $M_s$  and  $M_n$  respectively represent the preambles allocated for delay-sensitive and delay-tolerant devices.

According to the configuration of RACH, we know:

$$M_s + M_n = c \quad (2)$$

Connecting Equation (1) with Equation (2) where  $c = 54$ , we solve:

$$M_s = \frac{c\beta}{1 + \beta} \quad (3)$$

$$M_n = \frac{c}{1 + \beta} \quad (4)$$

Moreover, we denote  $f_s$  and  $f_n$  as the access barring factors of two clusters, since we adopt the ACB scheme to access the network discussed in Section 4.3.

Since every MTC device selects the preamble randomly from the available resources pool, so collisions would occur if at least one MTC device transmits the same preamble. Moreover, according to [35], the contention-based RA procedure adopts a slotted-aloha as the access protocol, where the number of preambles available is analogous to the number of slots. In this way, the access success probability could be calculated by:

$$P_s = e^{-\frac{N}{P}} \quad (5)$$

where  $N$  is the total number of devices and  $P$  is the number of preambles available within an RA slot. On the basis of the illustration above, the number of devices successfully completing the access attempts  $S_N$  is defined as:

$$S_N = A_s f_s \times e^{-\frac{A_s f_s}{M_s}} + A_n f_n \times e^{-\frac{A_n f_n}{M_n}} \quad (6)$$

Then, substitute Equations (3) and (4) into Equation (6), and we can easily get:

$$S_N = A_s f_s \times e^{-\frac{A_s f_s (1+\beta)}{c\beta}} + A_n f_n \times e^{-\frac{A_n f_n (1+\beta)}{c}} \quad (7)$$

Obviously, our ultimate purpose is to maximize  $S_N$ . However, from the curve of  $S_N$  versus  $\beta$ , we can observe that when  $\beta$  is relatively smaller,  $S_N$  probably reaches its maximum, which implies that  $S_N$  could reach its maximum in the case of a suitable value of  $\beta$ . In other words, we can maximize the  $S_N$  by limiting the number of preambles allocated for delay-sensitive devices, which relates to the number of sensitive devices successfully accessing the network. Consequently, by intentionally setting the bounds of the sensitive devices successfully accessing the network, our scheme is formulated as the optimization problems depicted in Equation (8).

$$\left\{ \begin{array}{ll} \max_{\beta} & \left\{ A_s f_s \cdot e^{-\frac{A_s f_s}{c}(1+\frac{1}{\beta})} + A_n f_n \cdot e^{-\frac{A_n f_n(1+\beta)}{c}} \right\} & \text{if } A_s f_s \in (0, 1] \\ \max_{A_s f_s e^{-\frac{A_s f_s}{c}(1+\frac{1}{\beta})} \geq 1} & \left\{ A_s f_s \cdot e^{-\frac{A_s f_s}{c}(1+\frac{1}{\beta})} + A_n f_n \cdot e^{-\frac{A_n f_n(1+\beta)}{c}} \right\} & \text{if } A_s f_s \in (1, 3) \\ \max_{A_s f_s e^{-\frac{A_s f_s}{c}(1+\frac{1}{\beta})} \geq 2} & \left\{ A_s f_s \cdot e^{-\frac{A_s f_s}{c}(1+\frac{1}{\beta})} + A_n f_n \cdot e^{-\frac{A_n f_n(1+\beta)}{c}} \right\} & \text{if } A_s f_s \in [3, +\infty) \end{array} \right. \quad (8)$$

The objective function in Equation (8) maximizes the total number of devices successfully completing the access procedure under three restrictions of  $\beta$ . Then, we analyze the formulated optimization problems and correspondingly find their optimal solutions (denoted as  $\beta^*$ ), respectively.

**Case 1:**  $A_s f_s \in (0, 1]$

When  $A_s f_s \in (0, 1]$ , our formulated problem becomes an unconstrained optimization problem. First, we get the first-order derivative of the objective function depicted in Equation (9):

$$\frac{A_s^2 f_s^2}{c \beta^2} \times e^{-\frac{A_s f_s(1+\beta)}{c \beta}} - \frac{A_n^2 f_n^2}{c} \times e^{-\frac{A_n f_n(1+\beta)}{c}} \equiv 0 \quad (9)$$

and let Equation (9) be equal to zero. By simplifying Equation (9), we have:

$$\frac{A_s f_s}{c} \left( 1 + \frac{1}{\beta} \right) - \frac{A_n f_n}{c} (1 + \beta) + 2 \log \beta = \log \left( \frac{A_s^2 f_s^2}{A_n^2 f_n^2} \right) \quad (10)$$

Since there is no closed solution for Equation (10) through using Mathematica, we decide to find the approximate solution by means of numerical analysis. We adopt the **Newton iteration algorithm** depicted in Algorithm 2 to find the approximate solution  $\beta^*$ .

---

**Algorithm 2 :** Newton iterative algorithm.

---

- 1: **Initialization:** endow  $A_s, A_n, f_s, f_n$  with any feasible values and then set the initial value of iteration  $\beta_0$ ;
  - 2: **Iteration:** taking  $\beta_0$  as the iteration initial value and  $10^{-5}$  as the given precision, the Newton iterative algorithm is used to search for the first approximate solution  $\beta_1$ ;
  - 3: **Recursion:** take  $\beta_1$  as the iteration initial value and  $10^{-5}$  as the given precision to get the second approximate solution  $\beta_2$ , and repeat the steps above to get the solution set  $\{\beta_1, \beta_2, \dots, \beta_m\}$ ;
  - 4: **Confirmation:** substitute  $\{\beta_1, \beta_2, \dots, \beta_m\}$  into the objective function in turn and select the  $\beta^*$  for maximizing the objective function;
  - 5: **End**
- 

**Case 2:**  $A_s f_s \in (1, 3)$

Different from **Case 1**, the problem under this case becomes an optimization problem with an inequality constraint. First, we determine the feasible domain described as:



$$D_1 = \left\{ \beta \mid \beta \geq \frac{A_s f_s}{c * \log(A_s f_s) - A_s f_s} \right\} \quad (11)$$

and then determine the monotonicity of the objective function in the feasible domain. Since the objective function is monotonically decreasing within the feasible domain, the maximum value of the objective function is gained at the boundary of the feasible region, namely:

$$\beta^* = \frac{A_s f_s}{c * \log(A_s f_s) - A_s f_s} \quad (12)$$

**Case 3:**  $A_s f_s \in [3, +\infty)$

Similar to **Case 2**, the feasible domain under this case is illustrated as:

$$D_2 = \left\{ \beta \mid \beta \geq \frac{A_s f_s}{c * \log(A_s f_s / 2) - A_s f_s} \right\} \quad (13)$$

Similar to the analysis in **Case 2**, the optimal  $\beta^*$  is expressed as:

$$\beta^* = \frac{A_s f_s}{c * \log(A_s f_s / 2) - A_s f_s} \quad (14)$$

In summary, according to the conceptual design discussed above in a certain slot, once we have known the  $A_s, A_n, f_s, f_n$ , the eNB would obtain the optimal preamble partition based on the proposed scheme.

#### 4.3. Access Class Barring for Two Clusters Respectively

After obtaining the preamble partition, the devices belonging to different clusters adopt the ACB scheme respectively to access the network. According to the ACB mechanism, we propose that the ac\_barrington of two clusters is configured as the same. In order to simplify our proposed architecture, the connection between two ac\_barrington factors would be devised as the following linear correlation:  $f_n = p + q * f_s$  where  $f_s \in (0, 1)$  and the value of  $p$  and  $q$  should satisfy  $f_n \in (0, 1)$ . In Section 6, we will show the parameter settings and illustrate the proofs in detail.

### 5. Analysis of the Resource Partition Scheme

In this part, we mainly address the theoretical performance analysis of the proposed scheme. Before that, the estimation of the access loads should be taken into consideration, since we assume that the traffic loads during one slot are pre-known in the proposed scheme. As previously mentioned, once obtaining  $\beta^*$  in a slot, the eNB would allocate the corresponding number of preambles to the two clusters based on the proposed preamble partition scheme. Based on the number of allocated preambles and the collision status in each slot, we can apply a Markov-based approach, which was developed in [38], to well estimate the traffic load of each cluster dynamically.

Next, before analyzing the performance of the proposed scheme, we consider the following performance indexes:

(a) *Access success probability (ASP)*, defined as the ratio between total devices completing the RA procedures and the total access attempts within the same slot.

(b) *Preamble collision probability (PCP)*, defined as the ratio between the number of preamble collision and the total preambles transmitted within the same slot.

(c) *Average access delay (AAD)*, defined as the average delay between the first attempt and the completion of all RA procedures for the devices that successfully access the network.

### 5.1. Analysis of Access Success Probability

Under the assumptions and notations defined in Section 4, we will subsequently analyze the performance from three aspects. Additionally, so as to compare with the simulation results in Section 6, we suppose that the ratio of the number for two clusters is denoted as  $\gamma$ . Then, we can get  $A_n/A_s = \gamma$ , as well as  $A_s + A_n = A_i$ . Accordingly,  $A_s$  and  $A_n$  are expressed as follows:

$$A_n = \frac{\gamma A_i}{1 + \gamma} \quad (15)$$

$$A_s = \frac{A_i}{1 + \gamma} \quad (16)$$

Next, once we have solved the  $\beta^*$  in the proposed scheme, we would obtain the number of devices successfully completing the access attempts  $S_N$  in a certain slot by substituting  $\beta^*$  into Equation (7), described as:

$$S_N = A_s f_s \times e^{-\frac{A_s f_s (1 + \beta^*)}{c \beta^*}} + A_n f_n \times e^{-\frac{A_n f_n (1 + \beta^*)}{c}} \quad (17)$$

Thus, the ASP in a certain slot is calculated by:

$$P_S = \frac{A_s f_s \times e^{-\frac{A_s f_s (1 + \beta^*)}{c \beta^*}} + A_n f_n \times e^{-\frac{A_n f_n (1 + \beta^*)}{c}}}{A_i} \quad (18)$$

Then, substituting Equations (15) and (16) into Equation (18), the expression Equation (18) can be simplified as:

$$P_S = \frac{f_s}{1 + \gamma} \times e^{-\frac{f_s A_i}{(1 + \gamma)c} (1 + \frac{1}{\beta^*})} + \frac{\gamma f_n}{1 + \gamma} \times e^{-\frac{\gamma A_i f_n}{(1 + \gamma)c} (1 + \beta^*)} \quad (19)$$

In accordance with the simulation setup in Section 6, here we generally present the parameter settings in advance. First we devise the connection between two ac\_barrangingfactors as:  $f_n = 0.6 + \frac{1}{3} * f_s$ , which are explained in Section 6.1. Moreover, the ratio of the number for two clusters is defined as  $\gamma = 9$ . The traffic model is devised as a  $\beta$ -distribution within 10 s. In this way, from a realization example of the number of MTC arrivals within 10 s, we can see that the number of access attempts falls within the range from one to 50. It is worth noting that in our analysis, we set the lower bound for  $A_i$  as one instead of zero, which is the actual minimum just to avoid the denominator in Equation (18) being zero, while not affecting the analysis results. On the other hand, we set the maximum value for  $A_i$  as 50.

Since the number of access attempts during every slot is rand, here we typically choose two extremes to present the analysis on ASP. One is the maximum number of access attempts; the other is the minimum one. Here, we depict the relationships between  $f_s$  and  $P_S$  in two extreme cases respectively in Figure 4. It can be seen that the ASPs of two extreme scenarios are both rising with the increase of ac\_barrangingfactor. Meanwhile, the larger the access load is, the worse the access condition becomes.

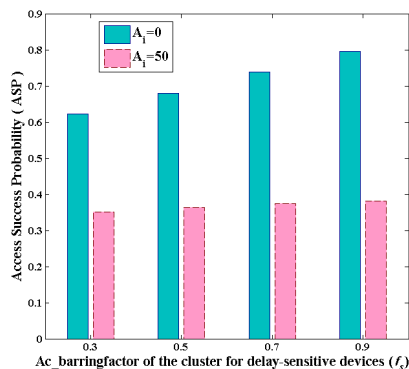


Figure 4. The access success probability (ASP) of two extreme cases versus  $f_s$  during one slot.

### 5.2. Analysis of Preamble Collision Probability

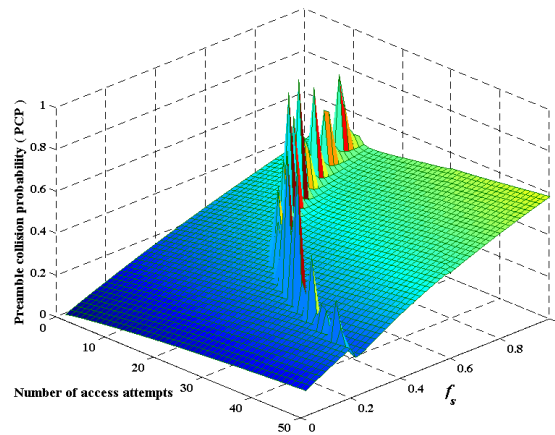
Similarly, on the basis of the derivation in ASP performance, we can analyze the preamble collision probability along this way. Inspired by Equations (5) and (7), we present the PCP during one slot defined as  $P_C$ :

$$P_C = \frac{A_s f_s \cdot (1 - e^{-\frac{A_s f_s}{c} (1 + \frac{1}{\beta^*})}) + A_n f_n \cdot (1 - e^{-\frac{A_n f_n}{c} (1 + \beta^*)})}{A_s f_s + A_n f_n} \quad (20)$$

After similar arrangements and simplifications, Equation (20) is described as Equation (21):

$$P_C = 1 - \frac{\frac{\gamma f_s}{1+\gamma} \cdot e^{-\frac{\gamma f_s A_i}{(1+\gamma)c} (1 + \frac{1}{\beta^*})} + \frac{f_n}{1+\gamma} \cdot e^{-\frac{A_i f_n}{(1+\gamma)c} (1 + \beta^*)}}{\frac{\gamma f_s}{1+\gamma} + \frac{f_n}{1+\gamma}} \quad (21)$$

We depict the three-dimensional curve, which presents the variation tendency of  $P_C$  when  $A_i$  and  $f_s$  synchronously change in Figure 5. During a certain slot, we can observe that the PCP is increasing as  $f_s$  raises. Furthermore, the PCP also climbs with the rising of the total attempts during one slot. Here, it is worth noting that there are several *singular points* changing suddenly. According to the conceptual design discussed in Section 4,  $A_s f_s = 1$  is taken as the “critical condition” distinguishing between two different ways of analysis, as well as solving. In Figure 5, it is verified that *singular points* approximately appeared near  $A_s f_s = \frac{A_s f_s}{1+\gamma} = 1$ .



**Figure 5.** Collision probability varies with the synchronously change of  $A_i$  and  $f_s$  during one slot.

### 5.3. Analysis of Average Access Delay

The access delay includes two main parts comprised of the random backoff  $T_{barred}$  in Equations (24) and (25) due to  $ac\_barringfactor$ , as well as the fixed backoff due to collisions. Since there is a random number (defined as  $rand$ ) ranging from zero to one in  $T_{barred}$ , here we consider the potential upper bound and lower bound during one slot.

First, in the case of the lower limit, we assume  $rand = 0$  in Equations (24) and (25), and the AAD (seconds) is described as:

$$D_{lower} = A_s (1 - f_s) \times (0.7 + 0.6 * 0) * 4 + A_n (1 - f_n) \times (0.7 + \frac{0.6 * 0}{f_n}) * 4 + P_C (A_s f_s + A_n f_n) \times 5 \times 10^{-3} \quad (22)$$

where  $5 \times 10^{-3}$ , denoting five milliseconds, represents the fixed backoff due to collisions. That is to say, once collision occurs, the access backs off for 5 ms.

Then, in the case of the upper bound, we assume  $rand = 1$ , and similarly, the AAD (seconds) is expressed as:

$$D_{upper} = A_s(1 - f_s) \times (0.7 + 0.6 * 1) * 4 + A_n(1 - f_n) \times (0.7 + \frac{0.6 * 1}{f_n}) * 4 + P_C (A_s f_s + A_n f_n) \times 5 \times 10^{-3} \quad (23)$$

## 6. Simulation Evaluation

In this section, we present the setup of the simulation parameters and scenarios, as well as the simulation results that demonstrate the advantageous performance of the proposed scheme for solving the RAN overload problems.

### 6.1. Simulation Setup

According to 3GPP protocols [39], the detailed LTE-A simulation parameters [40], as well as the typical traffic model [34] are depicted in Table 1. In order to evaluate our proposed scheme, we need to illustrate the basic parameters in the common ACB mechanism available in the proposed scheme.

(a) As discussed in Section 4, we first define the proportion between sensitive devices and non-sensitive devices as 1:9 due to the low incidences of sensitive devices according to our comprehensive research.

(b) As for the two parameters broadcast in the ACB mechanism, we shall assume that the  $ac\_barringtime$  of two clusters is equal to 4 s. According to [41], in case of being barred, a sensitive device re-attempts access after a delay of:

$$T_{barred} = (0.7 + 0.6 * rand) * ac\_barringtime \quad (24)$$

where  $rand$  represents a random number uniformly drawn from  $[0, 1)$ . Yet, the non-sensitive device re-attempts access after a delay of:

$$T_{barred} = (0.7 + \frac{0.6 * rand}{ac\_barringfactor}) * ac\_barringtime \quad (25)$$

(c) According to the ACB mechanism,  $ac\_barringfactor$  ranges from zero to one. Hence, we shall select suitable values for  $p$  and  $q$  in  $f_n = p + q * f_s$  in order to satisfy  $f_n \in (0, 1)$  when  $f_s$  is given in advance. Accordingly, through simulations in Figures 6 and 7, on the basis of the considerations of ASP and AAD, we choose the linear correlation of  $f_s$  and  $f_n$  as:  $f_n = 0.6 + \frac{1}{3} * f_s$ , which can achieve the optimal overall performance. Notably, there is a sharp decline in Figure 6 when  $f_s$  changes from 0.8 to 0.9 in the case of  $f_n = 0.6 + \frac{1}{3} * f_s$ . This is because when  $f_s$  changes from 0.8 to 0.9,  $f_n$  changes from 0.87 to 0.9, which means nearly 90% of delay-tolerant devices would compete for access. In addition, due to the huge number of delay-tolerant devices, devices competing for access do not successfully access the network more easily compared to the situations for fewer devices. The more competing devices there are, the more frequent the collisions are, which results in lower ASP. Thus, there is a sharp decline as  $f_s$  varies from 0.8 to 0.9. Therefore, setting  $f_s$  as 0.9 is practically meaningless in real scenarios. Consequently, our simulation is conducted in  $f_n = 0.6 + \frac{1}{3} * f_s$ .

(d) The baseline scheme we use for performance comparison is the original ACB scheme, since it is the currently-suggested solution for RAN overload in M2M networks by 3GPP. Notably, the baseline ACB scheme neither divides the MTC devices into cluster groups nor partitions the preambles.

Table 1. Basic simulation parameters.

Parameter	Settings
Cell bandwidth	5 MHz
Number of M2M devices	30,000
Attempts' distribution	Beta distribution
Distribution period	10 s
PRACH configuration index	6
Number of preambles for contention-based RA	54
Ra-ResponseWindowSize	5 sub-frames
Backoff indicator	5 ms

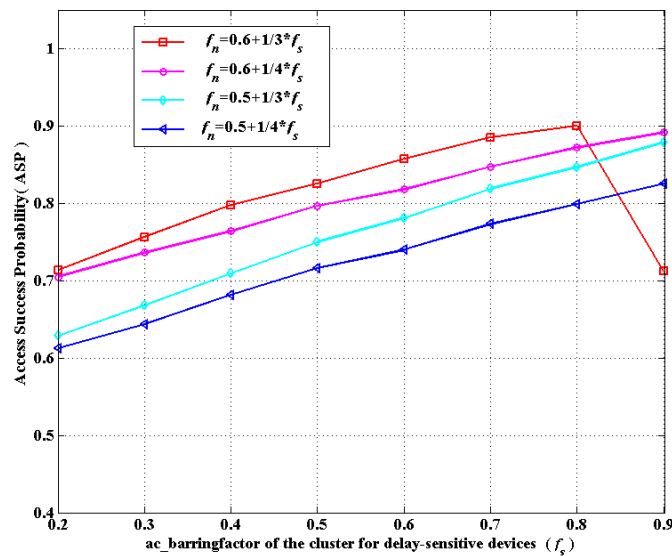


Figure 6. The ASP versus the linear correlation of  $f_s$  and  $f_n$ .

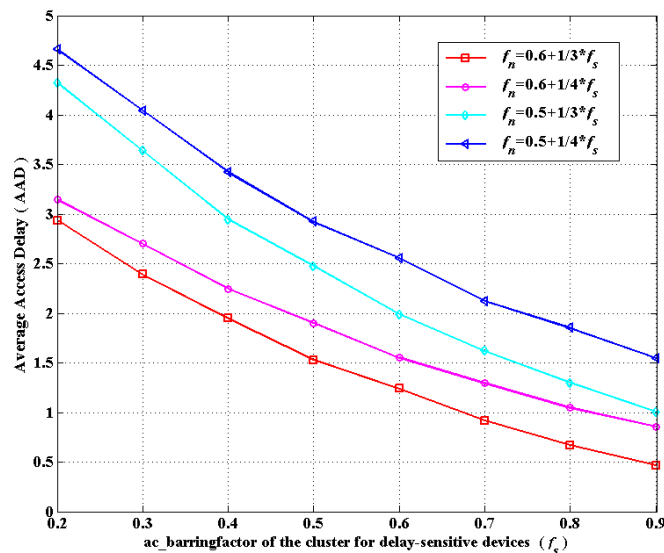
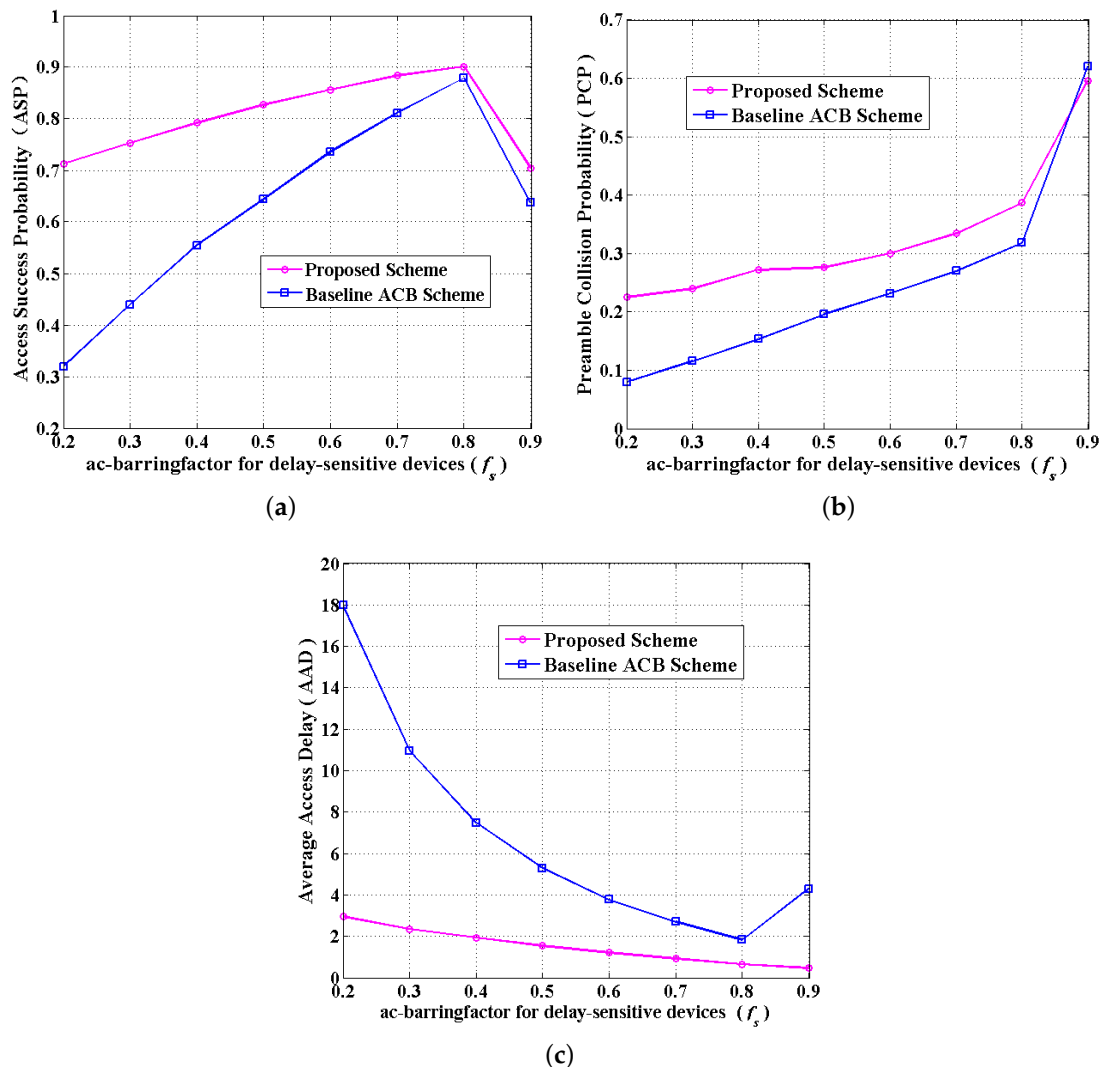


Figure 7. The average access delay (AAD) versus the linear correlation of  $f_s$  and  $f_n$ .

6.2. Simulation Results

In Figure 8, we demonstrate the simulation results in terms of the overall ASP, PCP and AAD under the proposed scheme compared to the baseline ACB scheme. Figure 8a,c shows the performance

of ASP and AAD *versus*  $f_s$  under the proposed scheme and the baseline ACB scheme. Compared to the baseline ACB scheme, the proposed scheme yields much better performance in improving the overall access success probability, as well as reducing the average access delay. From the general trends in Figure 8a, two curves almost simultaneously climb up when  $f_s$  varies from 0.2 to 0.8 and drop when  $f_s$  varies from 0.8 to 0.9. Furthermore, we can observe from Figure 8a that there is a large gap between two curves when  $f_s$  is smaller than 0.7, and the gap narrows when  $f_s = 0.8$ . This is because when  $f_s$  changes from 0.8 to 0.9,  $f_n$  changes from 0.87 to 0.9, which means nearly 90% of delay-tolerant devices would compete for access. In addition, due to the huge number of delay-tolerant devices, devices competing for access do not successfully access the network more easily compared to the situations for fewer devices. The more competing devices there are, the more frequent the collisions are, which results in lower ASP in Figure 8a and longer access delay in Figure 8c. Similarly, two curves in Figure 8c drop when  $f_s$  changes from 0.2 to 0.8, and the gap between them narrows continuously, as well a slight increase occurs in two curves when  $f_s$  varies from 0.8 to 0.9. Obviously, the optimal value for  $f_s$  can be adopted as 0.8 for the proposed scheme to obtain high ASP and lower AAD.



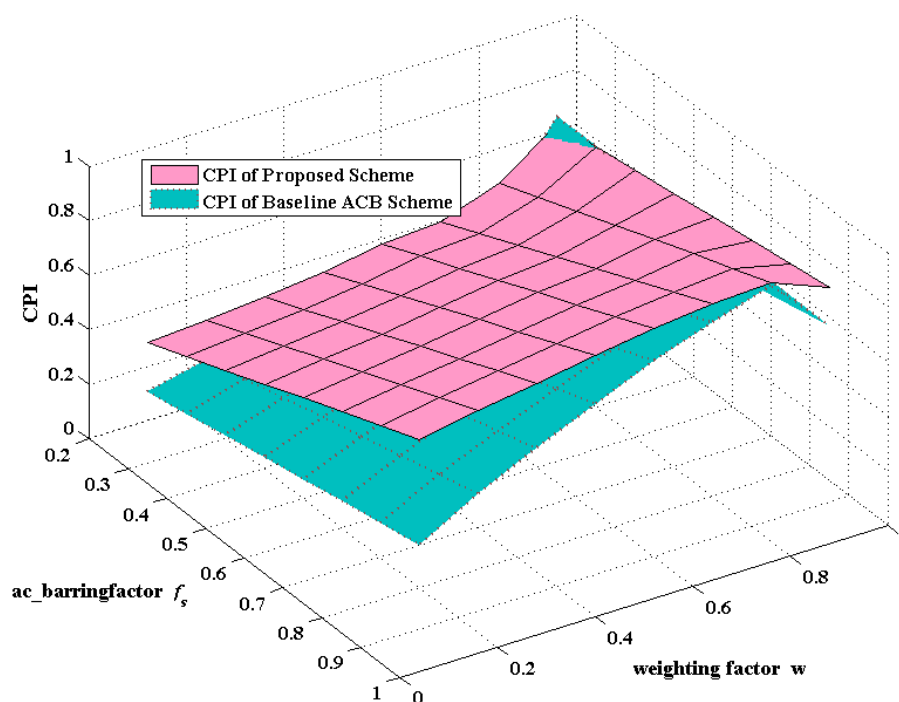
**Figure 8.** Performance comparison between the proposed scheme and the baseline access class barring (ACB) scheme. (a) The ASP *versus* the ac\_barringfactor of the cluster for delay-sensitive devices; (b) the preamble collision probability (PCP) *versus* the ac\_barringfactor of the cluster for delay-sensitive devices; (c) the AAD *versus* the ac\_barringfactor of the cluster for delay-sensitive devices.

Instead, the preamble collision probability of the proposed scheme performs higher PCP for about 10%, while the growth falls when  $f_s$  rises in Figure 8b, which is accounted for by the rise of the number of non-barred MTC devices when  $f_s$  is larger. Consequently, in order to visually present the trade-off between access success probability and collision probability, here we introduce a weighting factor denoted as  $\omega$  and a new indicator defined as the comprehensive parameter index (CPI). Namely, the CPI can be expressed as:

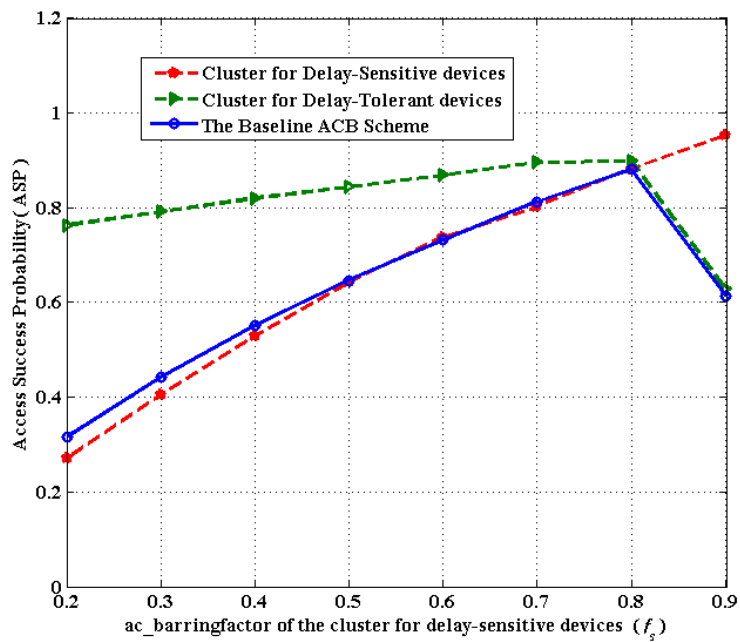
$$CPI = \omega \cdot P_S + (1 - \omega) \cdot P_C \quad (26)$$

Figure 9 depicts the three-dimensional curve, which demonstrates the variation tendency of CPI with the synchronous change of  $f_s$  and  $\omega$ . Furthermore, Figure 9 reveals that the CPI of the proposed scheme is higher than that of the ACB scheme with the variation of weighting factor  $\omega$  when fixing the given  $f_s$ .

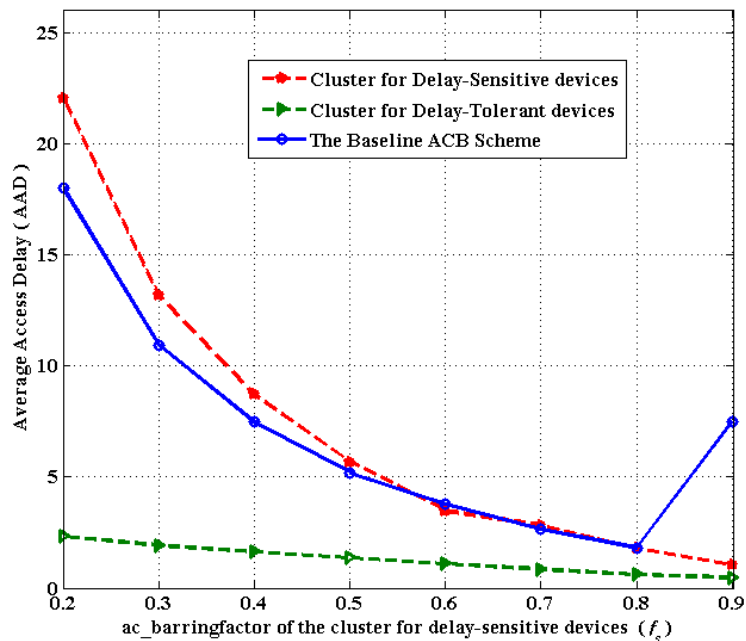
In comparison, Figure 10 also presents the performance of two clusters for delay-sensitive and delay-tolerant devices separately, which shows that our proposed scheme achieves higher performance without sacrificing any kinds of services. From Figure 10, we can observe that when  $f_s$  is larger than 0.6, the proposed scheme provides a slight effect on delay-sensitive services, while greatly improving the performance of delay-tolerant services in terms of ASP, which significantly increases the overall performance. In Figure 10a, the cluster for delay-sensitive devices yields a slightly lower ASP than when  $f_s$  is smaller than 0.5, which makes no difference, since  $f_s$  is set larger than 0.5 in a real system. Similarly, Figure 10b demonstrates that when  $f_s$  is larger than 0.5, the proposed scheme provides a slight effect on delay-sensitive services, while effectively reducing the AAD for delay-tolerant services, which significantly increases the overall performance. Notably, Figure 10a,b comprehensively demonstrates that the cluster for delay-tolerant devices yields more effective performance than the baseline ACB scheme.



**Figure 9.** The comprehensive parameter index (CPI) varies with the synchronous change of  $\omega$  and  $f_s$  between the proposed scheme and the baseline ACB scheme.



(a)



(b)

**Figure 10.** Performance comparison between two clusters for delay-sensitive and delay-tolerant devices and the baseline ACB scheme. (a) The ASP *versus*  $f_s$  between two clusters and the baseline ACB scheme; (b) the AAD *versus*  $f_s$  between two clusters and the baseline ACB scheme.

## 7. Conclusions

In this paper, we firstly presented the RAN overload issue caused by massive devices attempting to access the eNB in the M2M communication architecture. As for the scenario where delay-sensitive and delay-tolerant devices coexist, we then proposed a mechanism for access control to jointly guarantee the RAN overload requesting from differentiated services. Based on the loading condition for attempts to access from two kinds of devices, our proposed scheme dynamically allocates the preambles to



respectively accommodate both sides. In our work, the core is to devise an optimal preamble partition for both sides in order to realize the maximum access success probability during each slot. We then theoretically analyzed the performance from three aspects and provided different forms of the analysis results. In addition, computer simulations are conducted to demonstrate that our proposed scheme has performance superiority over the baseline ACB scheme in terms of overall access success probability and access success delay.

**Acknowledgments:** The research work reported in this paper is supported by the National Natural Science Foundation of China under the Grant No. 61461136001, the Specialized Research Fund for the Doctoral Program of Higher Education under Grant No. 20110201120014, and the Fundamental Research Funds for the Central Universities.

**Author Contributions:** Q.D. conceived of the study, proposed the idea, coordinated the research and corrected the manuscript. W.L. designed the specific scheme and drafted the manuscript. L.L. participated in the manuscript's design and provided valuable suggestions. P.R. participated in the coordination of the research and searched for funding to support the work. Y.W. and L.S. proofread the manuscript. All authors read and approved the final manuscript.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References and Notes

- Ghavimi, F.; Chen, H.H. M2M communications in 3GPP LTE/LTE-A networks: Architectures, service requirements, challenges and applications. *IEEE Commun. Surveys Tuts.* **2015**, *17*, 525–549.
- Jung, S.-J.; Chung, W.-Y. Non-Intrusive healthcare system in global Machine-to-Machine networks. *IEEE Sens. J.* **2013**, *13*, 4824–4830.
- Ide, C.; Dusza, B.; Putzke, M.; Muller, C.; Wietfeld, C. Influence of M2M communication on the physical resource utilization of LTE. In Proceedings of the Wireless and Mobile Networking Conference (WMNC), Dubai, UAE, 23–25 April 2013; pp. 1–4.
- Lam, R.K.; Chen, K.C. Congestion control for M2M traffic with heterogeneous throughput demands. In Proceedings of the Wireless Communications and Networking Conference (WCNC), Shanghai, China, 8 January 2013; pp. 1452–1457.
- Chen, Y.-S.; Chiang, W.-L. A Spiderweb-Based massive access management protocol for M2M wireless networks. *IEEE Sens. J.* **2015**, *15*, 5765–5776.
- 3GPP TR 37.868 V11.0.0. Study on RAN Improvements for Machine-Type Communications. (Release 11) September 2011.
- Wu, G.; Talwar, S.; Johnsson, K.; Himayat, N.; Johnson, K.D. M2M: From mobile to embedded Internet. *IEEE Commun. Mag.* **2011**, *49*, 36–43.
- Phuyal, U.; Vannithamby, R. Controlling access overload and signaling congestion in M2M networks. In Proceedings of the Asilomar Conference on Signals, Systems and Computers (ASILMOR), Pacific Grove, CA, USA, 6–9 November 2012; pp. 591–595.
- Wang, S.-H.; Su, H.-J.; Hsieh, H.-Y.; Yeh, S.-P.; Ho, M. Random access design for clustered wireless machine to machine networks. In Proceedings of the IEEE BlackSeaCom, Batumi, Georgia, 3–5 July 2013; pp. 107–111.
- Lo, A.; Law, Y.; Jacobsson, M. A cellular-centric service architecture for machine-to-machine (M2M) communications. *IEEE Wirel. Commun.* **2013**, *20*, 143–151.
- Lien, S.-Y.; Chen, K.-C.; Lin, Y. Toward ubiquitous massive accesses in 3GPP machine-to-machine communications. *IEEE Commun. Mag.* **2011**, *49*, 66–74.
- Wang, Y.; Ren, P.; Du, Q.; Sun, L. Optimal power allocation for underlay-based cognitive radio networks with primary user's statistical delay QoS provisioning. *IEEE Trans. Wirel. Commun.* **2015**, *14*, 6896–6910.
- Wang, Y.; Ren, P.; Gao, F.; Su, Z. A hybrid underlay/overlay transmission mode for cognitive radio networks with statistical Quality-of-Service provisioning. *IEEE Trans. Wirel. Commun.* **2014**, *13*, 1482–1498.
- Chen, H.; Liu, L.; Matyjas, J.; Medley, M. Optimal resource allocation for sensing based spectrum sharing D2D networks. *Comput. Elec. Eng. (Elsevier)* **2015**, *44*, 107–121.
- Chen, H.; Liu, L. Resource allocation for sensing-based device-to-device (D2D) networks. In Proceedings of the 49th Asilomar Conference on Signals, Systems and Computers, Pacific Grove, CA, USA, 8–11 November 2015; pp. 1058–1062.

16. Du, Q.; Song, H.; Xu, Q.; Ren, P.; Sun, L. Interference-controlled D2D routing aided by knowledge extraction at cellular infrastructure towards ubiquitous CPS. *Pers. Ubiquitous Comput.* **2015**, *19*, 1033–1043.
17. Sun, L.; Du, Q.; Ren, P.; Wang, Y. Two birds with one stone: Towards secure and interference-free D2D transmissions via constellation rotation. *IEEE Trans. Veh. Technol.* **2015**, doi:10.1109/TVT.2015.2505715.
18. Sun, L.; Ren, P.; Du, Q.; Wang, Y.; Gao, Z. Security-aware relaying scheme for cooperative networks with untrusted relay nodes. *IEEE Commun. Lett.* **2015**, *19*, 463–466.
19. Pereira, C.; Aguiar, A. Towards Efficient Mobile M2M Communications: Survey and Open Challenges. *Sensors* **2014**, *14*, 19582–19608.
20. Zheng, K.; Ou, S.; Alonso-Zarate, J.; Dohler, M.; Liu, F.; Zhu, H. Challenges of massive access in highly dense LTE-advanced networks with machine-to-machine communications. *IEEE Wirel. Commun.* **2014**, *21*, 12–18.
21. Lee, J.; Jung, I. Adaptive-Compression Based Congestion Control Technique for Wireless Sensor Networks. *Sensors* **2010**, *10*, 2919–2945.
22. Bayat, S.; Li, Y.; Han, Z.; Dohler, M.; Vucetic, B. Distributed massive wireless access for cellular machine-to-machine communication. In Proceedings of the International Conference on Communications (ICC), Sydney, Australia, 10–14 June 2014; pp. 2767–2772.
23. Chen, Y.-S.; Chiang, W.-L. A Spiderweb-Based Massive Access Management Protocol for M2M Wireless Networks. *IEEE Sens. J.* **2015**, *15*, 5765–5776.
24. 3GPP TS 36.331 V10.5.0, Evolved Universal Terrestrial Radio Access (E-UTRA). Radio Resource Control (RRC). March 2012.
25. Chen, Y.; Wang, W. Machine-to-Machine Communication in LTE-A. In Proceedings of the IEEE Vehicular Technology Conference (VTC Fall), Ottawa, ON, Canada, 6–9 September 2010; pp. 1–4.
26. Amokrane, A.; Ksentini, A.; Hadjadj-Aoul, Y.; Taleb, T. Congestion control for machine type communications. In Proceedings of the IEEE International Conference on Communications (ICC), 10–15 June 2012; pp. 778–782.
27. Lo, A.; Law, Y.; Jacobsson, M.; Kucharzak, M. Enhanced LTE-Advanced Random-Access Mechanism for Massive Machine-to-Machine (M2M) Communications. In Proceedings of the 27th Meeting of Wireless World Research Forum (WWRF), Düsseldorf, Germany, 23–25 October 2011.
28. Taleb, T.; Kunz, A. Machine type communications in 3GPP networks: Potential, challenges, and solutions. *IEEE Commun. Mag.* **2012**, *50*, 178–184.
29. Lu, R.; Li, X.; Liang, X.; Shen, X.; Lin, X. GRS: The green, reliability, and security of emerging machine to machine communications. *IEEE Commun. Mag.* **2011**, *49*, 28–35.
30. Zhang, Y.; Yu, R.; Xie, S.; Yao, W.; Xiao, Y.; Guizani, M. Home M2M networks: Architectures, standards, and QoS improvement. *IEEE Commun. Mag.* **2011**, *49*, 44–52.
31. Cheng, J.-J.; Lee, C.-H.; Lin, T.-M. Prioritized random access with dynamic access barring for RAN overload in 3GPP LTE-A networks. In Proceedings of the IEEE Global Communications Conference (GLOBECOM) Workshops, Houston, TX, USA, 5–9 December 2011; pp. 368–372.
32. Yang, X.; Fapojuwo, A.; Egbogah, E. Performance Analysis and Parameter Optimization of Random Access Backoff Algorithm in LTE. In Proceedings of the IEEE Vehicular Technology Conf. (VTC Fall), Quebec, QC, Canada, 3–6 September 2012; pp. 1–5.
33. Pratas, N.K.; Thomsen, H.; Stefanovic, C.; Popovski, P. Code-Expanded Random Access for Machine-Type Communications. In Proceedings of the IEEE Global Communications Conference (GLOBECOM) Workshops, Anaheim, CA, USA, 3–7 December 2012; pp. 1681–1686.
34. Cheng, M.Y.; Lin, G.-Y.; Wei, H.-Y.; Hsu, A.-C. Overload control for Machine-type-communications in LTE-Advanced system. *IEEE Commun. Mag.* **2012**, *50*, 38–45.
35. Laya, A.; Alonso, L.; Alonso-Zarate, J. Is the random access channel of LTE and LTE-A suitable for M2M communications? a survey of alternatives. *IEEE Commun. Surveys Tuts.* **2014**, *16*, 4–16.
36. Afrin, N.; Brown, J.; Khan, J.Y. Performance analysis of an enhanced delay sensitive LTE uplink scheduler for M2M traffic. In Proceedings of the Telecommunication Networks and Applications Conference (ATNAC), Christchurch, New Zealand, 20–22 November 2013; pp. 154–159.
37. Park, I.; Kim, D.; Har, D. MAC achieving low latency and energy efficiency in hierarchical M2M networks with clustered nodes. *IEEE Sens. J.* **2015**, *15*, 1657–1661.
38. He, H.; Du, Q.; Song, H.; Li, W.; Wang, Y.; Ren, P. Traffic-aware ACB scheme for massive access in machine-to-machine networks. In Proceedings of the IEEE International Conference on Communications, London, UK, 8–12 June 2015.

39. 3GPP TSG RAN WG2 #71 R2-104663, [70bis#11] LTE: MTC LTE Simulations. ZTE, Madrid, Spain, 23 August 2010.
40. 3GPP TSG RAN WG2 #71 R2-104662, MTC Simulation Results With Specific Solutions. ZTE, Madrid, Spain, 23 August 2010.
41. 3GPP TR 23.898, Access Class Barring and Overload Protection. (Release 7). V7.0.0, March 2005.



© 2016 by the authors; licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons by Attribution (CC-BY) license (<http://creativecommons.org/licenses/by/4.0/>).