

Research Article

Efficient Data Forwarding for Machine Type Communications in Internet of Things Networks

Yujin Lim¹ and Jaesung Park²

¹Department of IT Engineering, Sookmyung Women's University, 100 Cheongpa-ro 47-gil, Yongsan-gu, Seoul 04310, Republic of Korea

²Department of Information Security, University of Suwon, San 2-2, Wau-ri, Bongdam-eup, Hwaseong, Gyeonggi-do 445-743, Republic of Korea

Correspondence should be addressed to Jaesung Park; jaesungpark@suwon.ac.kr

Received 19 September 2017; Accepted 4 December 2017; Published 24 December 2017

Academic Editor: Dongkyun Kim

Copyright © 2017 Yujin Lim and Jaesung Park. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The MTC (Machine Type Communications) system is one of the most promising technologies to provide IoT (Internet of Things) applications. The MTC system suffers from congestion due to limited data transmission capacity and burst traffic. The congestion disturbs data delivery, results in an increase of energy consumption due to data retransmission, and finally poses a threat to the providers of IoT applications. In this paper, we focus on the congestion problem and present an efficient data forwarding mechanism to regulate the burst channel access from a large number of MTC devices. To regulate the communication channel access from the devices, we predict the number of devices by separating traffic load into the prediction of the number of devices that newly attempt to have access and the prediction of the number of devices that retry to access the channel. Through simulations, we show that our mechanism improves access success probability and reduces collision probability and access delay.

1. Introduction

IoT (Internet of Things) is a system that enables data transfer among interrelated computing devices over a network without requiring human-to-computer interactions. Popular IoT applications include smart homes, wearables devices, smart grids, and smart cities. In IoT applications, a large number of devices generate data to communicate with each other. There exist many communication technologies to provide connectivity between the devices. Among these technologies, cellular networks such as LTE (Long Term Evolution) can be used for data delivery over long distances. The 3GPP (3rd-Generation Partnership Project) has specified MTC (Machine Type Communications) in LTE networks to support data delivery [1, 2].

In many IoT applications, traffic is highly synchronized owing to event-driven characteristics. In other words, a large number of devices try to access a network within a short period of time, which may cause congestion problems. To solve the problem, an access control mechanism to regulate the access must protect the system's performance and service quality [3].

In MTC, a device which has data to send communicates with eNB (evolved Node-B) to reserve a channel resource in the following handshaking procedure [4, 5]. First, a device which has data to send selects one of several predefined preamble signatures. The device sends a request message with the selected preamble to eNB in order to reserve an uplink resource. Second, eNB assigns the uplink resource to the requesting device. If multiple devices select the same preamble and send request messages to eNB, the devices are assigned the same uplink resource. Third, the device sends its identity information to eNB by using the assigned uplink resource. In the case of multiple devices with the same preamble, if they send their identity information by using the same uplink resource simultaneously, a collision occurs. Finally, eNB acknowledges the device whose message is successfully decoded. In the case of a collision, eNB fails to decode the message from the collided device and it does not acknowledge the device. The unacknowledged device fails in its random access attempt and retries the random access procedure after a preset backoff time. When the number of retrials reaches the maximum number of preamble

transmissions [1], the device finally fails in its channel access attempt.

As the number of devices which have data to send increases, the competition for the RACH (Random Access Channel), which represents a sequence of time-frequency resources (RA slots), increases and collisions increase in number. To alleviate the collision problem, a mechanism must efficiently control the random access process.

In this paper, we propose a means to regulate random access from MTC devices. First, we estimate the number of backlogged devices which contend for the channel resources in an RA slot. To estimate the number of backlogged devices, we separate the backlogged devices into the devices which are carried over from the previous RA slot and the devices which are newly activated in the current RA slot. By separating, we estimate the number of backlogged devices in the current RA slot. Then, based on the number of backlogged devices in the current RA slot, we predict the number of backlogged devices in the next RA slot. With the number of backlogged devices in the next RA slot, we control the random access attempts of devices to reduce the resource competition. The simulation results show that our mechanism improves the number of devices which succeed in their random access attempts.

The rest of this paper is organized as follows: In Section 2, we present related research. In Section 3, we propose our control mechanism to regulate random access attempts. In Sections 4 and 5, we evaluate the performance of the proposed mechanism and conclude the paper.

2. Related Work

3GPP has specified the ACB (Access Class Barring) mechanism to solve the congestion problem in IoT applications [6]. In ACB, eNB periodically broadcasts a barring factor. When a device has data to send, it generates a random number. If the random number is equal to or greater than the barring factor, then its random access attempt is barred. However, 3GPP leaves a specific algorithm to determine the value of the barring factor as an implementation issue. There is a trade-off in the barring factor p ($0 \leq p \leq 1$) such that if a severe congestion occurs in a cell, eNB sets p to an extremely low value, and most of the devices are barred. It results in few collisions but long access delays as well as underutilization of network resources. The access delay indicates the time between the generation of an access request and the completion of the random access procedure. On the other hand, if eNB sets p to an extremely high value, collisions occur in most of the preambles.

To solve the congestion problem, several mechanisms have been proposed [7]. In [8], a mechanism has been proposed to determine when access control is enabled or disabled. eNB turns on or off its access control mechanism according to a congestion coefficient. The congestion coefficient indicates the ratio of the number of devices which fail in random access to the number of contending devices. When the congestion coefficient exceeds a specific threshold, the control mechanism is activated. The control mechanism is then deactivated once the congestion coefficient falls below the threshold.

Once a control mechanism is enabled, mechanisms to regulate the access attempts of devices are initiated. Some literature focuses on the optimization problem to determine an optimal barring factor. In [9], the barring factor is determined to maximize the expected number of devices to be successfully served in each RA slot. In [10, 11], the barring factor is selected to minimize the expected total service time. The expected total service time is the time used by all devices to successfully access RACH. To solve the optimization problem, a high cost due to complexity and long computation time is inevitable.

For practical scenarios, heuristic algorithms are proposed to adaptively change the barring factor. In [12], a reject probability of random access attempts based on traffic load using a proportional integrative derivative controller is proposed. In [13], the load state of the network is monitored based on the expected number of successful devices gaining random access. Based on the load state, the number of preamble transmissions is controlled. In [14], a cooperative mechanism to globally control congestions over multiple cells is proposed. The barring factor of each eNB is jointly decided among all eNBs rather than individually decided in each eNB to achieve global stabilization and access load sharing. In [15], focus is placed on access control over differentiated MTC services including delay-sensitive and delay-tolerant services. A RACH partition strategy is proposed that dynamically partitions and allocates RACH resources to each type of service.

These heuristic mechanisms use the current traffic load or predicted traffic load. However, when the traffic load is estimated, the conventional mechanisms usually use the number of devices successfully served in each RA slot, because eNB has no knowledge of the number of backlogged devices. In the MTC system, traffic load is divided into two parts. The first part includes the devices that did not succeed in the previous RA slot and are carried forward into the current slot. The second part includes the devices which are newly activated within the current RA slot. The number of devices in the first part depends on the barring factor in the previous slot. The number of devices in the second part may follow the traffic prediction model [1, 16]. Thus, to estimate the number of backlogged devices, it is reasonable to consider traffic load as two parts.

3. Proposed Mechanism

Our mechanism estimates the number of backlogged devices, N_B , in the next RA slot to determine an optimal barring factor. We postulate that the backlogged devices consist of two types of devices: the devices which are carried over from the previous RA slot and the devices which are newly activated in the current RA slot. As shown in Figure 1, we separate N_B prediction process into two parts and propose a new mechanism to control the barring factor dynamically. First, we forecast the newly activated load, N_{new} , in the next RA slot based on a traffic model (Figure 1(c)). Then, we estimate the load carried over, N_{carried} , in the next slot based on the barring factor in the current slot (Figure 1(b)). Finally, using the forecasted newly activated load and estimated load

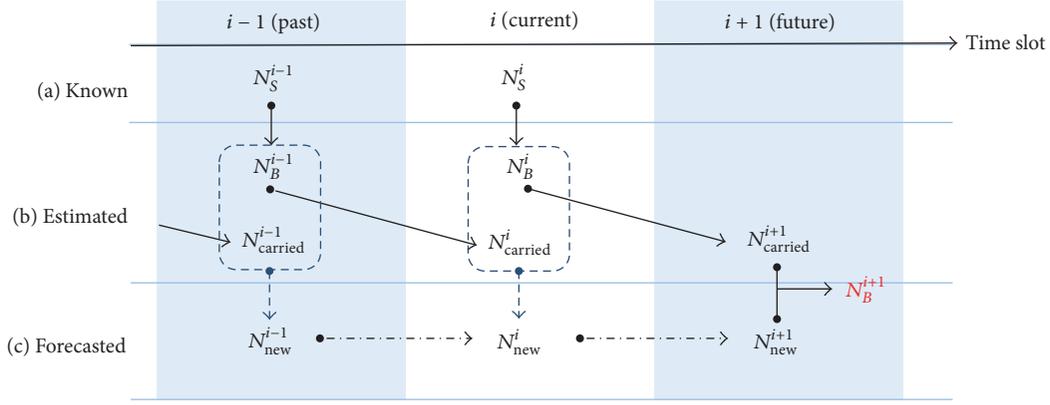

 FIGURE 1: The estimation process of N_B .

TABLE 1: Summary of symbols.

Symbol	Definition
p^i	ACB barring factor in RA slot i
M	The number of predefined preambles in the system
N_S^i	The number of devices which succeed in random access in RA slot i
N_{ACB}^i	The number of devices which pass the ACB check in RA slot i
N_B^i	The number of backlogged devices in RA slot i
N_{new}^i	The number of newly activated devices in RA slot i
$N_{carried}^i$	The number of devices which do not succeed in RA slot $i-1$ and are carried forward into RA slot i

to be carried over, we estimate N_B in the next slot. Using the estimated number of backlogged devices, we determine the barring factor p in the next slot to maximize the number of successful devices attempting random access. A summary of symbols is given in Table 1.

Our goal is to determine the barring factor p for the next RA slot, $i+1$. The eNB only knows N_S^i in the current RA slot i when it receives messages from devices in the third step of the random access procedure. Using N_S^i , we estimate N_{ACB}^i as follows:

$$N_{ACB}^i = \frac{W(N_S^i (M-1) \log((M-1)/M) / M)}{\log((M-1)/M)}. \quad (1)$$

$W(x)$ is the Lambert function. The Lambert function is the inverse relation of the exponential function of a complex number. Using the estimated N_{ACB}^i , we determine N_B^i as follows:

$$N_B^i = \frac{N_{ACB}^i}{p^i}. \quad (2)$$

We can also determine the number of devices which are carried over from the previous RA slot, $i-1$, from $N_{carried}^i = N_B^{i-1} - N_S^{i-1}$. With N_B^i and $N_{carried}^i$, we estimate $N_{new}^i = N_B^i - N_{carried}^i$.

Using the series of $\{\dots, N_{new}^{i-2}, N_{new}^{i-1}, N_{new}^i\}$, we predict future new RA attempts N_{new}^{i+1} using any quantitative forecasting method, such as moving average, exponential smoothing, trend forecast, extrapolation, and growth curves [17]. Additionally, we estimate $N_{carried}^{i+1}$ by using N_B^i and N_S^i . Thus, we can determine N_B^{i+1} .

In this paper, to determine p^{i+1} in RA slot, $i+1$, we need to know N_{ACB}^{i+1} , as well as N_B^{i+1} . We calculate N_{ACB}^{i+1} to maximize the number of devices which succeed in random access as follows:

$$\arg \max E(N_S^{i+1} | N_{ACB}^{i+1}). \quad (3)$$

The expected number of successful preamble transmissions in RA slot $i+1$ is as follows [10, 11]:

$$\begin{aligned} E(N_S^{i+1} | N_{ACB}^{i+1}) &= M \binom{N_{ACB}^{i+1}}{1} \frac{1}{M} \left(1 - \frac{1}{M}\right)^{N_{ACB}^{i+1} - 1} \\ &= N_{ACB}^{i+1} \left(1 - \frac{1}{M}\right)^{N_{ACB}^{i+1} - 1}. \end{aligned} \quad (4)$$

Thus, we can determine N_{ACB}^{i+1} as follows:

$$N_{ACB}^{i+1} = \frac{1}{\log(M/(M-1))}. \quad (5)$$

Finally, using N_B^{i+1} and N_{ACB}^{i+1} , we set the barring factor p^{i+1} as follows:

$$p^{i+1} = \frac{N_{ACB}^{i+1}}{N_B^{i+1}}. \quad (6)$$

4. Performance Analysis

We adopted the MTC traffic model for smart metering applications, which is used as an experimental scenario

TABLE 2: Simulation parameters for RACH in LTE FDD (Frequency-Division Duplex).

Parameter	Setting
Cell bandwidth	5 MHz
PRACH configuration index	6
Total number of preambles	64
Maximum number of preamble transmissions	10
Number of uplink grants per Random Access Response	3
Number of Control Channel Elements allocated for Physical Downlink Control Channel	16
Number of Control Channel Elements per Physical Downlink Control Channel	4
Ra-ResponseWindowSize	5 subframes
mac-ContentionResolutionTimer	48 subframes
Backoff indicator	20 ms

where a large number of devices access RACHs in a highly synchronized manner [1]. Smart meters are used for many fields such as automatic meter reading, energy demand management, and microelectric generation management. In the smart metering model, the housing density of an urban area of London located within a single cell was used to define the density of meters (MTC devices). We set the number of devices N to 35,670. Each device requested one data transmission on a reading frequency, T . We set the reading frequency to 5 min [18]. The number of devices which newly begin the RA procedure in the i th RA slot is defined based on the Beta-distribution traffic model as follows:

$$N_{\text{new}}^i = N \int_{t_i}^{t_{i+1}} p(t) dt. \quad (7)$$

In the above, t_i is the time of the i th RA slot and $p(t)$ below follows the Beta-distribution with values of $\alpha = 3$ and $\beta = 4$ [1].

$$p(t) = \frac{t^\alpha (T-t)^{\beta-1}}{T^{\alpha+\beta-1} \text{Beta}(\alpha, \beta)}. \quad (8)$$

Beta(α, β) is the Beta function. To evaluate the performance of our mechanism, we developed a C-based discrete-event simulator. We assumed a typical PRACH (Physical Random Access CHannel) configuration, $\text{PRACH-ConfigIndex} = 6$, where the subframe length is 1 ms. Table 2 lists the basic parameters used throughout our experiments [1].

Figure 2 shows the estimated values of N_{ACB} and N_B using (1) and (2). In the figure, to improve the legibility of charts, we plot the traffic load per 3 s (e.g., the time index), instead of per each RA slot. Traffic load N_B is the number of devices sending RACH preambles in an RA slot. When the number of devices that try to access the RACHs sharply increases, the number of devices that fail to have access due to the ACB check or collision also increases. Thus, the failed devices are carried over to the next RA slot. As shown in the figure, to analyze the accuracy of the estimated values (N_{ACB} and N_B), we compare the estimated values to the real

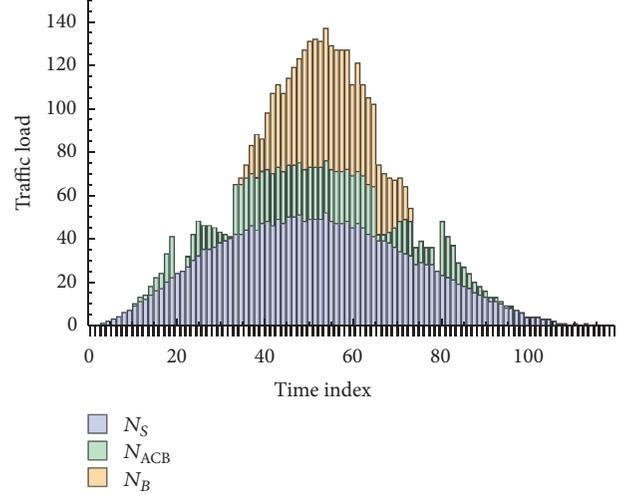


FIGURE 2: Estimated traffic load (N_{ACB}, N_B) from N_S .

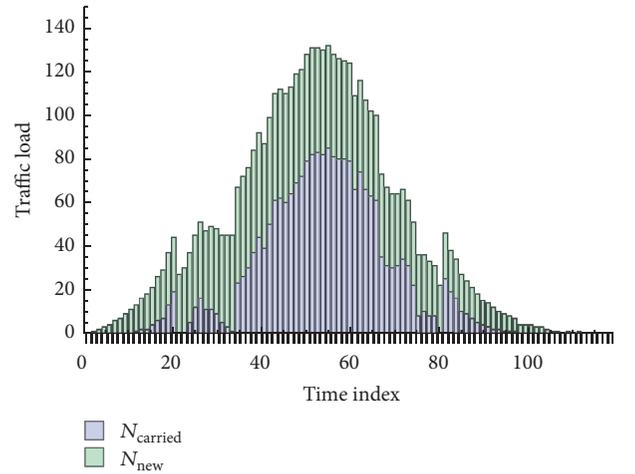


FIGURE 3: Estimated traffic load (N_B) from N_{new} and N_{carried} .

values by using the error percentage. The error percentages are approximately 11% and 14% for N_{ACB} and N_B , respectively. Figure 3 shows the estimated values of N_{new} and N_{carried} . In the experiment, we predicted the number of newly activated devices by using an autoregressive model of order 1, AR(1), in the series of previous N_{new} devices [19]. However, other quantitative forecasting methods such as moving average, exponential smoothing, or trend forecast can be used instead of AR(1).

N_{carried} in the current RA slot is calculated as the difference between N_B and N_S in the previous RA slot. As the traffic load increases, the number of failed devices increases so the number of devices carried over also increases. The number of devices carried over also influences the number of devices successfully served in the next RA slot. To analyze the accuracy of the estimated values (N_{new} and N_{carried}), we use the error percentage to compare the estimated values with the real values. The error percentages are approximately 11.5% and 15% for N_{new} and N_{carried} .

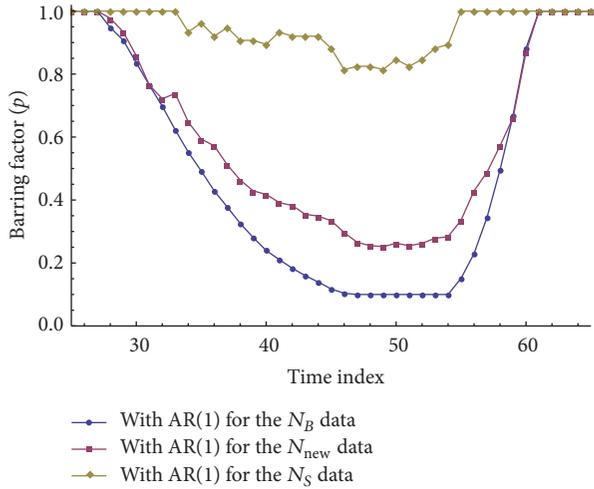


FIGURE 4: Barring factor determined by the estimated traffic load.

Figure 4 shows the variance in the barring factor. When the traffic load in the $i + 1$ slot is estimated, we used the relationship $N_B^{i+1} = N_{\text{new}}^{i+1} + N_{\text{carried}}^{i+1} = N_{\text{new}}^{i+1} + (N_B^i - N_S^i)$. Based on the estimated traffic load (N_B^{i+1}), the value of p^{i+1} is selected. In the figure, “with AR(1) for the N_{new} data” indicates our mechanism in which AR(1) is used to predict N_{new}^{i+1} using N_{new}^i and N_B^{i+1} is estimated from N_{new}^{i+1} and N_{carried}^{i+1} . The item “with AR(1) for the N_B data” indicates that AR(1) is used to predict N_B^{i+1} using N_B^i . The item “with AR(1) for the N_S data” means that AR(1) is used to predict N_B^{i+1} using N_S^i . In the case of “with AR(1) for the N_B data,” the barring factor is set conservatively and it leads to underutilization of channel resources. In the case of “with AR(1) for the N_S data,” the barring factor is set aggressively and it results in high competition. For “with AR(1) for the N_{new} data,” the barring factor is set moderately to exploit the available channel resources.

In the literature, the traffic load is usually estimated based on N_S [13]. We compare the performance of our mechanism (“with AR(1) for the N_{new} data”) with the conventional mechanism (“with AR(1) for the N_S data”). We also use the original ACB [6] as a reference. In the original ACB, once the congestion control is activated, the barring factor is set to 0.1. We used three performance metrics in the experiments, access success probability, collision probability, and access delay. The access success probability is defined as the ratio of the number of successful devices to the number of contending devices in an RA slot. The collision probability is the ratio of the number of occurrences when two or more devices with the same preamble try to access RACH to the total number of random access preambles. In Figures 5, 6, and 7, the cumulative density functions (CDFs) of the three metrics are shown.

In Figure 5, the measured access success probability is depicted. When the values of the three mechanisms at the 90th percentile were compared for the CDFs, the probability of “with AR(1) for the N_{new} data” is slightly better than “with AR(1) for the N_S data” and “original ACB.” In Figure 6, the

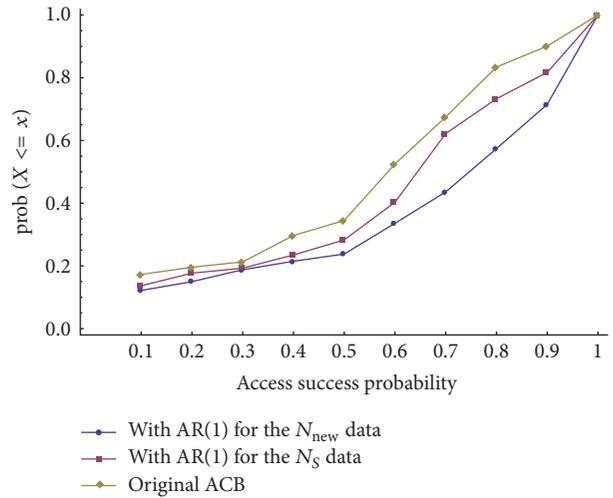


FIGURE 5: Performance evaluation with access success probability.

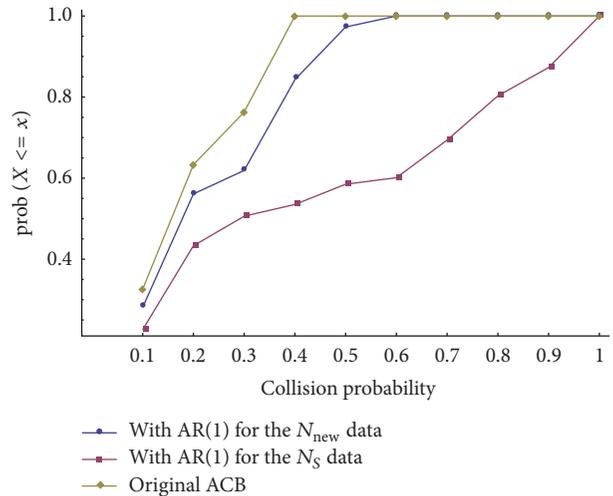


FIGURE 6: Performance evaluation with collision probability.

comparison of collision probabilities of the three mechanisms is shown. In the case of “original ACB,” because only 10% of the devices can attempt random access, the collision probability is the smallest. When the other two mechanisms are compared, our proposed mechanism has approximately 51% lower collision probability than the conventional mechanism at 90th percentile. Figure 7 shows the access delay. The performance of “original ACB” is the worst. The access delay of our mechanism is approximately 41% and 260% better than “with AR(1) for the N_S data” and “original ACB,” respectively.

In “original ACB,” the collision probability decreases but the access delay invariably increases due to the conservative approach. In “with AR(1) for the N_S data,” the aggressive approach causes the collision probability to increase and the access success probability to decrease. In our mechanism, “with AR(1) for the N_{new} data,” the barring factor is controlled to maximize the number of successful devices requesting random access. Thus, the success probability and the access delay are better than the other two mechanisms. However, the

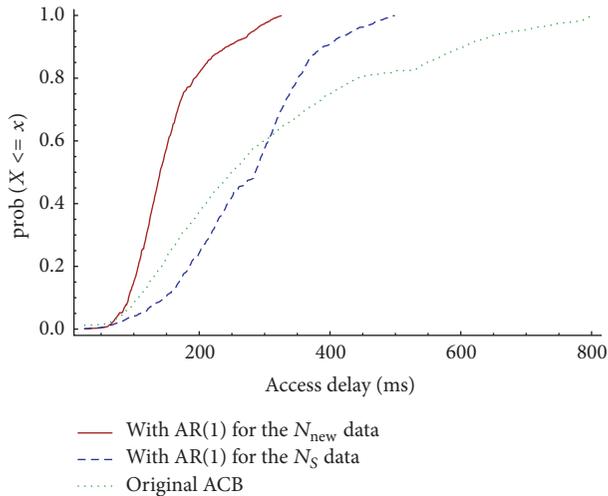


FIGURE 7: Performance evaluation with access delay.

performance of our mechanism depends on the accuracy of the prediction methods applied. In detail, AR(1) method is used to predict future new RA attempts for smart metering applications. According to the type of applications, the performance of the prediction method can be different. In order to further evaluate the accuracy problem, a compensation method can be considered in future research.

5. Conclusion

To provide efficient data forwarding in MTC networks for IoT applications, the congestion problem due to the burst traffic needs to be solved. To address this problem, we present a mechanism to dynamically control the barring factor in the 3GPP ACB mechanism based on traffic load prediction. To predict the traffic load, we consider two types of traffic. The first type of traffic indicates the traffic load from newly activated devices. The second type of traffic indicates the traffic load from devices that failed to have access in the last cycle and retry random access in the new cycle. Considering the predicted traffic load, we adjust the barring factor to maximize the number of devices that succeed in the new random access cycle. To evaluate the effectiveness of our mechanism, we compare with other mechanisms for smart metering applications. The results show that the proposed mechanism improves the performance in terms of the access success probability and access delay.

For future work, we improve our traffic regulation mechanism to meet different QoS requirements of MTC devices for diverse application scenarios such as smart cities and smart homes.

Conflicts of Interest

The authors declare no conflicts of interest.

Acknowledgments

This research was supported by Basic Science Research Program through the National Research Foundation of

Korea (NRF) funded by the Ministry of Education (NRF-2015R1D1A1A01060117 and NRF-2015R1D1A1A09057141).

References

- [1] "3rd Generation Partnership Project, Technical specification group radio access network; study on RAN improvements for machine-type communications, 3GPP TR 37.868 V11.0.0, 2011".
- [2] "3rd Generation Partnership Project, Evolved universal terrestrial radio access (E-UTRA); radio resource control (RRC); protocol specification, 3GPP TS 36.331 V12.7.0, 2015".
- [3] P. Jain, P. Hedman, and H. Zisimopoulos, "Machine type communications in 3GPP systems," *IEEE Communications Magazine*, vol. 50, no. 11, pp. 28–35, 2012.
- [4] M.-Y. Cheng, G.-Y. Lin, H.-Y. Wei, and A. C.-C. Hsu, "Overload control for machine-type-communications in LTE-advanced system," *IEEE Communications Magazine*, vol. 50, no. 6, pp. 38–45, 2012.
- [5] A. Laya, L. Alonso, and J. Alonso-Zarate, "Is the random access channel of LTE and LTE-A suitable for M2M communications? A survey of alternatives," *IEEE Communications Surveys & Tutorials*, vol. 16, no. 1, pp. 4–16, 2014.
- [6] M. Hasan, E. Hossain, and D. Niyato, "Random access for machine-to-machine communication in LTE-advanced networks: issues and approaches," *IEEE Communications Magazine*, vol. 51, no. 6, pp. 86–93, 2013.
- [7] R.-G. Cheng, J. Chen, D.-W. Chen, and C.-H. Wei, "Modeling and analysis of an extended access barring algorithm for machine-type communications in LTE-A networks," *IEEE Transactions on Wireless Communications*, vol. 14, no. 6, pp. 2956–2968, 2015.
- [8] Intel Corporation, "Further performance evaluation of EAB information update mechanisms, 3GPP R2-120270, RAN WG2 Meeting #77," 2012.
- [9] Z. Wang and V. W. S. Wong, "Optimal access class barring for stationary machine type communication devices with timing advance information," *IEEE Transactions on Wireless Communications*, vol. 14, no. 10, pp. 5374–5387, 2015.
- [10] S. Duan, V. Shah-Mansouri, and V. W. S. Wong, "Dynamic access class barring for M2M communications in LTE networks," in *Proceedings of the IEEE Global Communications Conference (GLOBECOM '13)*, pp. 4747–4752, IEEE, Atlanta, Ga, USA, December 2013.
- [11] S. Duan, V. Shah-Mansouri, Z. Wang, and V. Wong, "D-ACB: adaptive congestion control algorithm for bursty M2M traffic in LTE networks," *IEEE Transactions on Vehicular Technology*, no. 99, p. 1, 2016.
- [12] A. Ksentini, Y. Hadjadj-Aoul, and T. Taleb, "Cellular-based machine-to-machine: overload control," *IEEE Network*, vol. 26, no. 6, pp. 54–60, 2012.
- [13] J.-P. Cheng, C.-H. Lee, and T.-M. Lin, "Prioritized Random Access with dynamic access barring for RAN overload in 3GPP LTE-A networks," in *Proceedings of the IEEE GLOBECOM Workshops (GC Wkshps '11)*, pp. 368–372, Houston, Tex, USA, December 2011.
- [14] S.-Y. Lien, T.-H. Liao, C.-Y. Kao, and K.-C. Chen, "Cooperative access class barring for machine-to-machine communications," *IEEE Transactions on Wireless Communications*, vol. 11, no. 1, pp. 27–32, 2012.
- [15] Q. Du, W. Li, L. Liu, P. Ren, Y. Wang, and L. Sun, "Dynamic RACH partition for massive access of differentiated M2M services," *Sensors*, vol. 16, no. 4, 2016.

- [16] M. Laner, P. Svoboda, N. Nikaein, and M. Rupp, "Traffic models for machine type communications," in *Proceedings of the 10th IEEE International Symposium on Wireless Communication Systems 2013, ISWCS 2013*, pp. 651–655, deu, August 2013.
- [17] C. M. Chou, C. Y. Huang, and C.-Y. Chiu, "Loading prediction and barring controls for machine type communication," in *Proceedings of the 2013 IEEE International Conference on Communications, ICC 2013*, pp. 5168–5172, Hungary, June 2013.
- [18] Verizon, "Smart grid traffic behaviour discussion," *RAN WG2 Meeting #69b*, Article ID 3GPP R2-102340, 2010.
- [19] R. Bos, S. De Waele, and P. M. T. Broersen, "Autoregressive spectral estimation by application of the Burg algorithm to irregularly sampled data," *IEEE Transactions on Instrumentation and Measurement*, vol. 51, no. 6, pp. 1289–1294, 2002.



Hindawi

Submit your manuscripts at
<https://www.hindawi.com>

