Random Access for Machine-to-Machine Communication in LTE-Advanced Networks: Issues and Approaches

Monowar Hasan and Ekram Hossain, University of Manitoba
Dusit Niyato, Nanyang Technological University

ABSTRACT

Machine-to-machine communication, a promising technology for the smart city concept, enables ubiquitous connectivity between one or more autonomous devices without or with minimal human interaction. M2M communication is the key technology to support data transfer among sensors and actuators to facilitate various smart city applications (e.g., smart metering, surveillance and security, infrastructure management, city automation, and eHealth). To support massive numbers of machine type communication (MTC) devices, one of the challenging issues is to provide an efficient way for multiple access in the network and to minimize network overload. In this article, we review the M2M communication techniques in Long Term Evolution-Advanced cellular networks and outline the major research issues. Also, we review the different random access overload control mechanisms to avoid congestion caused by random channel access of MTC devices. To this end, we propose a reinforcement learning-based eNB selection algorithm that allows the MTC devices to choose the eNBs (or base stations) to transmit packets in a self-organizing fashion.

INTRODUCTION

To achieve integrated management of cities, the concept of a *smart city* focuses on applying next-generation information technologies in our daily life and forming the Internet of Things (IoT). IoT for a smart city can be achieved through emerging technologies such as machine-to-machine (M2M) communication and cloud computing. As wireless connectivity is rapidly expanding beyond traditional mobile devices, wireless M2M communication is expected to enable full automation of sensors and actuators used in various smart city applications. M2M communication, also known as machine type communication (MTC), allows not only sensors and actuators, but also computers and mobile phones to communicate over a network without or with minimal human intervention.[1] M2M communication, under the process of standardization by the Third Generation Partnership Project (3GPP), can support a wide range of applications to design a smart city, for example:

- *Smart metering*: monitoring smart grids and smart garbage bins
- *Surveillance and security*: secure access and monitoring in city buildings and neighborhoods
- *Infrastructure management*: load sensing for critical infrastructures, managing historical sites
- *City automation*: smart parking system, traffic monitoring, real-time travel and route updates
- *eHealth*: health updates by wearable health monitors [1]

According to the 3GPP proposal, the higher layer connections among MTC devices are provided by attaching MTC devices to an existing cellular infrastructure (e.g., Long Term Evolution-Advanced [LTE-A]). For M2M communication, generally small amounts of data need to be transmitted from a huge number of devices. Therefore, the devices perform initial and periodic random access for delivering resource requests to the network. In LTE-A, MTC and H2H devices[2] can perform random access (RA) using the physical random access channel (PRACH). Although the data size is small, when a large number of MTC devices try to communicate over the same channel, the devices contend to access the shared radio channels and create the network overload problem.

In this article, we give an overview of M2M communication and the random access procedure in LTE-A networks. The major issues related to M2M communication in LTE-A are outlined. The current state of the art of the proposals to address the network overload problem due to random access of the MTC devices are reviewed next. To this end, a base station selection method is proposed to alleviate the overload problem in M2M communication.

---

[1] *By the term* MTC devices, *we refer to devices communicating without human intervention.*

[2] *By the term* H2H devices, *we refer to traditional (human) cellular users.*

0163-6804/13/$25.00 © 2013 IEEE

M2M ACCESS METHODS AND M2M COMMUNICATION IN LTE-A

In LTE-A, the MTC devices contend for the resource blocks (RBs) for their data transmission using an RA method. Such an RA mechanism uses the contention resolution method based on a uniform backoff algorithm, the details of which are described later in this article. Also, the network architecture for M2M communication in the LTE-A system is briefly explained.

M2M ACCESS METHODS

The access network connects the MTC devices to the infrastructure, which can be either *wired* (i.e., cable, xDSL, and optical) or *wireless*. Wireless access methods can be either *capillary* (i.e., WLAN, ZigBee, and IEEE 802.15.4x) or *cellular* (i.e., General Packet Radio Service [GPRS], 3G, LTE-A, and WiMAX). Although the wired solution can provide high reliability, high rates, and small delay, it is not suitable for smart city implementation due to cost effectiveness, lack of scalability, and mobility. Alternatively, the wireless capillary solution, mainly used for shared short-range links, is less expensive, generally scalable, and low power. However, low rate, weak security, interference, and lack of universal infrastructure/coverage limit its applicability to smart city applications. On the other hand, the wireless cellular solution provides ubiquitous coverage, mobility, roaming, and security, which makes cellular M2M a promising infrastructure-based solution for smart city rollouts. Among the available cellular solutions, LTE-A for M2M communication can be a conceivable choice due to its *longevity* (i.e., long-term deployment of infrastructure), *lower service cost* (compared to 2G or 3G), and *scalability*. In the following, we provide a review of the cellular M2M access method in the LTE-A network.

LTE-A NETWORK ARCHITECTURE FOR M2M COMMUNICATION

The LTE-A network comprises two parts: the core network (CN) and the radio access network (RAN) [2]. The CN is responsible for overall control of mobile devices and establishment of an IP packet flow. The RAN is responsible for wireless communication and radio access. The RAN, which provides necessary user and control plane protocols for communicating with mobile devices in LTE-A, consists of base stations, referred to as evolved Node-Bs (eNBs). A mobile device is referred to as user equipment (UE). The UE can be either an H2H or MTC device. eNBs are interconnected through the X2 interface. Besides, the RAN (i.e., eNB) is connected to the CN through the S1 interface. A high-level architecture of LTE-A networks with M2M communication is shown in Fig. 1, where the MTC devices are connected to the eNBs either directly or via the MTC gateway (MTCG). The MTCG is responsible for providing a suitable path and facilitating local control for M2M communication. The MTC devices can also communicate with each other directly. The eNB-to-MTCG wireless links follow the 3GPP LTE-A specification. However, the MTCG-to-M2M and M2M-to-M2M links can be based on either LTE-A or other wireless standards (e.g., 3G, low-power WiFi, and IEEE 802.15.x [3]).

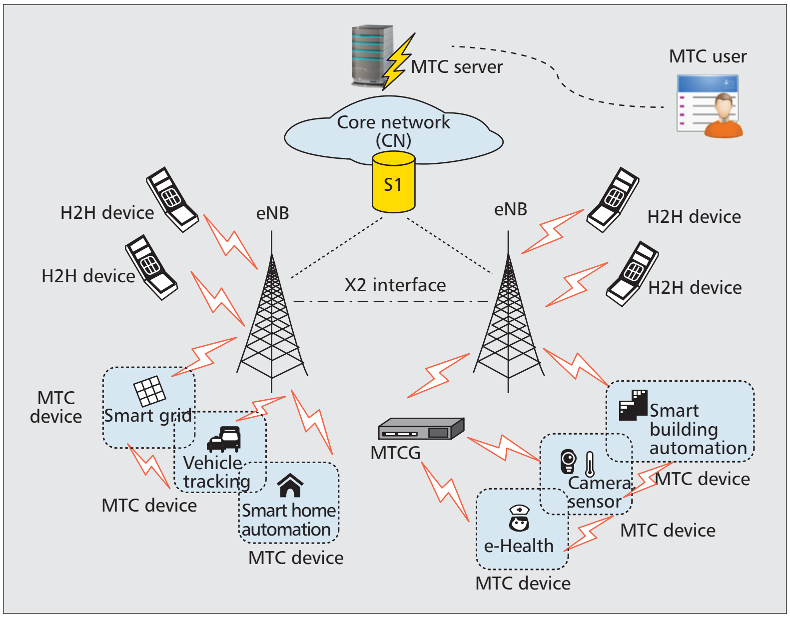


Figure 1. *Machine-to-machine communication in LTE-A networks.*

RESOURCE BLOCK AND RANDOM ACCESS PROCEDURE

The minimum resource scheduling unit for downlink and uplink transmission is referred to as an RB. One RB consists of 12 subcarriers (180 kHz) in the frequency domain and one subframe (1 ms) in the time domain. Details of the LTE-A physical layer is provided in [2, Sec. 5], and a brief review of M2M scheduling and signaling over LTE-A is provided in [4]. When a UE has packets to transmit, it performs random access (RA) during an allowable time slot, called an access grant time interval (AGTI) or RA opportunity (i.e., RA-slots). The time-frequency resource (i.e., RB) on which RA is performed is the PRACH. RA allows MTC devices to request a connection initialization. The RA-slots have bandwidth corresponding to six RBs (1.08 MHz) in the frequency domain, and the basic duration of an RA-slot is 1 ms in the time domain.

The RA process [2, 5, 6] is initiated by a UE device on two occasions: 1) when it is turned on, and it has either not acquired or lost the uplink timing synchronization; and 2) handover from one eNB to another is performed. In the LTE-A, RA can be used for initial access to establish a radio link, resource request when no uplink radio resource has been allocated, scheduling request if no dedicated scheduling request has been configured (i.e., no dedicated physical uplink control channel [PUCCH] is available), and re-establishing a radio link after failure.

The RA procedure in LTE-A is classified into two types: *contention-based* and *contention-free*. A UE device normally initiates RA in a contention-based manner by randomly choosing a preamble. A preamble is an orthogonal frequen-

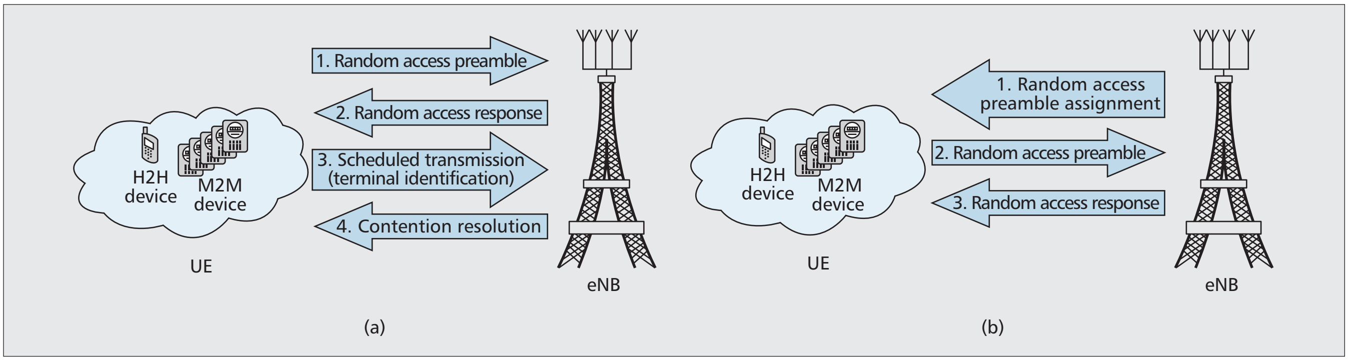


Figure 2. *Random access procedures in LTE-A: a) contention-based RA procedure; b) contention-free RA procedure.*

cy-division multiplexing (OFDM)-based signal using narrower subcarrier spacing, and therefore is not orthogonal to the PUCCH and physical uplink shared channel (PUSCH). More than one device can possibly choose the same preamble, which necessitates further contention resolution processes. In the contention-free RA, an eNB has explicit control over devices to initiate RA by using dedicated preambles. Hence, it is faster than the contention-based approach and mainly used in handover where the time delay is crucial.

Figure 2a illustrates the contention-based RA procedure, which consists of the following four steps:

- Preamble transmission
- Random access response (RAR)
- Scheduled transmission (terminal identification)
- Contention resolution

In step 1, a UE device (either an MTC or H2H device) selects one of the preamble sequences (i.e., a code sequence) assigned to the corresponding cell for contention-based RA and transmits it using the PRACH (i.e., RA-slots). If two or more devices simultaneously transmit different preamble sequences, the codes used to carry the requests from different devices will be different. Therefore, the eNB will be able to decode the requests from the different devices, and hence collision will not occur. However, collision occurs when the devices are trying to transmit the same preamble on the PRACH. Before a UE device transmits a preamble to the eNB, the UE device must know the required parameters (i.e., random access response [RAR] window size, maximum number of allowable retransmissions, power of transmission based on retransmission times, and available set of PRACH resources for transmitting preambles), which is broadcast by the eNB [6].

After receiving an RA preamble, the eNB transmits a RAR on the physical downlink shared channel (PDSCH) in step 2. The RAR contains an ID indicating the time-frequency slot in which the preamble was detected, an uplink scheduling grant for transmitting the next step message, uplink timing information to synchronize subsequent uplink transmissions, and assignment of a temporary identifier, the cell radio network temporary identifier (C-RNTI), for further communication between UE and eNB. The UE expects to receive RAR within a timing window. If the UE device does not receive the response within the configured time window, the access attempt is considered to have failed, and the UE retransmits the preamble after a random backoff time. The eNB may include a *backoff indicator* in the RAR message to inform the UE to back off for a period of time before retrying the RA procedure. After receiving the RAR, the UE synchronizes its uplink transmission timing and continues to the next step. In step 3, the actual RA message (i.e., radio resource request, scheduling request, or tracking area update), which also contains a temporary C-RNTI and either a C-RNTI (if the device has been assigned C-RNTI) or a unique identity (i.e., international mobile subscriber identity [IMSI]), is transmitted using the PUSCH. If preamble collision occurs in step 1, the colliding devices will receive the same temporary C-RNTI in RAR. Consequently, the colliding UE devices may not be decoded by the eNB (even if one UE device is decoded, others may not be aware of the collision), and a further contention resolution procedure will be required in the next step. The last step involves contention resolution using the PDSCH. If the eNB can decode any of the messages from step 3, it replies with the identifier (i.e., C-RNTI or IMSI) and only the UE device, which can detect its own ID, acknowledges the message. The rest of the colliding devices should discard the message and try to initiate another RA procedure after a random backoff.

A contention-free RA procedure, shown in Fig. 2b, is simplified with three steps by allocating dedicated preamble signatures to a UE device. The procedure starts with the RA preamble assignment by the eNB. After the transmission of an assigned RA preamble by the UE, the eNB responds with the RAR. This is the last step of the contention-free RA since there is no need to resolve further collision. In this article, we concentrate on the contention-based RA since MTC devices will mostly transmit packets randomly and need to contend for RBs.

RESEARCH CHALLENGES AND POSSIBLE SOLUTIONS FOR RA-BASED M2M COMMUNICATION

The major research challenges and some alternatives for RA-based M2M communication can be summarized as follows.

PRACH overload control: The amount of data transmitted by MTC devices is small in most cases. However, the number of MTC devices in a cell is expected to be very large [7]), and their frequency of making data connections is much higher than that of H2H devices. When a large number of MTC devices try to access the network simultaneously, it leads to a low RA success rate and high network congestion in the PRACH. This may cause unexpected delays, packet loss, waste of radio resources, extra energy consumption, and even service interruption. The channel can be further overloaded when the MTC devices repeat their access attempts after collisions. Therefore, efficient overload control mechanisms are required for RA-based M2M communication.

Mode selection and quality of service (QoS) provisioning: Both H2H and MTC devices will contend for their data transmission channel using the same set of PRACHs. However, the MTC devices should not collide with H2H devices and at the same time need to satisfy their own QoS requirements. As already mentioned, the MTC devices can communicate either directly with the eNB or via the MTCG. Selection of the eNB or MTCG, especially through RA, may be based on signal strength or link condition (channel gain). However, in some practical cases, choosing the optimal mode might need to consider the required contention-based packet-level QoS performances. For example, although the signal strength is strong in the direct transmission mode, selecting MTCG may provide a better success RA probability or lower packet transmission delay. Therefore, the optimal mode selection should consider the QoS (i.e., high data rate and lower delay) requirements in the context of MTC. In addition to mode selection, when the MTC devices are in the coverage region of multiple eNBs or MTCGs, selection of eNB or MTCG for RA is also important.

To guarantee QoS and balance the network load, it is necessary to adopt an efficient eNB (MTCG) selection mechanism. Unfortunately, until now most of the approaches proposed in the literature do not consider eNB or MTCG selection mechanisms while addressing RA overload control. However, from our perspective, the *reinforcement learning* approach (i.e., Q-learning) can be useful where MTC devices select an eNB or MTCG based on their perceived QoS parameters. Later in this article we propose an eNB selection method.

Efficient group management: Group management and addressing (i.e., selection of a group coordinator) of MTC devices could be vital for M2M communication. When the number of groups as well as the number of members in the groups grow large, the challenging issue is to allocate RA-slots and reduce signaling overhead. In addition, an efficient algorithm for preamble allocation is necessary. *Game theory* can be a useful tool to optimize preamble allocation. In addition, formation and reformation of groups should be adaptive based on a traffic condition. Also, there must be provisioning for peer-to-peer communication between group members.

Opportunistic RA (cognitive M2M): Due to a large number of MTC devices, there may not be enough resources to accommodate all MTC devices. It would be beneficial to control network congestion if the MTC devices are able to access the radio channels opportunistically. Enabling cognitive radio capability in the MTC devices for random access can be a propitious area of research.

In the next section we briefly review the existing approaches for controlling PRACH overload to support M2M communication in LTE-A networks.

PRACH OVERLOAD CONTROL MECHANISMS

To support MTC in LTE-A, the following solutions have been proposed for controlling PRACH overload.

Access class barring (ACB) scheme: In a legacy ACB mechanism, initially an eNB broadcasts an access probability (AP) and access class (AC) barring time. When a device initiates RA, the device draws a random number between zero and one, and compares this with AP. If the number is less than AP, the device is allowed to proceed with the RA process. Otherwise, the device delays for the AC barring duration. In the recent releases of LTE-A (Release 10 and on), the existing ACB mechanism is extended to allow one or more new ACs for MTC devices, and an individual access barring factor can be assigned for each of the classes.

3GPP also proposes an extended access barring (EAB) scheme [7] in which when EAB is activated, the devices belonging to certain ACs (i.e., delay-tolerant devices) are not allowed to perform RA. Using the ACB mechanism, the eNB can control PRACH overload by lowering the value of access probability. Although this reduces RA attempts, it may cause longer RA delays to some devices.

In [8], a cooperative ACB scheme is proposed without considering priorities among devices where the eNBs select the ACB parameter jointly based on the level of network congestion. An optimization problem is formulated to balance the number of devices attached to each eNB. The approach does not consider priorities among devices.

A prioritized RA (PRA) scheme is proposed in [9] where PRACH opportunities (i.e., RA-slots) are pre-allocated to different MTC classes, and a class-dependent backoff procedure is used.

PRACH resource separation scheme: A resource separation scheme can be used to allocate orthogonal PRACH resources (i.e., preambles) to H2H and MTC devices. In [10], two RA preamble separation approaches are compared. In the first approach, the set of available RA preambles is completely split into two subsets, one for only H2H devices and another for MTC devices. In the second approach, the preambles are also split into two subsets, but one set of preambles is dedicated to H2H devices only, and another set is shared by both H2H and MTC devices. After passing the ACB procedure, the devices can transmit preambles using RA. However, selection of eNBs and backoff procedure

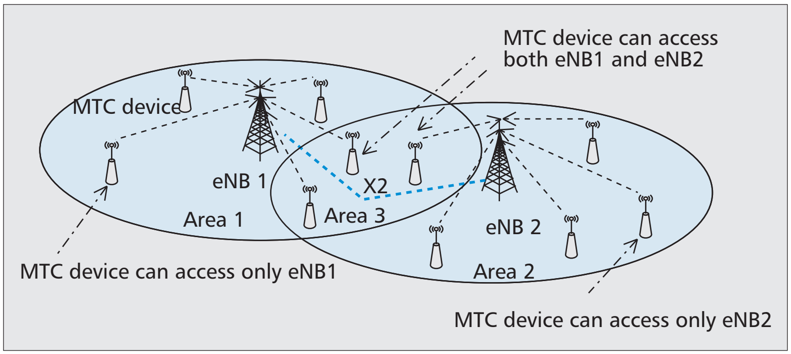


Figure 3. *MTC devices in the overlapping coverage area of multiple eNBs.*

are not considered in any of these methods.

Slotted access scheme: In this scheme, each MTC device is allowed to perform RA only in its dedicated access slot. The MTC device can calculate the allowable access slots through its ID and RA-cycle. The RA-cycle is an integer number multiple of a radio frame that is broadcast by the eNB. If the total number of unique access slots is smaller than the number of MTC devices in a particular cell, several MTC devices share the same access slot, and collision can occur. Increasing the RA-cycle can reduce collision but creates unacceptable delay of an RA request.

Dynamic allocation of RA resources: In a dynamic resource allocation approach, eNBs can dynamically allocate PRACH resources based on PRACH overload and overall network load. When a subframe is used for the PRACH, part of that subframe cannot be used for data transmission. To meet the QoS requirement and reduce delay, the eNB in a cell should use a certain number of subframes for the PRACH. This issue is addressed in [11]. An optimization problem is formulated to minimize the number of subframes allocated to the PRACH where the expected RA delay is less than a given delay bound.

Based on the 3GPP proposed slotted access scheme, a self-optimizing algorithm is proposed in [12] where the eNBs can automatically increase or decrease the number of RA-slots for preamble transmission based on channel load.

Grouping or clustering of MTC devices: According to 3GPP, MTC devices can form groups for radio resource allocation, and there should also be a mechanism to associate the MTC devices with one or more MTC groups. A group-based scheme is proposed in [13] by forming clusters based on QoS requirements.

Devices can be grouped according to an application type or geographical location. Then the group head is selected, which will communicate with the eNB on behalf of group members. As peer-to-peer communication between MTC devices is supported by LTE-A, the group head can receive requests from group members and relay the request to the eNB. A group-based scheme considering energy efficiency is proposed in [14] where the group head guarantees low energy consumption by limiting the access among the MTC devices and eNB. The objective of this scheme is to minimize overall energy consumption by the MTC devices in a group.

MTC-specific backoff scheme: The backoff scheme is used to delay the RA attempts of H2H and MTC devices separately. In this scheme, the backoff time for the H2H devices is set to a small value, while the backoff time for the MTC devices is set to a large one. Although the scheme improves performance in low channel overload, it cannot solve the congestion problem in high overload situations when a huge number of MTC devices perform RA at the same time.

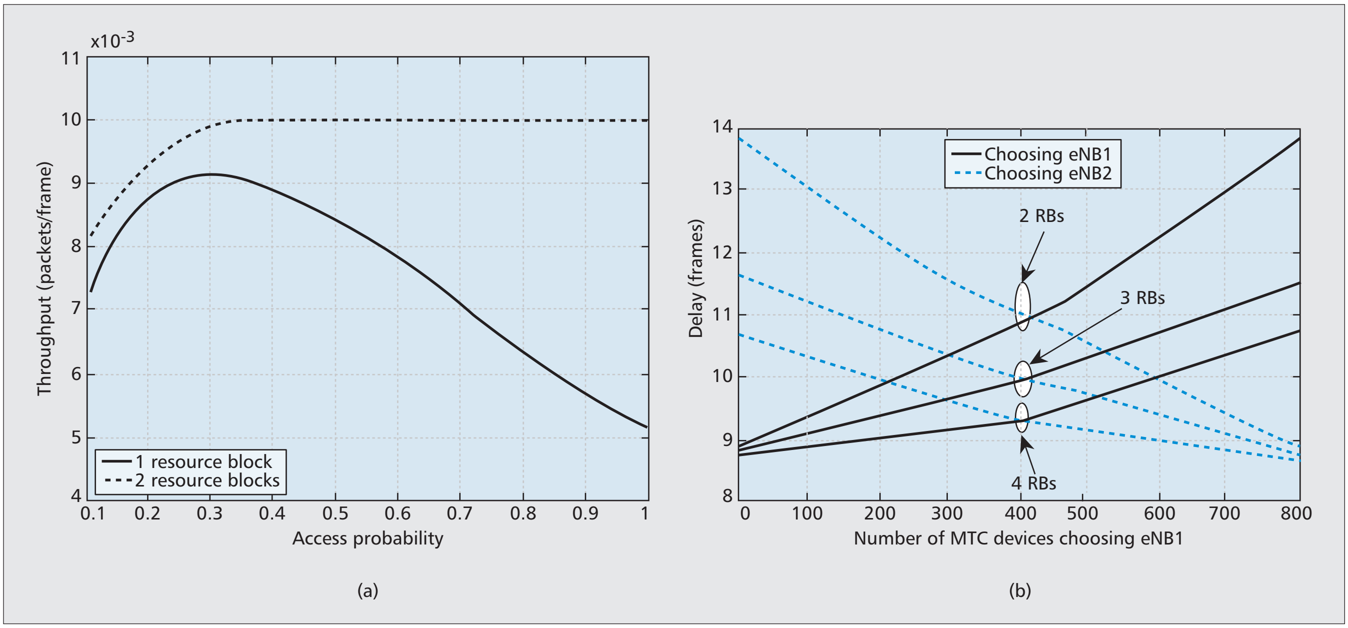
Pull-based scheme: In the pull-based scheme, an MTC server requests the eNB to page MTC devices. Upon receiving a paging message from the eNB, the MTC device will start RA. In this centralized scheme, the eNBs can control the number of devices to be paged based on PRACH load and resource availability. However, the scheme requires extra control channel resources to page massive MTC devices.

The problem of eNB selection was not considered in the overload control schemes described above. An efficient method for eNB selection along with an effective overload control method will be required for RA-based M2M communication. An MTC device in an overlapping area of multiple eNBs can choose the eNB that maximizes its QoS performance. In this case, the MTC device will observe, learn, and adapt the eNB selection decision independently. In the next section, we demonstrate the use of a reinforcement learning algorithm to address the problem of eNB selection by MTC devices. Specifically, this is a load balancing method to distribute MTC devices among the available eNBs, which can thus control the overload at each eNB.

REINFORCEMENT LEARNING-BASED eNB SELECTION

The MTC devices contend for RBs through the RA procedure described earlier, and a contention resolution method based on a uniform backoff algorithm is applied. When an MTC device has a packet to transmit, first it performs an ACB check. If the MTC device passes the ACB check, it randomly chooses an integer number between 0 and the maximum backoff counter value. Then the backoff counter decreases by one in every time slot. When the backoff counter becomes zero, the MTC device transmits the packet. The MTC device waits for an acknowledgment from the eNB. If the MTC device does not receive the acknowledgment, it assumes that the transmitted packet has collided with the transmission(s) of other device(s). Therefore, the MTC device performs retransmission by repeating the same process. The retransmission is allowed for a given maximum number of times. When the maximum number of retransmissions is reached, the MTC device discards the packet and proceeds to transmit the next one.

Given the above transmission process, the MTC device can measure the QoS performance (e.g., packet transmission delay). The MTC device can choose a proper eNB for its communication (e.g., to minimize the packet delay). To make a suitable decision on eNB selection, the following reinforcement learning (i.e., *Q*-learning) algorithm is applied.



(a)

(b)

Figure 4. *a) Throughput of the MTC device given different access probability and available RB; b) packet delay performance of MTC devices choosing different eNBs.*

1. Each MTC device initializes the Q-value $Q(s, a)$, where s is the state (i.e., current selected eNB), and a is the action (i.e., the new eNB to select), for all states and actions.
2. The MTC device performs an *exploration* step with a small probability (e.g., 0.03) by randomly choosing action a. Otherwise, the MTC device performs an *exploitation* step by choosing an action a such that $a = \arg\max_{a'} Q(s, a')$.
3. The MTC device observes the QoS performance (i.e., packet delay) at the current state s given action a denoted by $D_{s,a}$.
4. The MTC device updates Q-value as follows:

$$Q(s,a) \leftarrow (1-\kappa)Q(s,a) + \kappa(D_{s,a} + \gamma \max_{a'} Q(s',a')) \tag{1}$$

where κ is the learning rate (e.g., $\kappa = 0.5$) and γ is the discount rate (e.g., $\gamma = 0.1$). s' is the next state if action a is taken at state s.

5. Go to step 2.

The performance results for the above reinforcement-learning-based eNB selection are obtained by simulations using MATLAB. The simulation focuses on the ACB scheme and the contention resolution method in the medium access control (MAC) layer. Specifically, a discrete-time simulation is implemented, where the time is divided into frames. Each frame is then divided into subframes. The action event is generated at the beginning of each frame, and the result (e.g., success or collision) is observed at the end of the frame. Based on the results, the devices decide the next action (e.g., retransmission) for the next frame. There are two eNBs denoted by eNB1 and eNB2. The packet generation rate of the MTC devices is 0.001 packets/frame. eNB1 serves 100 MTC devices (i.e., these devices are in area 1 of Fig. 3); eNB2 serves 200 MTC devices (i.e., these devices are in area 2 of Fig. 3). There are 800 devices to select from between eNB1 and eNB2 (i.e., these devices are in area 3 of Fig. 3). The maximum backoff counter and number of retransmissions are set to 15 and 5, respectively.

Figure 4a shows the throughput of the MTC device from Area 3 (in Fig. 3) when the access probability of the ACB check is varied. When multiple MTC devices contend for one RB, congestion can occur. As a result, if the access probability is large, the throughput drops due to heavy collision. However, when two RBs are available, the congestion can be alleviated, and the throughput is higher compared to that for one RB. Also, the access probability can be increased without throughput drop. By allocating different numbers of RBs to the MTC devices, the performance can be controlled.

Next, we consider the delay performance when the number of MTC devices in area 3 of Fig. 3 choosing eNB1 is varied. Figure 4b shows the results on delay performance. Clearly, when the number of MTC devices choosing eNB1 increases, the packet delay of the MTC devices choosing eNB1 increases due to higher congestion. When more MTC devices choose eNB1, fewer MTC devices are connected with eNB2. As a result, the congestion and delay at eNB2 decrease. Again, the delay performance depends on the allocated RBs from the eNB. The MTC devices, which have choices of eNBs to which to connect, can observe the performance and adjust the eNB selection accordingly.

Figure 5 shows the number of MTC devices choosing eNB1 using the reinforcement-learning-based eNB selection algorithm. With the proposed eNB selection algorithm, MTC devices have the ability to switch to the eNB that provides better performance (i.e., smaller delay).

Clearly, when the number of allocated RBs by an eNB is higher, the performance will be improved, and more MTC devices will choose that eNB. In addition, allocation of RBs by one eNB will impact the selection of other eNBs. Figure 6 shows the delay performance of MTC devices in area 3 of Fig. 3, where the random eNB selection algorithm is adopted for comparison. With random eNB selection, MTC devices randomly choose the eNBs to transmit their data with equal probability. Clearly, the proposed reinforcement-learning-based algorithm outperforms random eNB selection by achieving lower delay.

Given the reinforcement-learning-based eNB selection algorithm, eNBs can allocate RBs to achieve their desired objectives. For example, eNBs can jointly optimize the allocation (e.g., using an optimization method) or non-cooperatively allocate the RBs (e.g., using game theory).

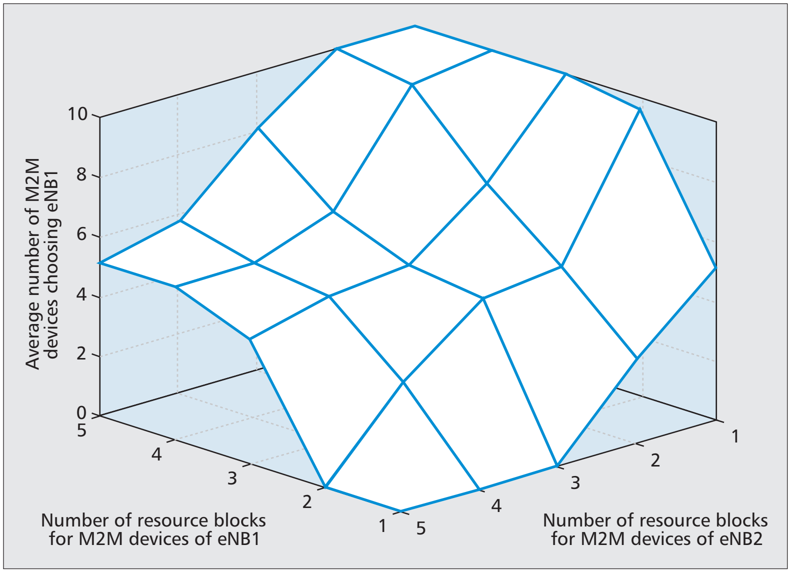


Figure 5. *Number of MTC devices choosing eNB1 under different number of available RBs.*

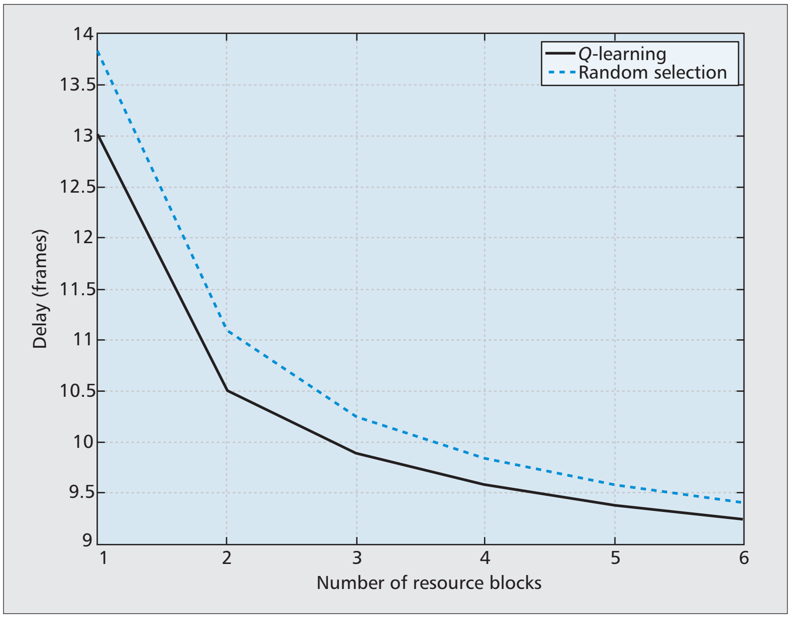


Figure 6. *The average delay compared between the proposed reinforcement-learning-based algorithm and random eNB selection.*

CONCLUSION

Emerging technologies such as M2M communication will enable the smart city to become a reality. M2M communication will be an important component of the future Internet-of-Things for linking machine type communication devices with the Internet. IoT will play a major role in smart city to support many intelligent applications. One of the major challenges in M2M communication is efficient random access of a large number of devices and avoiding network congestion. In this article, we have provided a comprehensive survey of various approaches to control network congestion for random-access-based M2M communication. Along with an effective random access procedure, an efficient method for eNB selection will also be required for M2M communication. To this end, we have presented a reinforcement-learning-based eNB selection algorithm and some preliminary performance evaluation results for this algorithm.

ACKNOWLEDGMENT

This work was supported in part by the University of Manitoba Research Grant Program (URGP) and in part by the Natural Sciences and Engineering Research Council of Canada (NSERC).

REFERENCES

[1] 3GPP TS 22.368 V11.5.0, “Service Requirements for Machine-Type Communications,” June 2012.
[2] 3GPP TS 36.300 V11.2.0, “Evolved Universal Terrestrial Radio Access (E-UTRA) and Evolved Universal Terrestrial Radio Access Network (EUTRAN), Overall Description,” June 2012.
[3] K. Zheng *et al.*, “Radio Resource Allocation in LTE-Advanced Cellular Networks with M2M Communications,” *IEEE Commun. Mag.*, vol. 50, no. 7, July 2012, pp. 184–92.
[4] A. Gotsis, A. Lioumpas, and A. Alexiou, “M2M Scheduling over LTE: Challenges and New Perspectives,” *IEEE Vehic. Tech. Mag.*, vol. 7, no. 3, Sept. 2012, pp. 34–39.
[5] S. Sesia, I. Toufik, and M. Baker, *LTE, The UMTS Long Term Evolution: From Theory to Practice*, Ch. 19, Wiley, 2009.
[6] 3GPP TS 36.321 V10.5.0, “Medium Access Control (MAC) Protocol Specification,” Mar. 2012.
[7] 3GPP TR 37.868 V11.0.0, “Study on RAN Improvements for Machine Type Communications,” September 2011.
[8] S.-Y. Lien *et al.*, “Cooperative Access Class Barring for Machine-to-Machine Communications,” *IEEE Trans. Wireless Commun.*, vol. 11, no. 1, Jan. 2012, pp. 27–32.
[9] J.-P. Cheng, C. han Lee, and T.-M. Lin, “Prioritized Random Access with Dynamic Access Barring for RAN overload in 3GPP LTE-A Networks,” *Proc. 2011 IEEE GLOBECOM Wksps.*, Dec. 2011, pp. 368–72.
[10] K.-D. Lee, S. Kim, and B. Yi, “Throughput Comparison of Random Access Methods for M2M Service over LTE Networks,” *Proc. 2011IEEE GLOBECOM Wksps.*, Dec. 2011, pp. 373–77.
[11] S. Choi *et al.* “Automatic Configuration of Random Access Channel Parameters in LTE Systems,” *Proc. 2011 IFIP Wireless Days*, Oct. 2011, pp. 1–6.
[12] A. Lo, Y. W. Law, and M. Jacobsson, “Enhanced LTE-Advanced Random-Access Mechanism for Massive Machine-to-Machine (M2M) Communications,” *Proc. 27th Meeting of Wireless World Research Forum*, Oct. 2011.
[13] S.-Y. Lien, K.-C. Chen, and Y. Lin, “Toward Ubiquitous Massive Accesses in 3GPP Machine-to-Machine Communications,” *IEEE Commun. Mag.*, vol. 49, no. 4, Apr. 2011, pp. 66–74.

[14] C.-Y. Tu, C.-Y. Ho, and C.-Y. Huang, "Energy-Efficient Algorithms and Evaluations for Massive Access Management in Cellular-Based Machine to Machine Communications," *Proc. 2011 IEEE VTC-Fall*, Sept. 2011, pp. 1–5.

BIOGRAPHIES

MONOWAR HASAN received his B.Sc. degree in computer science and engineering from Bangladesh University of Engineering and Technology (BUET), Dhaka, in 2012. He is currently working toward his M.Sc. degree in the Department of Electrical and Computer Engineering at the University of Manitoba, Winnipeg, Canada. His research interests include cognitive radio networks, machine-to-machine communications, and resource allocation in mobile cloud computing.

EKRAM HOSSAIN is a professor in the Department of Electrical and Computer Engineering at the University of Manitoba. He received his Ph.D. in electrical engineering from the University of Victoria, Canada, in 2001. His current research interests include design, analysis, and optimization of wireless/mobile communications networks, cognitive radio systems, and network economics. He has authored/edited several books in these areas (http://www.ee.umanitoba.ca/~ekram). He serves as the Editor-in-Chief of *IEEE Communications Surveys and Tutorials* (for the 2012–2013 term), an Editor of *IEEE Journal on Selected Areas in Communications* (Cognitive Radio Series), and *IEEE Wireless Communications*. Also, he serves on the IEEE Press Editorial Board (for the term 2013–2015). Previously, he served as Area Editor for *IEEE Transactions on Wireless Communications* on Resource Management and Multiple Access from 2009 to 2011 and an Editor of *IEEE Transactions on Mobile Computing* from 2007 to 2012. He has won several research awards including the University of Manitoba Merit Award in 2010 (for Research and Scholarly Activities), the 2011 IEEE Communications Society Fred Ellersick Prize Paper Award, and the IEEE Wireless Communications and Networking Conference 2012 Best Paper Award. He is a Distinguished Lecturer of the IEEE Communications Society for the term 2012–2013. He is a registered Professional Engineer in the province of Manitoba, Canada.

DUSIT NIYATO is currently an assistant professor in the School of Computer Engineering at the Nanyang Technological University, Singapore. He obtained his B.Eng. in computer engineering from King Mongkut's Institute of Technology Ladkrabang, Bangkok, Thailand. He received his Ph.D. in electrical and computer engineering from the University of Manitoba. His research interests include machine-to-machine communications, device-to-device communications, and heterogeneous networks (HetNets).