社団法人　電子情報通信学会
THE INSTITUTE OF ELECTRONICS,
INFORMATION AND COMMUNICATION ENGINEERS

信学技報
TECHNICAL REPORT OF IEICE.

# 多数のM2M/IoT端末からの集中アクセスを考慮したモバイルコアネットワークの実験評価

上野　真生†　長谷川　剛††　村田　正幸†

† 大阪大学大学院情報科学研究科　〒560–0871 大阪府吹田市山田丘 1–5
†† 大阪大学サイバーメディアセンター　〒560–0043 大阪府豊中市待兼山町 1–32
E-mail: †{m-ueno,murata}@ist.osaka-u.ac.jp, ††hasegawa@cmc.osaka-u.ac.jp

**あらまし**　セルラネットワークにおけるトラヒックの増加，また，M2M/IoT サービスの広がりによって，LTE や第 5 世代移動通信システムなどのセルラネットワークにおける輻輳への対応が課題となっている．多くの既存研究では SDN や NFV といった仮想化技術をモバイルコアネットワークに適用することで，セルラネットワークの性能の向上を図っている．しかしながら，それらの手法の性能評価は主に数学的な解析やシミュレーションに基づいているため，それらの手法の性能を正しく評価するためには，実際のネットワーク環境を用いた実験的評価が必要である．本報告では，モバイルコアネットワークを構築するための実ソフトウェアを仮想化プラットフォーム上に展開し，端末が通信を行う際に必要となるシグナリング処理の処理遅延時間を評価した．モバイルコアネットワークの構築には OpenAirInterface を用い，更に多数のユーザ端末をクラウドコンピューティングプラットフォーム上に展開することで，複数の端末がコアネットワークに接続する環境を構築した．評価の結果，1 コア 1 GHz の CPU を持つ仮想マシンに MME の機能を持たせた場合，128 台のユーザ端末からの同時アクセスによって，ベアラ確立時間が最大で 450 % 増加することがわかった．

**キーワード**　モバイルコアネットワーク, M2M/IoT 通信, Long Term Evolution (LTE), シグナリング処理, virtualized Evolved Packet Core (vEPC)

## 1. Introduction

Handling congestion in cellular networks including Long Term Evolution (LTE) [1] and 5th generation mobile communication system [2] has become a critical issue due to recent and rapid increase in users of mobile terminals and their functional enhancement. In addition, the concept of attaching M2M/IoT terminals to cellular networks has attracted a lot of attention. Because of this trend, a more urgent problem is the increasing traffic load on mobile core networks, especially on the control plane.

There are some types of M2M/IoT terminals which have different communication characteristics from traditional rich user terminals: an enormous number of terminals perform periodical communication with a small amount of data packets. There are concerns about congestion due to M2M/IoT communication characteristics when accommodating them into cellular networks. For this reason, standardization organizations, including the 3rd Generation Partnership Project (3GPP) publicizes several cellular-based access technologies of accommodating M2M/IoT communication such as Narrow Band IoT (NB-IoT) [3] and enhanced Machine Type Communication (eMTC) [4]. Moreover, various existing works [5–8] have argued that virtualization technologies such as Software Defined Network (SDN) and Network Function Virtualization (NFV) are possible solutions for improving network capacity of mobile core networks.

Our research group focused on mobile core network architecture for accommodating M2M/IoT terminals [9, 10]. In [10], we conducted mathematical evaluations of models of mobile core network architecture considering the bursty access from massive M2M/IoT terminals, and we clarified the effect of server virtualization, optimal resource allocation, and C/U plane separation. In the evaluation, the processing loads of signaling messages were determined by a simple queueing model and the number of statements obtained by OpenAirInterface [11], an implementation of LTE and an Evolved Packet Core (EPC) network written in C language. However, the actual signaling processing load does not always correlate with the number of statements of implementation codes, because the actual signaling processing is performed by execution codes generated after compiling implementation codes, and counting the number of statements does not take the behavior of conditional branches into account. Therefore, observing the signaling processing load on the real system is required, in order to examine the C/U plane separation and resource allocation of mobile core network nodes.

In this report, we show the experimental evaluation results of the performance of a mobile core network to assess the impact of massive accesses from M2M/IoT terminals. First, we construct the network system for experimental evaluations, based on open-source implementation of mobile core networks and emulated user terminals and radio base stations. We also build up massive number of emulated user terminals and radio base stations on the public cloud computing platform. Then, we conduct experiments of simultaneous access from at most 128 user terminals. We evaluate the processing delay of the attach procedure at each mobile core node, and discuss how the attach requests from multiple user terminals affects the performance of the mobile core network nodes.

## 2. Mobile Core Network

### 2.1 Network Model

Figure 1 depicts the model of a mobile core network that includes EPC nodes, interfaces between nodes, and bearers

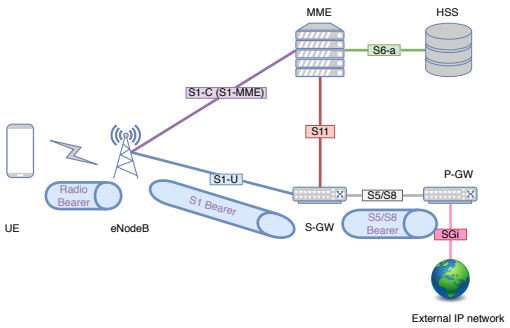



Figure 1 Mobile core network model

established when UE starts data transmission. The nodes have the following functions.

- **User Equipment (UE):** User terminals, including smartphones, tablets, and M2M/IoT terminals.
- **evolved Node Base (eNodeB):** Radio base stations that exchange control messages and data packets with UEs through radio channels. eNodeBs also exchange data packets with the S-GW and control messages with the MME.
- **Mobility Management Entity (MME):** The node that performs the core of the signaling processing, such as authentication of UEs, handling UEs' handover in wireless networks, and bearer setting for data-plane packet transmission between UEs and external IP networks.
- **Home Subscriber Server (HSS):** The database node that manages user specific information, such as the contract information of each user, data for authentication, and the location data of each UE.
- **Serving Gateway (S-GW):** The node that relays IP packets between UEs and the P-GW according to control from MME. It also performs as an anchor point when UEs move between eNodeBs.
- **Packet Data Network Gateway (P-GW):** The node that exchanges IP packets with external IP networks.

Each node is connected by the following logical interfaces built on an IP network.

- **S1-C (S1-MME):** The control plane interface that connects the eNodeB and the MME to exchange control messages between UEs and the MME through the eNodeB.
- **S1-U:** The data plane interface that connects the eNodeB and the S-GW to exchange IP packets between UEs and the S-GW through the eNodeB.
- **S6-a:** The control plane interface that connects the MME and the HSS to exchange control messages such as authentication data and location data.
- **S11:** The control plane interface that connects the MME and the S-GW to exchange control messages including bearer information of each UE.
- **S5/S8:** The data plane interface that connects the S-GW and the P-GW to exchange user data.
- **SGi:** The data plane interface that connects the P-GW and the external IP network to exchange IP packets between UEs and the external IP network.

### 2.2 Signaling Flow for Attach Procedure

When a UE connects to the mobile network, three data-plane bearers are established before starting data transmission: a radio bearer between the UE and the eNodeB, an S1 bearer between the eNodeB and the S-GW, and an S5/S8 bearer between the S-GW and the P-GW. Figure 2 shows the signaling flow to establish the bearers when a UE attaches to the mobile core network before data transmission. Several abbreviations are used in the figure; req., res. ans. and cmp. mean request, response, answer and complete, respectively. Ctxt stands for Context and Msg. stands for Message. Additionally, UE and eNodeB are depicted as a single node (UE+eNodeB), and we omitted signaling messages between

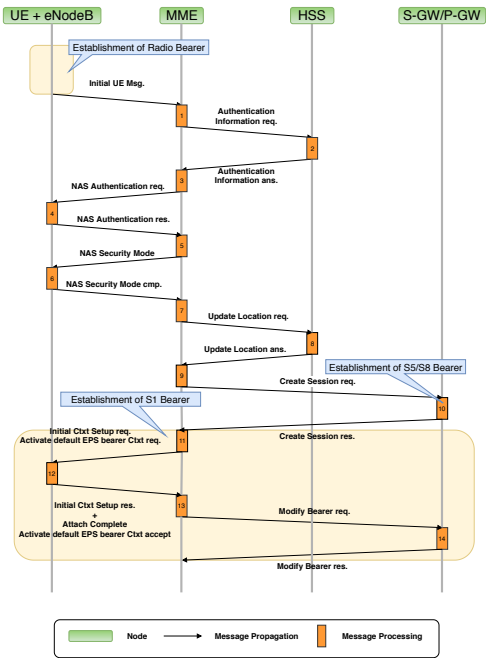



Figure 2 Signaling flow in attach procedure

UE and eNodeB since we did not evaluate them for this paper.

As shown in Figure 2, many control plane messages are exchanged between mobile core nodes before starting data transmission. Consequently, the load on control plane nodes becomes large when considering that massive M2M/IoT terminals are accommodated into the cellular network and their data transmissions are synchronized due to the application characteristics, even when transmitting small amounts of data per UE. This is because the signaling flow in Figure 2 is required in attach procedure regardless of the data size to be transmitted. Consequently, assessing the performance of the mobile core network against the access concentration is important.

## 3. Setup of Experiment

### 3.1 OpenAirInterface (OAI)

We exploited OAI, an open-source implementation of LTE/EPC networks, to construct an experimental environment including a mobile core network, UEs and eNodeBs. OAI includes components to operate UEs and eNodeBs on either actual equipment or in a simulator called OAISIM, and components to operate EPC nodes on servers. In OAI, an S-GW and a P-GW are implemented as a single node and the S5/S8 interface between the S-GW and the P-GW is realized by the interprocess communications. Therefore, S-GW and P-GW are referred to as "SP-GW" in what follows. Note that an eNodeB exists for each UE because of the current limitations of the OAISIM.

### 3.2 Network Configuration

In our experiment, OAISIM and EPC nodes are deployed on independent networks. We describe their detailed settings in the following subsections.

#### 3.2.1 EPC Network

Figure 3 depicts the configuration of EPC nodes in the experimental environment. The MME, the HSS, and the SP-GW were installed on separate virtual machines on a single physical host running VMWare ESXi 6.0 update 2. Table 1 shows the specifications of each node and virtualization environment

All LTE/EPC logical interfaces except for SGi belong to independent network segments from the Local Area Net-

Table 1 Specifications of EPC nodes and server virtualization environment

| Node Name | Operating System | Kernel Version | CPU Clock [GHz] | CPU Core | Memory Size [GB] |
|---|---|---|---|---|---|
| MME | Ubuntu 14.04 LTS | 3.13.0-24-generic | 1 | 1 | 1 |
| HSS | Ubuntu 14.04 LTS | 3.13.0-24-generic | 1 | 1 | 4 |
| SP-GW | Ubuntu 14.04 LTS | 4.7.5 | 1 | 1 | 4 |
| VM Host | VMWare ESXi 6.0u2 | build-3620759 | 2.40 | 12 | 16 |

Table 2 Specifications of Instances

| Instance type | vCPU | Memory [GiB] |
|---|---|---|
| `t2.micro` | 1 | 1 |
| `m5.large` | 2 | 8 |

work (LAN) of our laboratory in order to avoid impacting LAN traffic on the experimental network. In Figure 3, 192.168.3.0/22 represents the LAN of our laboratory, where the SGi interface was placed. 172.1.0.0/16 and 192.168.4.0/22 are independent network segments for the S1-C/S1-U and S6-a/S11 interfaces, respectively. As we describe in section 3. 2. 2, since OAISIM are executed on a public cloud computing platform, we have created globally accessible gateway to the EPC network. The gateway is enabled to handle Stream Control Transmission Protocol (SCTP) packets, which is utilized for S1 Application Protocol (S1AP) communications. The gateway forwards SCTP packets from the Internet to EPC nodes, and vice versa.

**3. 2. 2** OAISIM Network

We used Amazon Web Service (AWS) [12] Elastic Computing Cloud (EC2) as a public cloud computing platform for executing OAISIM. Figure 4 depicts the detailed configuration of OAISIM network. To reduce the propagation delay between OAISIM and our EPC nodes, we created our virtual private cloud (VPC) in Asia Pacific (Tokyo) region (identified as `ap-northeast-1`). We created three subnets in our VPC. One is a public subnet, which is able to allocate both global and private IP addresses to instances. The others are private subnets, which is only able to allocate private IP addresses to instances. Two private subnets are created on a separated availability zones (`apne1-az2` and `apne1-az4`) to avoid lack of resource on each availability zone. L3 connectivity among three subnets is ensured by the router in the VPC.

During experiments, a NAT instance is created on the public subnet during experiments and OAISIM instances are created on the private subnets. Packets sent from OAISIM instances to the Internet are routed as follows.

(1) Packets sent from OAISIM instances are relayed to the NAT instance by the router.
(2) The NAT instance applies IP masquerade to packets sent from private subnets, then send them to the router.
(3) The router forwards packets sent from the NAT instance to the Internet gateway.

We used `t2.micro` instance for the NAT instance and `m5.large` instance for OAISIM instances. The specifications of those instances are described in Table 2. Moreover, we measured Round Trip Time (RTT) from each private subnet to our LAN by `ping` command. Table 3 shows the average RTT of 100 ICMP packets sent from instances created on each private subnet to our globally accessible gateway. From these results, we can estimate that the one-way delay between OAISIM instances and our EPC network is roughly 5.95 [ms], because the propagation delay among the globally accessible gateway and EPC network is sufficiently small.

### 3. 3 Measurement Method

The assessment of experimental results was conducted based on packet capture data obtained by `tcpdump`. The packet capture was operated on the following two network interfaces, as depicted in Figure 3:

- **eth2 on the MME:** Monitors S1AP signaling packets passing through S1-C and S1-U interfaces.
- **eth0 on the Packet Capture Host:** Monitors Diameter signaling packets passing through an S6-a interface and GPRS Tunneling Protocol version 2 (GTPv2) signaling packets passing through an S11 interface.

The packet capture is activated just before an experiment begins and terminated just after the experiment finishes. Consequently, all signaling packets passing through S1-C, S1-U, S6-a, and S11 interfaces during the experiment are recorded in the packet capture data. Since each signaling packet contains identifiers of UEs, we can evaluate a detailed bearer establishment procedure of each UE.

To prevent time differences between the two capture points, the MME and the Packet Capture Host synchronize their clocks by Network Time Protocol (NTP). The MME runs as an NTP server, and an NTP client on the Packet Capture Host refers to the NTP server on the MME. NTP clients on the HSS and the SP-GW also refer to the NTP server on the MME to synchronize clocks of other nodes. NTP clients on OAISIM instances refer to Amazon Time Sync Service, which is accessible from AWS EC2 instances.

### 3. 4 Experiment Procedure

In our experiments, multiple UEs began the attach procedure over a short time duration, to assess the impact of massive accesses from UEs on the performance of the EPC and the bearer establishment procedure. For that purpose, the following steps were utilized for the experiment:

(1) Activate a NAT instance on the public subnet. Then, configure the router to forward packets from private subnets to the NAT instance.
(2) Activate required number of OAISIM instances on the private subnets. Each private subnet contains half of required number OAISIM instances.
(3) Notify the OAISIM instances of the time which is 120 [sec] after as synchronization time point. Additionally, a value of $T_{expect}$ [sec] is given to each instance.
(4) Send a signal to each OAISIM instance to execute an activation command of the eNodeB and UE.
(5) The eNodeB that is operated on each instance adjusts the timing when sending Initial UE Msg. by the following procedure.
  (a) $t_{adjust}$ is set to a random value between (0, $T_{expect}$) [sec].
  (b) The message transmission time point is calculated as the time point which is $t_{adjust}$ [sec] after the notified synchronization time point.
  (c) Sent Initial UE Msg. to the MME at the message transmission time point.

The above steps made it possible to send the concentrated attach request messages (Initial UE Msg. in Figure 2) from the UEs in the OAISIM instances to the MME. In addition, the concentration level of attach request messages could be configured by $T_{expect}$.

### 3. 5 Evaluation Method

The bearer establishment time for a certain UE is defined as the time difference between the time points when Initial UE Msg. that include an identifier of the UE arrives at the MME and the time point when Modify Bearer res. that in-

Table 3 Results of `ping` from AWS network to our laboratory LAN

| CIDR of Subnet | Availability zone | Avg. RTT [ms] |
|---|---|---|
| 172.16.1.0/24 | `apne1-az4` | 11.3 |
| 172.16.1.0/24 | `apne1-az2` | 12.5 |

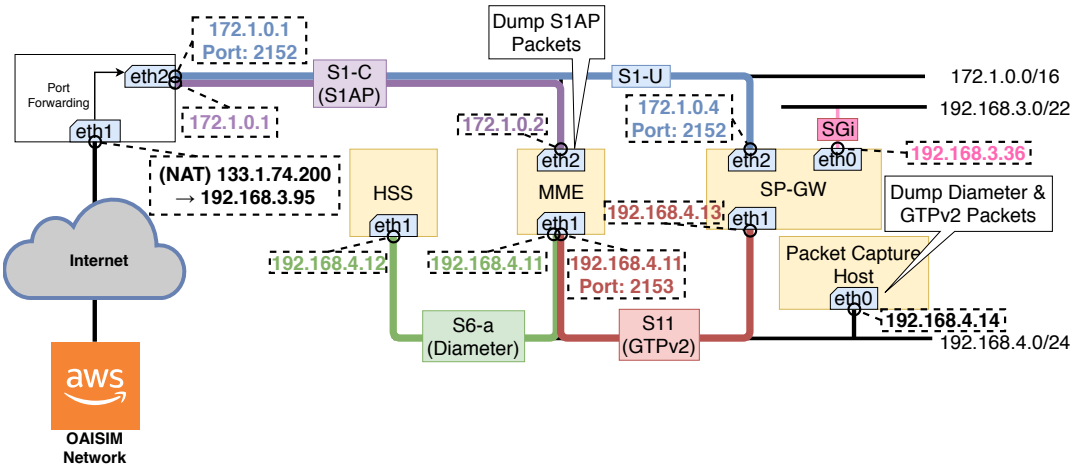



Figure 3 Configuration of EPC network

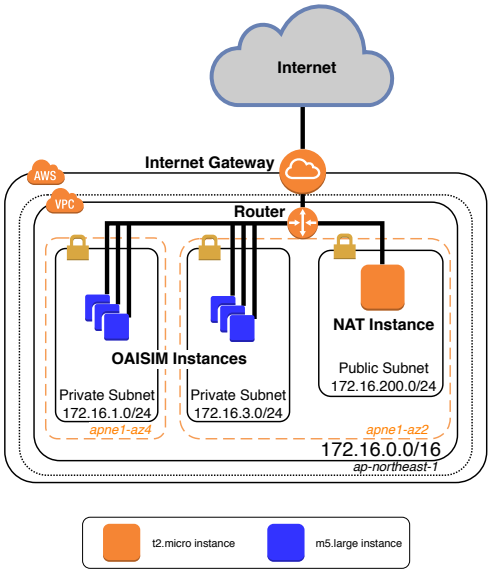



Figure 4 Configuration of OAISIM network

clude an identifier of the same UE arrive at the MME. The bearer establishment time includes the message propagation delays among the OAISIM instance and EPC nodes and the processing delay of messages at the MME, the HSS, the SP-GW, and the eNodeB and UE in the OAISIM instance. In addition, delays in processing each message (represented by orange square boxes in Figure 2) were evaluated. In detail, for each message processed, the processing delay was defined as a time difference between when the corresponding signaling message arrived at the node and when the corresponding message left the node after processing.

We conducted two types of experiments. First, we evaluated the impact of the number of UEs connected simultaneously on the mobile network. In this experiment, we utilized 1, 2, 4, 8, 16, 32, 64 and 128 OAISIM instances and $T_{expect}$ was set to 0 [sec]. Next, we evaluated how the concentration level affects the EPC performance. In this experiment, we utilized 128 OAISIM instances and $T_{expect}$ was set to 0.1, 0.2, 0.4, 0.8, 1.6, 3.2, 6.4 [sec]. We carried out ten times of experiment for each number of instances and the values of $T_{expect}$.

## 4. Evaluation Results

### 4.1 Bearer Establishment Time

Figure 5 shows the relationship between the number of OAISIM instances in logarithmic scale and the average bearer establishment time. Error bars laid on y axis explain the minimum and maximum bearer establishment time of ten experiments, where we set $T_{expect} = 0$ [sec] for all OAISIM instances. From these results, we can observe that the bearer establishment time slightly increased when the number of OAISIM instances increased from 1 to 64. However, when we set the number of OAISIM instances to 128, the bearer establishment time sharply increased; it increased by around 450% compared with the case when the number of OAISIM instances was 1. Figure 6 depicts the average message processing time at each EPC node as a function of the number of OAISIM instances. This graph presents the breakdown of bearer establishment time except for the message processing time at eNodeBs and UEs and one way delay between OAISIM network and the globally accessible gateway. Since eNodeBs and UEs were operated on OAISIM simulator, we omitted the processing time of eNodeBs and UEs in the latter evaluations as well. As shown in Figure 6, it is obvious that the increase in the bearer establishment time was mainly caused by the increase in the processing time at the MME. Consequently, we can conclude that the increase in the number of UEs significantly affected on the performance of the MME.

Figure 7 shows the relationship between $T_{expect}$ given to OAISIM instances represented by logarithmic scale and the average bearer establishment time, where 128 OAISIM instances are activated. Error bars laid on y axis explain the minimum and maximum bearer establishment time of each ten experiments. Note that the minimum bearer establishment time in case of $T_{expect} = 0.2$ [sec] is abnormally smaller than other results. This is because one of ten experiments with $T_{expect} = 0.2$ [sec] was conducted by only 99 OAISIM instances. We can obtain 2.8 [sec] as the minimum bearer establishment time when we eliminate the abnormal result. As shown in this figure, larger $T_{expect}$ reduced bearer establishment time even though the number of OAISIM instances is unchanged. When we set $T_{expect}$ to 6.4 [sec], the bearer establishment time was reduced by about 79 % compared with the case of $T_{expect} = 0$ [sec]. Figure 8 represents the relationship between given $T_{expect}$ and the message processing time on EPC nodes. In the graph we plot the average additional delay, caused by setting $T_{expect}$, on the top of each result. In this context, average additional delay represents the delay before each UE starts its data transmission regardless of the

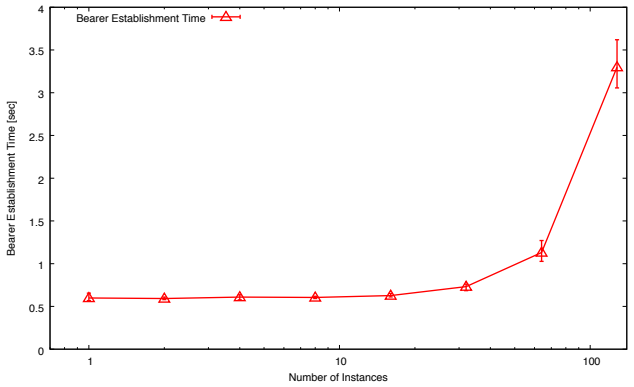



Figure 5 Relationship between the number of OAISIM instances and bearer establishment time

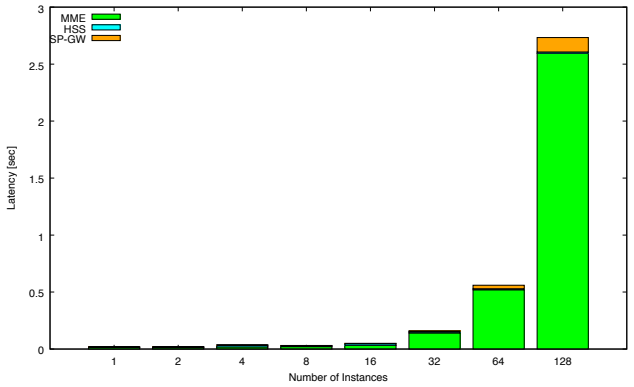



Figure 6 Relationship between the number of OAISIM instances and signaling processing time on each EPC node

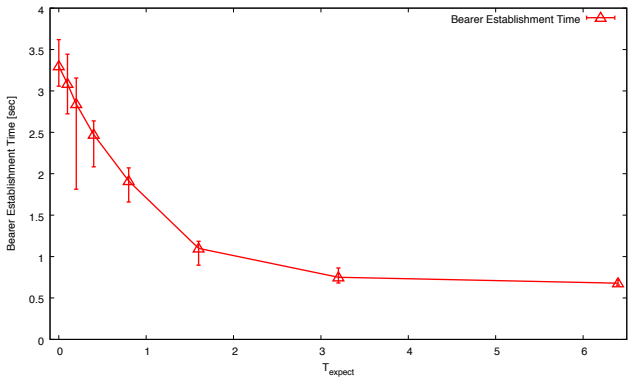



Figure 7 Relationship between $T_{expect}$ and bearer establishment time

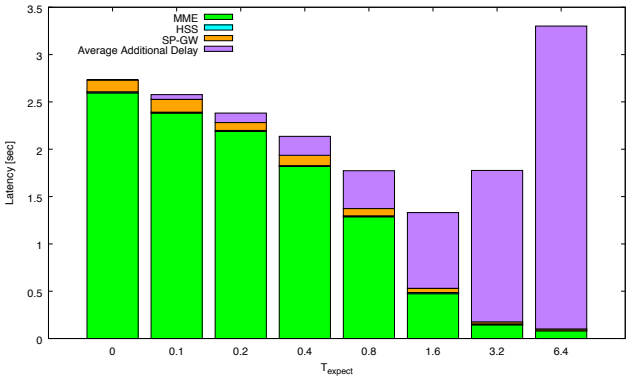
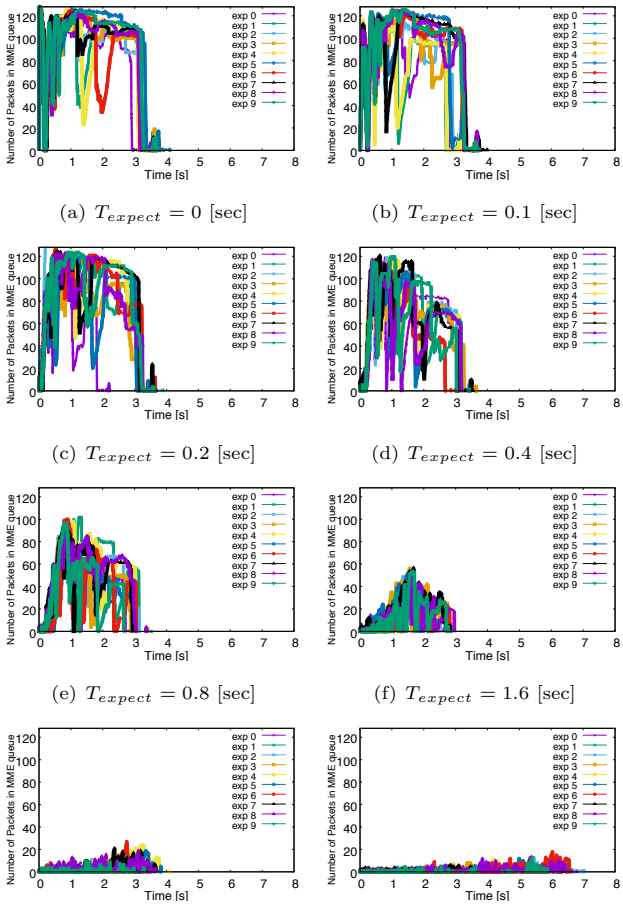



Figure 8 Relationship between $T_{expect}$ and signaling processing time on each EPC node




(a) $T_{expect} = 0$ [sec] (b) $T_{expect} = 0.1$ [sec]

(c) $T_{expect} = 0.2$ [sec] (d) $T_{expect} = 0.4$ [sec]

(e) $T_{expect} = 0.8$ [sec] (f) $T_{expect} = 1.6$ [sec]

(g) $T_{expect} = 3.2$ [sec] (h) $T_{expect} = 6.4$ [sec]

Figure 9 Queue length at the MME

actual load on the MME. It is calculated as a half of $T_{expect}$, because each eNodeB executed on OAISIM instances select $t_{adjust}$ between (0, $T_{expect}$) randomly. As shown in Figure 8, the message processing time at the MME significantly mitigated by increasing $T_{expect}$. However, large $T_{expect}$ causes substantial additional delay to each UE. We will discuss on this issue in detail in Section 4. 3.

### 4. 2 Queue Length at the MME

Figure 9 presents the temporal changes in the queue length at the MME, calculated as follows:

- When a signaling packet arrives at the MME at a certain time point, the queue length at that time point is incremented by one.
- When a signaling packet is sent from the MME at a certain time point, the queue length at that time point is decremented by one.

Note that the figures plot the results of all ten experiments.

As shown in the figures, the configuration of $T_{expect}$ apparently affected the queue length at the MME. The queue length kept large during experiments when $T_{expect}$ was set to 0, 0.1, 0.2, 0.4, 0.8 [sec]. This indicates that the packet processing speed at the MME is roughly the same as the packet arrival speed at the MME. Since the concentration level of attach request decreased when $T_{expect}$ became large, the queue length became less obviously.

### 4. 3 Discussion

As shown in Figure 6 and Figure 8, the main factor of the

inflation of the bearer establishment time was the increased signaling message processing time at the MME. Moreover, Figure 9 illustrates how the growth of the queue length at the MME strongly affected the processing time on the MME. Since this is the result with up to 128 UEs, the increased processing delay on the MME became a more critical issue when accommodating a larger number of UEs. When a particularly large number of M2M/IoT terminals simultaneously connect to the mobile network and start data transmission, the data transmission is delayed due to the concentration of attach requests, which increased the processing delay at the MME. Since some M2M/IoT terminals only transmit a small amount of data, the inflation of bearer establishment time becomes a substantial overhead on their communication.

The simplest way to deal with the above problems is the enhancement of computing resources for an MME. However, since most M2M/IoT terminals have an extremely low Average Revenue Per Unit (ARPU) compared to traditional terminals (for example, 2.20 USD per month [13]), it is difficult to recover the cost of reinforcing computing resources. Additionally, some M2M/IoT terminals communicate periodically; that is, not all of them always utilize the network. For this reason, the enhanced resource is temporally wasted. Consequently, static enhancement of computing resources is not always a desirable method for network operators in terms of OPEX and CAPEX. Therefore, methods such as server virtualization, optimal and adaptive resource allocation, and C/U plane separation with SDN technologies are required when accommodating M2M/IoT terminals.

One possible way to deal with the above problems is temporarily distributing attach requests from UEs intentionally. As shown in the results of our experiments in Figure 8, decreasing the concentration level of attach requests from UEs by using larger value of $T_{expect}$ significantly reduces the processing time on the MME. Therefore, increasing the value of $T_{expect}$ can decrease the load on the MME when accommodating massive M2M/IoT terminals to mobile core networks. However, it is obvious that introducing $T_{expect}$ brings additional delay in starting data transmission regardless of the actual load on the MME, as we described in Section 4. 1. Therefore, we should discuss whether or not the decrease of the bearer establishment time by introducing $T_{expect}$ can compensate the additional delay of the data transmission. From Figure 8, we observe that the processing time on the MME decreased by roughly 2.5 [sec] when $T_{expect}$ increased from 0 [sec] to 6.4 [sec]. However, $T_{expect} = 6.4$ [sec] causes 3.2 [sec] additional delay in average, thus the total latency of message processing became larger than the one of $T_{expect} = 0$ [sec]. In this case, introducing $T_{expect} = 6.4$ [sec] has a bad effect on the data transmission performance of UEs. On the other hand, when we set $T_{expect}$ to 1.6 [sec], the processing time on the MME is decreased by about 2.0 [sec]. The total amount of latency is reduced by 1.2 [sec] even when we considered 0.8 [sec] of average additional delay. In this case, introducing $T_{expect} = 1.6$ [sec] becomes more reasonable. We expect that the suitable value of $T_{expect}$ changes according to various factors such as the number of UEs to be attached and the amount of computing resources for EPC nodes.

## 5. Conclusion

In this report, we presented the experimental evaluation results of the performance of a mobile core network to assess the impact of massive accesses from M2M/IoT terminals. We established a simple experiment model based on open-source implementation of mobile core networks and emulated user terminals. Then, we evaluated the processing delay of the attach procedure at each mobile core node. We revealed that the simultaneous attach requests from 128 UEs increased the bearer establishment time on the MME by up to about 450%. We also found the relationship between the inflation of the processing time on the MME and queue length at the MME. Furthermore, we clarified that the optimal setting of $T_{expect}$ can improve the bearer establishment time including additional delay. For employing network slicing, which is considered in future 5G networks, the experimental results in this report can be applied to resource provisioning of EPC nodes according to UEs' communication characteristics in each slice.

In future work, we plan to investigate the effect of server resources (such as CPU speed and memory size) on the bearer establishment time to estimate the amount of resources required to accommodate massive M2M/IoT terminals. Furthermore, it is important to assess the effects of applying server virtualization and C/U plane separation with SDN to mobile core networks.

### References


[1] D. Astely, E. Dahlman, A. Furuskär, Y. Jading, M. Lindström, and S. Parkvall, "LTE: The Evolution of Mobile Broadband," *IEEE Communications Magazine*, vol. 47, pp. 44–51, April 2009.

[2] P. Marsch, I. D. Silva, O. Bulakci, M. Tesanovic, S. E. E. Ayoubi, T. Rosowski, A. Kaloxylos, and M. Boldi, "5G Radio Access Network Architecture: Design Guidelines and Key Considerations," *IEEE Communications Magazine*, vol. 54, pp. 24–32, November 2016.

[3] Third Generation Partnership Project, *Cellular System Support for Ultra-low Complexity and Low Throughput Internet of Things (CIoT)*, Nov. 2015. V13.1.0.

[4] Third Generation Partnership Project, *Study on provision of low-cost Machine-Type Communications (MTC) User Equipments (UEs) based on LTE*, June 2013. V12.0.0.

[5] A. Tawbeh, H. Safa, and A. R. Dhaini, "A Hybrid SDN/NFV Architecture for Future LTE Networks," in *Proceedings of 2017 IEEE International Conference on Communications (ICC)*, pp. 1–6, May 2017.

[6] A. Basta, W. Kellerer, M. Hoffmann, H. J. Morper, and K. Hoffmann, "Applying NFV and SDN to LTE Mobile Core Gateways; The Functions Placement Problem," in *Proceedings of the 4th Workshop on All Things Cellular: Operations, Applications, & Challenges*, pp. 33–38, ACM New York, NY, USA, Aug. 2014.

[7] Z. A. Qazi, V. Sekar, and S. R. Das, "A Framework to Quantify the Benefits of Network Functions Virtualization in Cellular Networks," *CoRR*, vol. abs/1406.5634, July 2014.

[8] F. Z. Yousaf, J. Lessmann, P. Loureiro, and S. Schmid, "SoftEPC — Dynamic Instantiation of Mobile Core Network Entities for Efficient Resource Utilization," in *Proceedings of 2013 IEEE International Conference on Communications (ICC)*, pp. 3602–3606, June 2013.

[9] G. Hasegawa and M. Murata, "Joint Bearer Aggregation and Control-Data Plane Separation in LTE EPC for Increasing M2M Communication Capacity," in *Proceedings of 2015 IEEE Global Communications Conference (GLOBECOM)*, pp. 1–6, Dec 2015.

[10] S. Abe, G. Hasegawa, and M. Murata, "Effects of C/U Plane Separation and Bearer Aggregation in Mobile Core Network," *IEEE Transactions on Network and Service Management*, vol. 15, pp. 611–624, June 2018.

[11] "OpenAirInterface." available at `http://www.openairinterface.org`.

[12] "Amazon Web Services (AWS) - Cloud Computing Services." available at `https://aws.amazon.com/`.

[13] S. Kechiche, "Cellular M2M forecasts and assumptions: 2010-2020," tech. rep., GSMA Intelligence, 2014.