RESEARCH   **Open Access**

CrossMark

Proposal and evaluation of SDN-based mobile packet core networks

Van-Giang Nguyen and Younghan Kim[*]

Abstract

Recently, several researchers have proposed adopting software-defined networking (SDN) concepts for mobile network architectures, particularly for LTE-evolved packet core (LTE/EPC). Although several new designs have been introduced for architectures based on SDN or its concrete realization, Openflow, few studies have presented in-depth discussions of real procedures that are relevant to such architectures. To this end, this paper first surveys the current approaches and solutions for adopting SDN/Openflow in LTE/EPC architectures and then introduces a new Openflow-enabled EPC (OEPC) architecture. This work provides detailed analyses of five procedures that commonly occur in LTE/EPC architectures, and the analyses are further elaborated with the separation of the control and data planes and the support of extended Openflow protocol. The analysis shows that the data management of these procedures is simpler relative to traditional LTE/EPC. In addition, in order to prove that efficient data management takes place for these procedures relative to a traditional LTE/EPC architecture, the numbers of signalling messages that are processed by control entities (i.e., the MME and the controller) are taken into account as a metric to evaluate the OEPC architecture. Moreover, the results of a numerical evaluation also show the benefits of this proposal relative to another Openflow-based LTE/EPC architecture.

Keywords: Mobile packet core; Software-defined mobile network; Openflow; Signaling load

1 Introduction

Modern mobile networks are experiencing excessive traffic loads due the sharp increase in smart devices that are connected (e.g., smart phone, IoT devices, etc.) as well as the introduction of new services and applications. The rate of growth in network use is a challenge for current mobile network architectures due to the increase in complexity for operation and management, high upgrade costs, and slow time-to-market for new innovations and services. The 3GPP LTE/EPC and LTE-A standards were released to cope with the increased demand for high-speed mobile networks. However, several issues related to the inherent design of these architectures necessitate radical changes. The 3GPP technical specification (TS 23.401) [1] indicates that the LTE/EPC architecture is composed of four main entities including an eNB, a mobility management entity (MME), a serving gateway (SGW), and a PDN gateway (PGW). Two additional entities related to subscriber management as well as policy and charging

management include a home subscriber server (HSS) and a policy and charging rule function (PCRF). The MME is the main control entity and is responsible for maintaining the mobility states of the UEs as well as setting up the bearer and forwarding path to carry the user traffic. The user data packets are then forwarded through the GTP tunnels (GPRS tunneling protocol) between the eNB and the PGW.

The LTE/EPC architecture faces problems in that, first, the control plane is still tightly coupled with the user or the data plane at the SGW and the PGW. Second, a change in the UE state between idle and connected states causes an exchange that requires many signalling messages between network entities as well as signalling messages that need to be processed by the MME. Third, data plane management is performed in a distributed manner which means that a forwarding plane needs to be established for all procedures that require a hierarchical exchange of a large number of signalling messages.

Software-defined networking (SDN) [2] is a new networking paradigm, which separates the control and data planes. In SDN, open interfaces (e.g., Openflow [3]) are

*Correspondence: younghak@ssu.ac.kr
Soongsil University, Dongjak-Gu, Seoul 156-743, South Korea

used to provide network programmability while the SDN controller itself contains control functions and uses Openflow or other protocols to control and configure simple forwarding devices in the data plane. As a result, network management tasks are simplified, new ideas and innovations can be deployed faster, and the sources of revenue for network operators increase. Openflow [3] is an actual implementation of SDN, and it is widely used as a southbound interface between the controller and the forwarding devices. Openflow was first introduced for use in campus networks and is now maintained and promoted by the Open Networking Foundation (ONF) [4].

The success of SDN/Openflow in wired network environments inspired the adoption of these two concepts for mobile cellular network environments. Researchers in both academia and industry have presented several proposals to use SDN/Openflow to re-design mobile network architectures, particularly LTE/EPC. With the feature of separation of control and data planes and programmability, SDN can ease the configuration and management and enables fast time-to-market for new services or applications in the mobile network. Several protocols were proposed to use as control interfaces between the control and data planes, but Openflow is the most dominant and strongly supported by research community and standardization groups. With this reason, Openflow is also considered as the protocol between the control and data planes in SDN-based mobile networks. However, some studies have been limited in that they present straightforward realizations of SDN/Openflow but lack a detailed analysis of the necessary procedures [5–13]. These studies have only proposed that SDN/Openflow can provide much-needed benefits but have not shown the details for the new architecture. The authors in [14, 15] proposed a partial approach for adopting SDN/Openflow in LTE/EPC architecture. In this architecture, the control and data planes are decoupled only at SGWs while these planes are still coupled to each other at PGW. These papers have performed analysis of the procedures and showed the reduction of signalling load compared to that in the traditional LTE/EPC architecture. However, the signalling load is still high.

In our previous study [16], we presented an alternative re-design of LTE/EPC network architecture by using SDN/Openflow technologies. We proposed OEPC (Openflow-enabled LTE/EPC), a new LTE/EPC architecture that is fully realized in Openflow. In this architecture, the control and data planes are completely separated, and the Openflow protocol substitutes the GTP-C protocol and is used for path management, tunnel management, mobility management, etc. This architecture takes advantages of SDN/Openflow technologies such as flexibility, programmability, fast time-to-market for innovations, and ease of configuration and management. For example, the

flexibility means the ability to adapt systems to new requirements such as applications or bandwidth. By fully separating the control and data plane of all entities in LTE/EPC, the signalling cost in any operational procedure of OEPC architecture is much lower than that of the traditional LTE/EPC architecture.

This paper is an extension of that work, and we mainly focus on analyzing the procedures necessary for the entire OEPC architecture and also try to show the actual operations for the proposed architecture. Five main procedures that commonly occur in the traditional LTE/EPC architecture are taken into consideration including the initial attachment, user-triggered service request, network-triggered service request, handover, and tracking area update. These procedures are described under OEPC, and we also show how these would be simpler than those of the traditional LTE/EPC architecture. Finally, the efficiency of our proposed architecture is demonstrated by using the signalling load as a metric to evaluate and compare the performance against that of the traditional LTE/EPC architecture as well as the reference architecture [14, 15]. A detailed signalling analysis is used to calculate the number of messages that are processed by the control entities (i.e., the MME and controllers) under different scenarios and under different UE states (i.e., Idle or Connected).

The remainder of this paper is organized as follows. In Section 2, we describe some related works that also leverage SDN/Openflow in modern mobile network architectures. The OEPC architecture is presented in Section 3. Section 4 provides a detailed analysis of the five main procedures that occur in LTE/EPC architectures, and Section 5 evaluates the performance of OEPC. Finally, we conclude the paper in Section 6.

2 Related works

In the last three years, several researchers have attempted to adapt SDN technology throughout mobile network architectures, from the radio access network to the mobile packet core network. Since the mobile packet core network is composed of wired entities (e.g., SGW, PGW, etc.), it seems that these would be easier to re-design by leveraging SDN technology rather than redesigning the radio access part. To this end, most studies have attempted to centralize control functions, including the control function of the MME as well as those of the SGW and PGW, as packages in an SDN controller or as VMs running in a cloud environment. The data plane for the EPC entities is simplified and can be thus programmed from the control plane by using open interfaces.

For example, MobileFlow [5] is a new architecture based on SDN that is designed for next-generation mobile carrier networks. In this architecture, the control plane and the data plane are separated into the MobileFlow controller (MFC) and MobileFlow forwarding engine (MFFE).

While MFC includes all control functions for the EPC entities and is responsible for managing and controlling the entire network, the MFFEs are simple, software-driven forwarding devices. SoftCell [6] is another high-level SDN design for mobile networks. SoftCell substitutes all legacy entities with Openflow switches and commodity middleboxes, and in this new mobile architecture, the routing mechanism utilizes tag-based routing instead of GTP tunneling. Similarly, the authors in [7] proposed a programmable mobile packet core architecture that replaces EPC entities with switches and routers. In contrast to SoftCell, the method in [7] still keeps the use of MME entity to handle control messages. Motoyoshi et al. [8], Raza et al. [9], Basta et al. [10], and Sama et al. [11] discussed several deployment scenarios that use SDN in future mobile networks or 5G networks. Yazici et al. [12] proposed programmable all-SDN 5G network architecture by deploying several SDN controllers from the RAN into the core network part. Each controller is in charge of controlling and managing each part of the mobile network. For example, device controllers are responsible for managing the device-to-device (D2D) communication. Yang et al. [13] proposed an end-to-end 5G architecture based on SDN technology. It is a cross-layer approach that combines the SoftRAN [17] and SoftCell [6] by using a service coordinator. It enables service-oriented feature in mobile network and efficiently guarantees the end-to-end QoS and quality of experience (QoE). The authors in [14, 15] presented an Openflow-based LTE/EPC architecture in which the Openflow protocol is used for the southbound interface with the SGW data planes (SGW-Ds).

The above studies introduced several architectures that implement SDN or Openflow in future mobile networks, particularly for LTE/EPC architectures. However, these studies have not discussed how these proposed architectures operate according to real procedures or how messages are exchanged between entities as well as between the control and data planes. Although the authors in [14, 15] considered the operational procedures, those procedures were still limited in that some commonly occurring procedures were not considered, including network-triggered requests or handovers. In addition, in that architecture only separated the control and data planes of the SGWs and kept a PGW that was the same as that of a traditional LTE/EPC architecture. We argue that such reason causes high signalling load due to the communication between the PGW and SGW-C function on Openflow controller.

In this paper, we aim at redesigning LTE/EPC architectures by fully realizing Openflow technology compared to partial approaches proposed in [14, 15]. The main contributions of this study are that we describe how messages are exchanged in a new Openflow-based architecture by analyzing five commonly occurring

procedures, including initial attachment, user-triggered service request, network-triggered service request, handover, and tracking area update.

3 OEPC architecture
3.1 Architecture overview
The overall architecture is depicted in Fig. 1b. In this architecture, the mobile controller (MC) is the brain of the network and is responsible for establishing the user session and managing the forwarding elements. All LTE/EPC control functions (MME, SGW-C, and PGW-C) are realized as an application running on top of the MC. The user plane consists of SGW-Us and PGW-U that act as extended Openflow switches capable of processing GTP packets. All radio access functions for the eNBs are kept the same as those of a traditional architecture. The MC calculates and installs rules into the user forwarding elements, including eNB, SGW-U, and PGW-U, according to service policies of the Openflow protocol. The functionality for each entity in the OEPC architecture is the following.
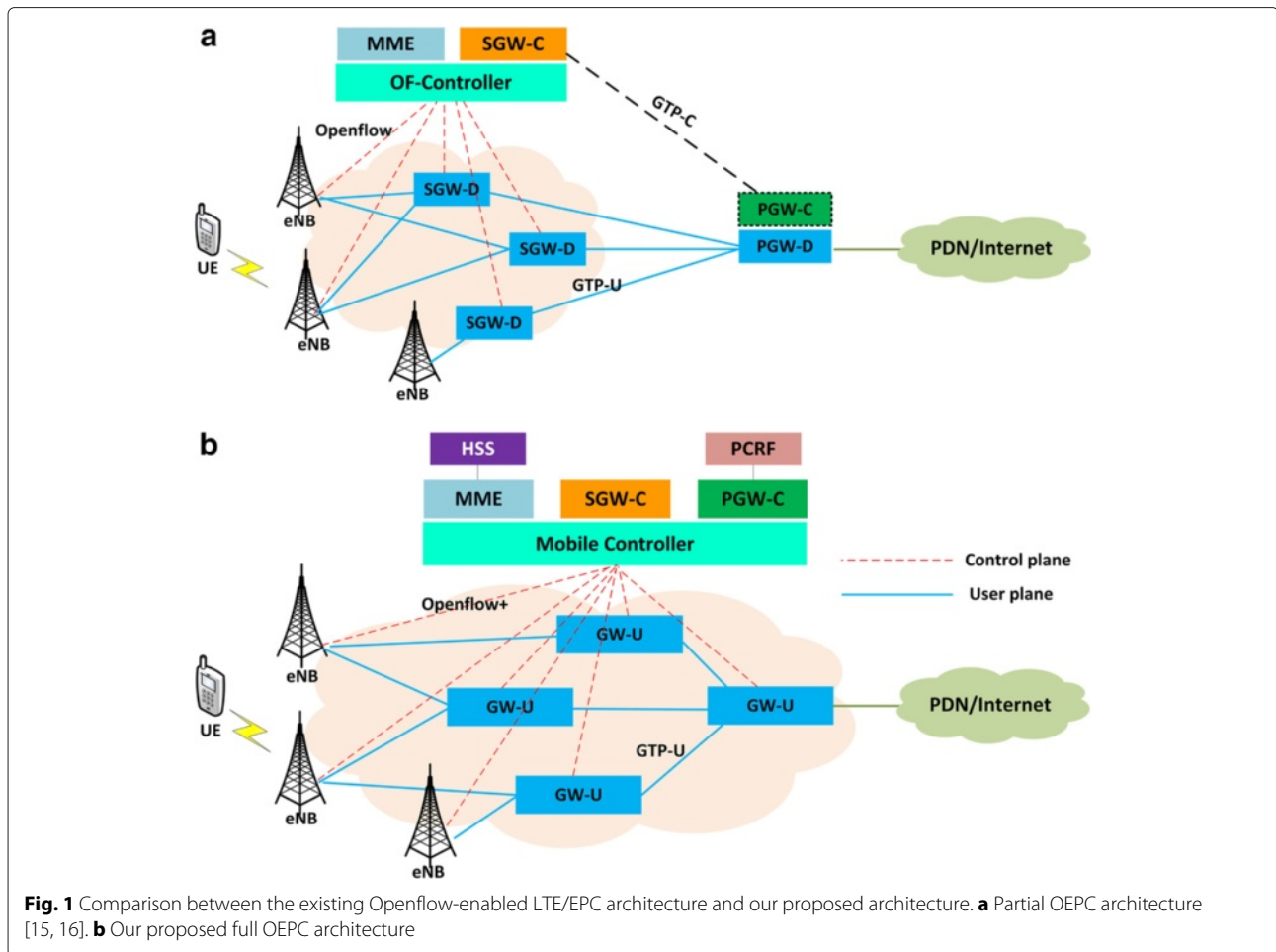
Mobile controller (MC) is the brain of OEPC and is in charge of establishing user sessions, installing the forwarding table of the GW-Us, and monitoring the network.

MME is an application on the MC and is responsible for mobility management and UE authentication. It communicates with the MC by using the REST (REpresentational State Transfer) API.

SGW-C and PGW-C represent the control functions of the SGW and PGW, respectively. They are responsible for allocating the tunnel endpoint identifiers (TEIDs) when establishing GTP tunnels, UE IP allocation, and session establishment. These functions, together with the MME, are virtualized and packaged as applications on top of the MC. In fact, SGW-C and PGW-C can be combined as a single application instead of running separately. However, for easy software development, these two gateway control functions are developed as two independent modules on the mobile controller instead of only one module which has the functionality of both.

Gateway user plane, or *GW-Us*, acts as an Openflow switch with some extensions for processing GTP packets. Even though these devices are unified, there are still some distinguishing characteristics among them. For example, the GW-U that is located between the OEPC and the PDN network or the Internet will function as a GTP termination point in order to remove the GTP header from the packets towards the Internet and to add the GTP header for packets coming to the OEPC.

The *Openflow protocol* is the Openflow protocol version 1.4 [18] and has some extensions for GTP header awareness. As specified in [18], a new match field named OXM_OF_TUNNEL_ID or Tunnel ID metadata (with 64 bits of length) was introduced for supporting tunnel

Fig. 1 Comparison between the existing Openflow-enabled LTE/EPC architecture and our proposed architecture. **a** Partial OEPC architecture [15, 16]. **b** Our proposed full OEPC architecture

encapsulation and decapsulation (e.g. VxLAN, MPLS, GRE) over logical ports. Actions corresponding to this match field were defined in Set_Field Actions structure [18]. However, the specification has not mentioned about GTP support and the introduction of Tunnel ID metadata as well as its corresponding actions was infancy. We are going to manipulate and extend this field as well as design set of corresponding actions for support GTP tunnel. An example of Openflow entry used in OEPC architecture is depicted in Fig. 2. One of our ongoing extensions is marked in green boxes. It should be noted that this figure shows a semantic design of Openflow extension. The detail design is elaborated in our other work, which is going to be contributed to ONF standardization group [3].

eNBs are enhanced with programmability. On the other hand, they are Openflow-based eNB and under the instruction of MC. The radio functions are kept the same as described in 3GPP technical specification.

3.2 Traffic flow in OEPC architecture
Figure 3 shows an example of the flow of the user plane traffic in the OEPC architecture that accesses the Internet.

Figure 3a shows the flow for the uplink traffic from the UE to the Internet, and Fig. 3b shows the flow for the downlink traffic from the Internet to a UE. The IP packets are forwarded through the GTP tunnel from the eNB to the GW-U (PGW). These GTP tunnels are established when the user performs a UE-triggered service request and a network-triggered service request, which are detailed in the next section. In the context of Openflow, each entity in the OEPC architecture has its own flow table with match and action fields. Whenever a packet arrives, these entities check whether or not a flow entry corresponding to that packet exists in their flow tables. In order to make the GTP tunnel over interfaces from the eNB to the GW-U (PGW), some extensions of Openflow protocol are needed. In this context, the action field not only includes some original actions like output, drop, etc. but also includes some required actions for GTP encapsulation and decapsulation. For example, for the uplink traffic flow, the GTP information is added to the packets, like TEID and the IP addresses of the eNB and GW-U (SGW), at the eNB side, and then the GTP information will be removed at the GW-U (PGW) side. Since the tunnel between eNB and

Fig. 2 An example of Openflow entry used in OEPC architecture
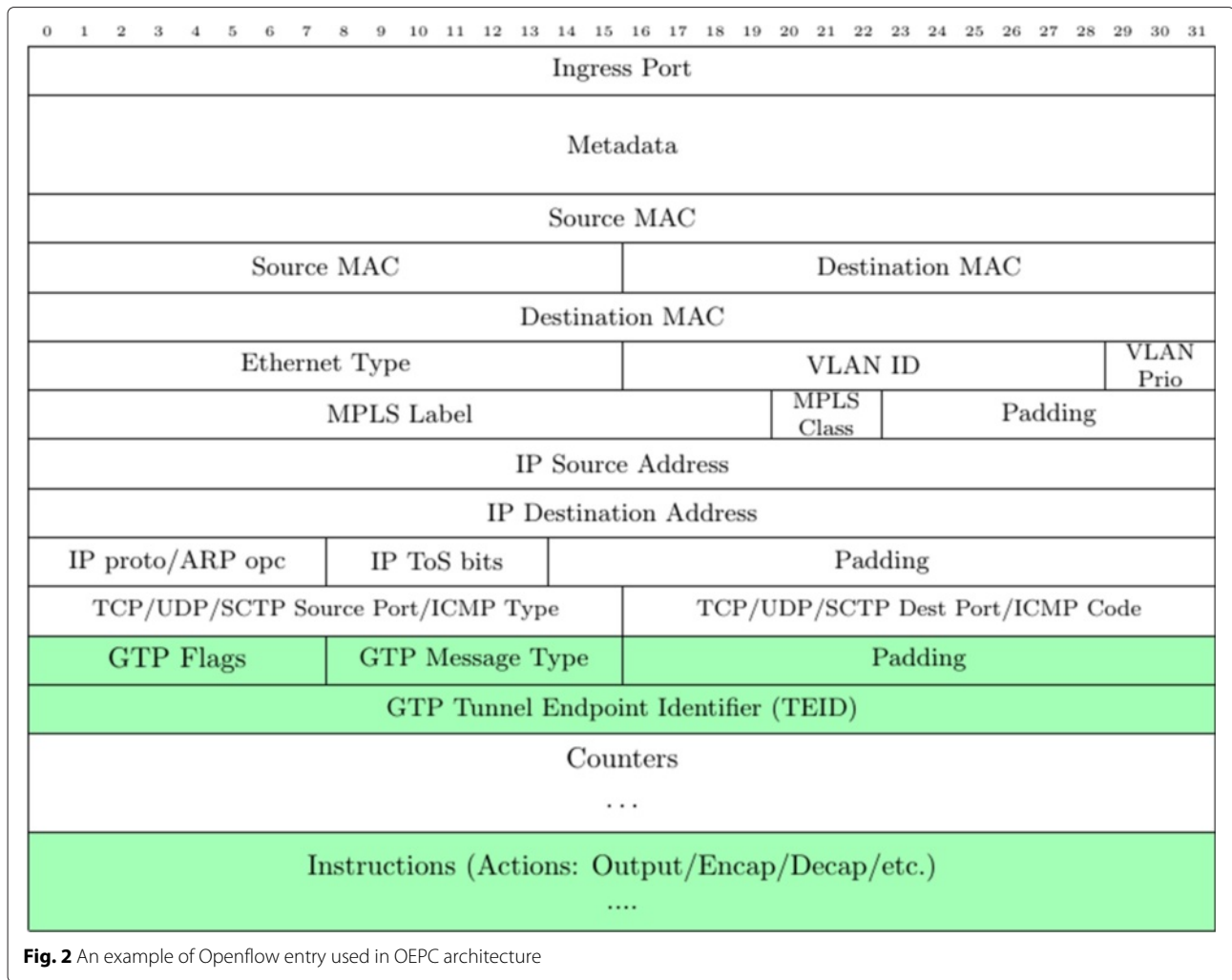
GW-U (SGW) is different from the tunnel between GW-U (SGW) and GW-U (PGW), we manipulate the concept of the rewrite action, as described in the original Openflow specification, in order to rewrite the TEID value at the GW-U (SGW). These TEID values that are used to establish the GTP tunnel are centrally assigned from the mobile controller instead of locally assigned as in the traditional LTE/EPC architecture [1]. The downlink traffic flow can also be easily observed.

4 OEPC procedures
For LTE/EPC, a UE has several states according to the state of the EPS (Evolved Packet System) Mobility Management (EMM) and the state of the EPS Connection Management (ECM). These states can be EMM-Deregistered/Registered or ECM-Idle/Connected, as specified in [1]. Two first states of UE indicate whether the UE is registered and authorized to the network or not. When the UE is already registered to the network but it does not use any service so it is called ECM-Idle

state. In contrast, the UE is in ECM-Connected state when it is registered to the network and is using services (e.g. Internet, Video, etc.). In short, we call these states Deregistered/Registered and Idle/Connected, respectively. We next describe in detail the five typical procedures and also show the transition of the UE state among these states described above.

4.1 Initial attachment
The initial attachment is the first step for the UE to register to the network after it is switched on. This initial attachment is slightly different from that described in [14, 15]. In our proposed architecture, the initial attachment is only used to register the UE information and authorize this UE without creating any session between SGW and PGW so that the message exchanging between the mobile controller and PGW for such session establishment is reduced. During this procedure, the IP address is allocated to the UE from PGW-C application through the mobile controller for future data transfers. The call

Fig. 3 Traffic flow in OEPC architecture. **a** Uplink traffic from UE to the Internet. **b** Downlink traffic from the Internet to UE
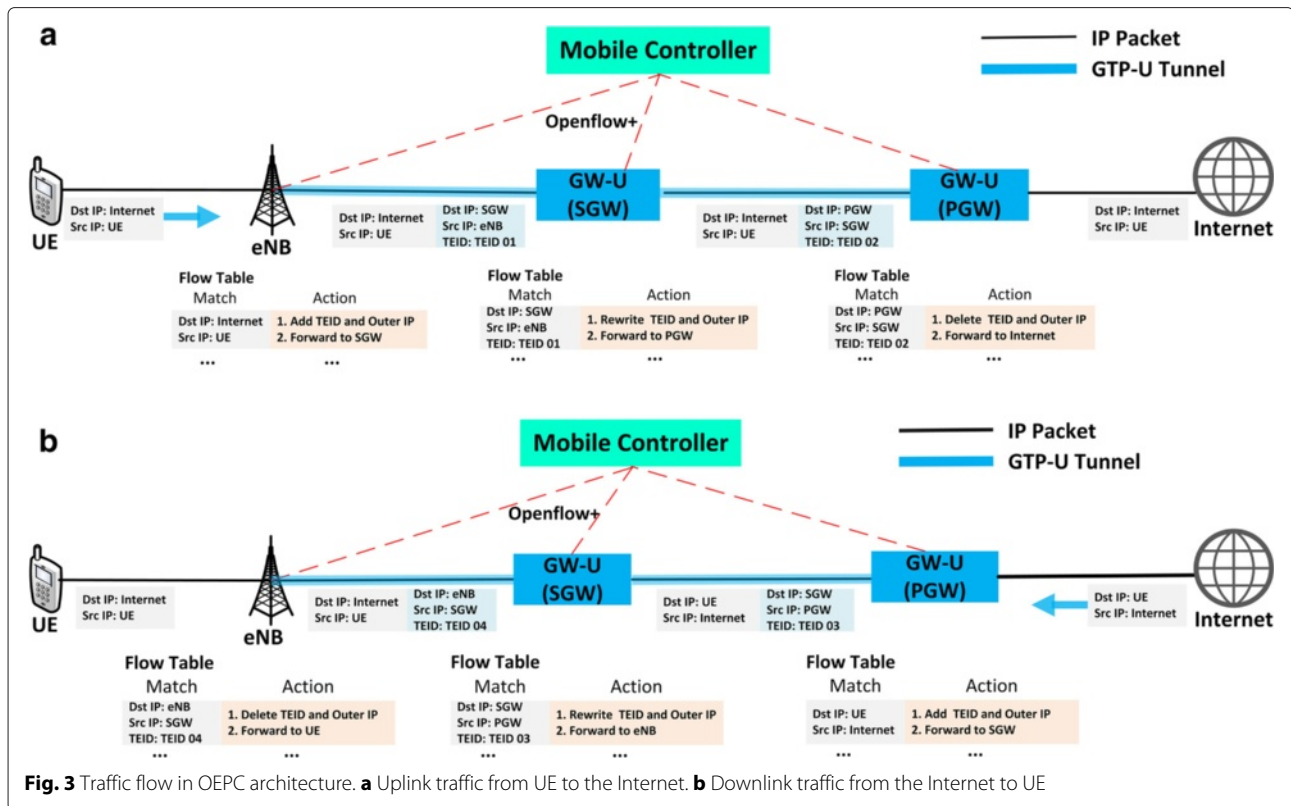
flow for the initial attachment is detailed in Fig. 4. First, the UE sends an Attachment Request message, including its IMSI, to the eNB. This message is embedded in an Openflow Initial UE message and is sent to the mobile controller (MC). The Tracking Area Identity (TAI) and the E-UTRAN Cell Global Identifier (ECGI) values are also included in the Initial UE message. The MC forwards this information to the MME function and then triggers the procedure to authenticate and authorize the UE. After the authentication and authorization steps, an Attach Accept message is embedded in the Openflow Initial Context Setup Request message that is sent from the MC to the eNB. The eNB reconfigures the radio connection, and the Attach Accept message is forwarded to the UE. To end the initial attachment procedure, the UE replies to the MME by sending an Attach Complete message over the eNB. At the end of this procedure, the state of the UEs transitions from Deregistered to Registered and the IP address is allocated to the UE.

4.2 UE-triggered service request

This procedure occurs when the UE that is in an idle state wants to use a service from the Internet or the PDN networks. All of the messages exchanged in Fig. 5 are used to setup the data forwarding path between the UE and its destination in the Internet. In contrast to traditional LTE/EPC where the data plane (bearer) between the SGW

and PGW (S5 bearer) is "always on", the bearer in the OEPC architecture is established on-demand. In Fig. 5, the UE triggers a Service Request message and sends it to the eNB. This message is embedded into an Openflow Initial UE message and is forwarded to the MC. If the UE passes the authentication check, the MME function prepares the resources and initiates an Openflow Initial Context Setup Request message for the eNB. After the eNB reconfigures the radio connection, it replies to the MC by sending an Openflow Initial Context Setup Response message. The UE transmits the first session packet to the eNB over the radio connection. Since this is the first packet, there is no matching flow entry in the flow table of the eNB. The eNB triggers an Openflow Packet In message to the MC. This message includes some information that is necessary to establish the data plane, such as the eNB IP address, etc. The MC analyzes the packet header to obtain the session information, such as the IP source or IP destination, etc. Next, the MC queries the information of the GW-Us by interacting with the SGW-C and the PGW-C functions, such as the IP addresses. Then, the MC creates flow rules for subsequent packets that belong to the same section and installs them for the eNB and GW-Us. Some examples of actions in the flow rules would be to add/remove TEID values or to rewrite the TEID values. The operation of the data plane has already been presented in Fig. 3. In the case where the session requires a specific QoS policy,

**Fig. 4** Procedure for the initial attachment
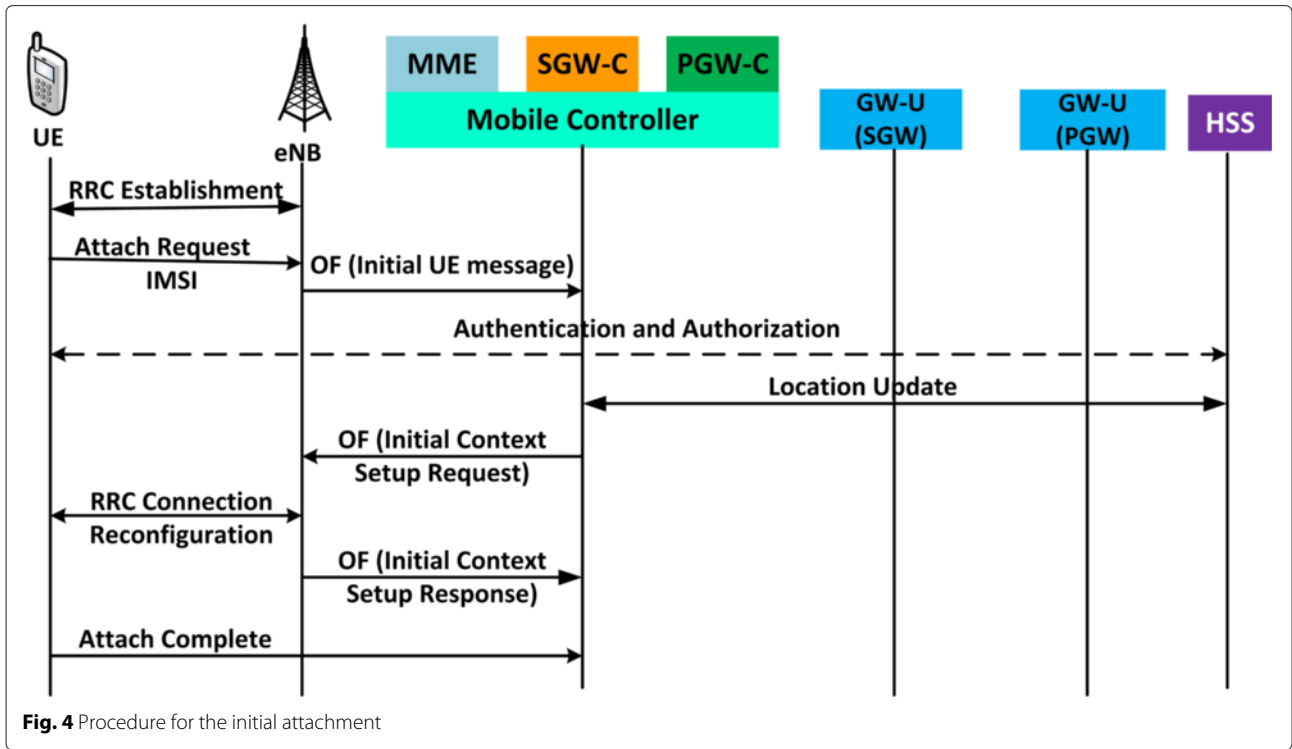
the MC needs to interact with the PCRF server through PGW-C function to get the corresponding policies before downloading the flow rules to the data plane. When the flow rules are associated with a QoS parameter, the MC will install them at the GW-Us. At the end of this procedure, the UE state is transitioned from Idle to Connected, and a data forwarding path is established from the UE to the Internet.

4.3 Network-triggered service request
The network-triggered service request is executed when the network has downlink traffic needed to deliver to a UE

Fig. 5 Procedure for UE-triggered service request
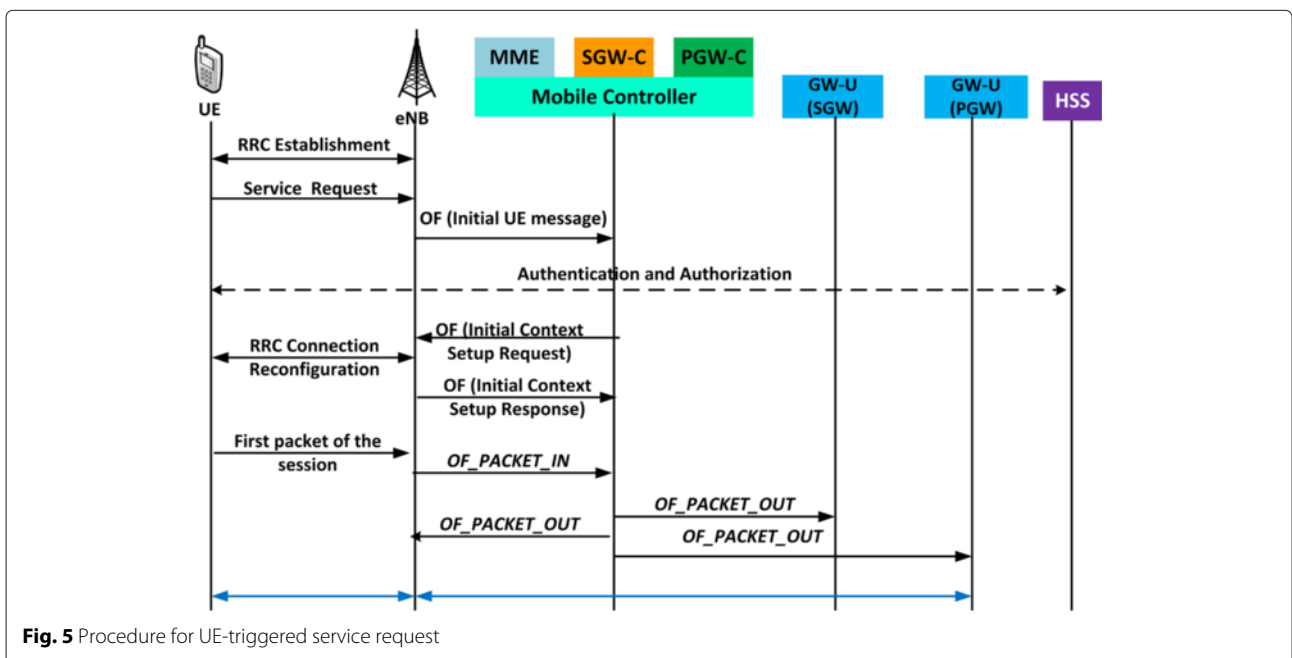
in the Idle state. The network now does not know the location of that Idle-state UE. Therefore, the network (MME, mobile controller) first has to perform the paging procedures to inform the new traffic toward the UE and then when UE initiates the UE-triggered service request procedure upon reception of paging indication as described in [1] and [19]. Figure 6 shows the call flow to establish a data forwarding path for the UE and its source when a new session or downlink traffic toward the UE originates from the Internet. When the GW-U receives a downlink packet for the UE, it matches the header to its flow table. Since this is a new packet for the UE, the GW-U will trigger an Openflow Packet In message to the MC as a notification. Next, the MC sends a paging message to each eNB belonging to the Tracking Area (TA) where the UE is registered. Then, the eNB performs a paging procedure to find the proper UE and activate it. The MC can send the paging message either as a unicast or as a multicast message, as described in [19]. When the paging request is received, the UE triggers a service request procedure. This procedure is the same as that described in Fig. 5. The data plane operation for this procedure is shown in Fig. 3. At the end of the network-triggered service request procedure, the state of the UE transitions from Idle to Connected, and the data forwarding path between the UE and its source in the Internet is established.

4.4 Handover

The handover is an important procedure for the LTE/EPC architecture. This procedure indicates how to re-establish a data forwarding path for the user data traffic to maintain data session continuity as the UE moves from one eNB to another. Depending on whether the X2 interface (the interface between the eNBs) is supported or not, the handover in the OEPC architecture is handled either as a handover with X2 support or a handover without X2 support. Depending on whether or not the two eNBs (old and new attachment points for the UE) are connected to the same GW-U, the procedures can be defined as either an intra-GW handover or an inter-GW handover.

In the following subsections, we describe all handover procedures in detail.

4.4.1 Handover with X2 support

If there is an X2 connection between the two eNBs between which the UE moves, the handover will be performed over this connection without the intervention of the MME. Figure 7 shows the procedure for an intra-GW handover with the support of an X2 connection. The procedure is similar to that of an X2 handover at the radio access network side of the traditional LTE/EPC architecture as described in [1]. The source eNB sends the Handover Request message to the target eNB, and this message is packaged in an Openflow message and is further forwarded to the MC. The MC replies with an Openflow Handover Response message to the target eNB, and the MC simultaneously notifies the GW-U that is connected to both eNBs and the target eNB to modify their flow tables by sending Openflow Packet Out messages. The largest difference in comparison to a traditional procedure is that the downlink (DL) data will now be forwarded directly to the target eNB instead of through an indirect tunnel through the source eNB. This data is buffered at the target eNB and waits for radio connection reconfiguration and synchronization. Finally, the buffered DL data is forwarded to the UE. Conversely, the uplink (UL) data will be sent to the target eNB. The resource to maintain the previous tunnel at the source eNB will expire due to the time-out value inside of the flow table. Thus, we do not need to perform any more actions to release the resources at the source eNB. In the case of an inter-GW handover or gateway relocation, the MC needs to send Openflow Packet Out messages to the target GW-U (SGW) and the GW-U (PGW) in order to modify their flow tables.

4.4.2 Handover without X2 support

If the X2 interface between the two eNBs is not supported, the handover of the UE is triggered by the MME function in the MC at the network side. Figure 8 shows the
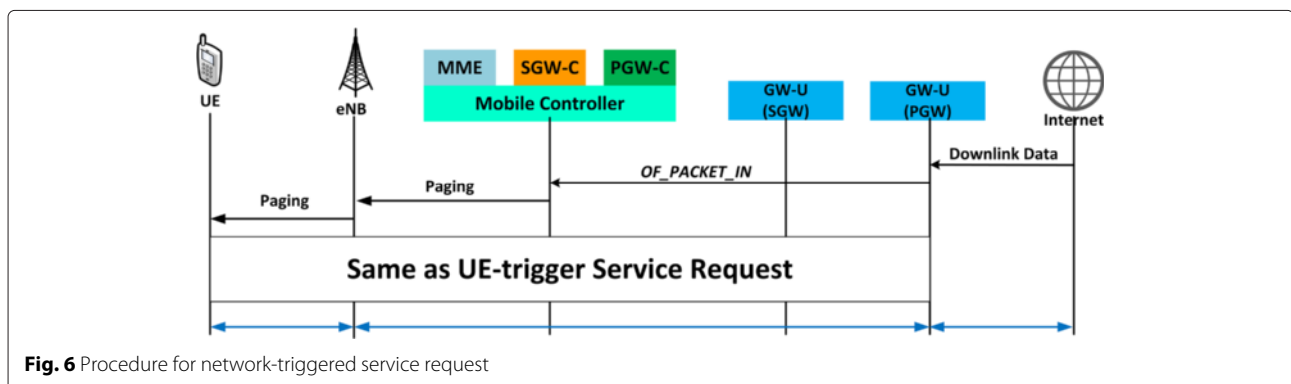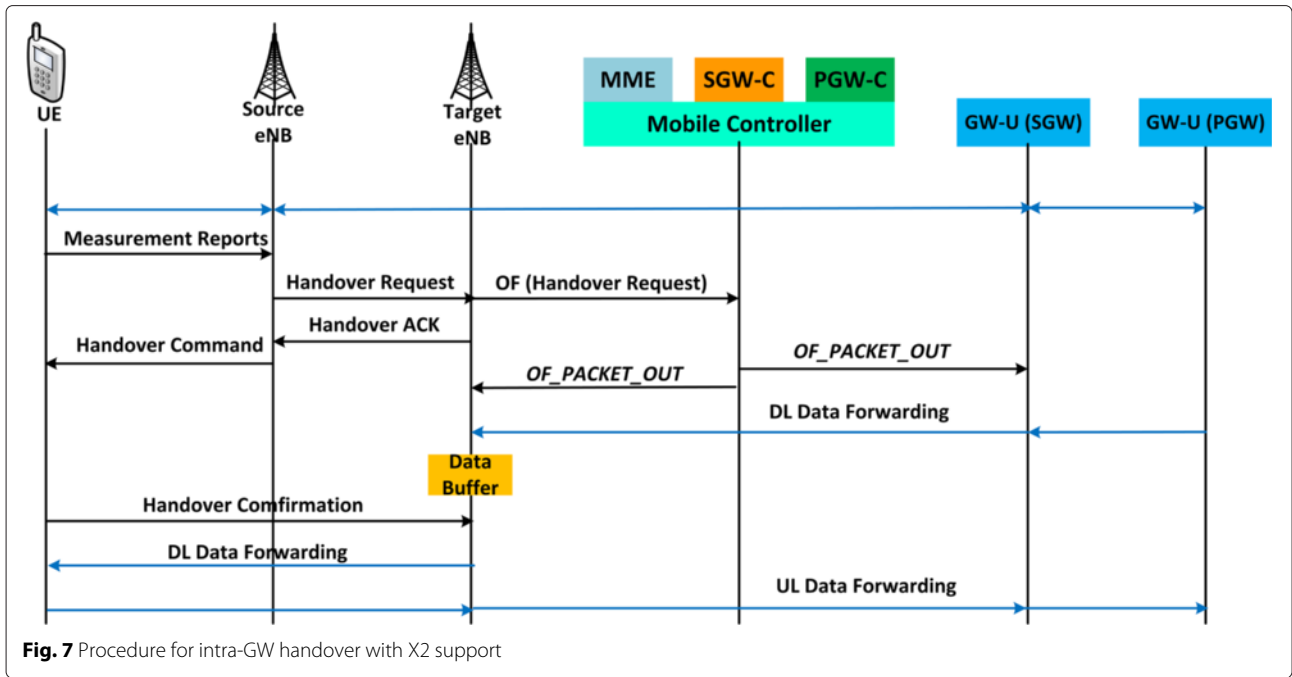

Fig. 6 Procedure for network-triggered service request

Fig. 7 Procedure for intra-GW handover with X2 support

procedure for an intra-GW handover without the support of an X2 connection. Overall, this procedure is quite similar to that of an S1 handover in a traditional LTE/EPC architecture, as described in [1]. When the source eNB realizes that the UE needs to undergo a handover to a

new eNB, it sends a Handover Required message packaged in an Openflow message to the MC. The MME function inside the MC will trigger the handover by sending a Handover Request message to the target eNB, and upon receiving the handover ACK from the target eNB,

Fig. 8 Procedure for intra-GW handover without X2 support
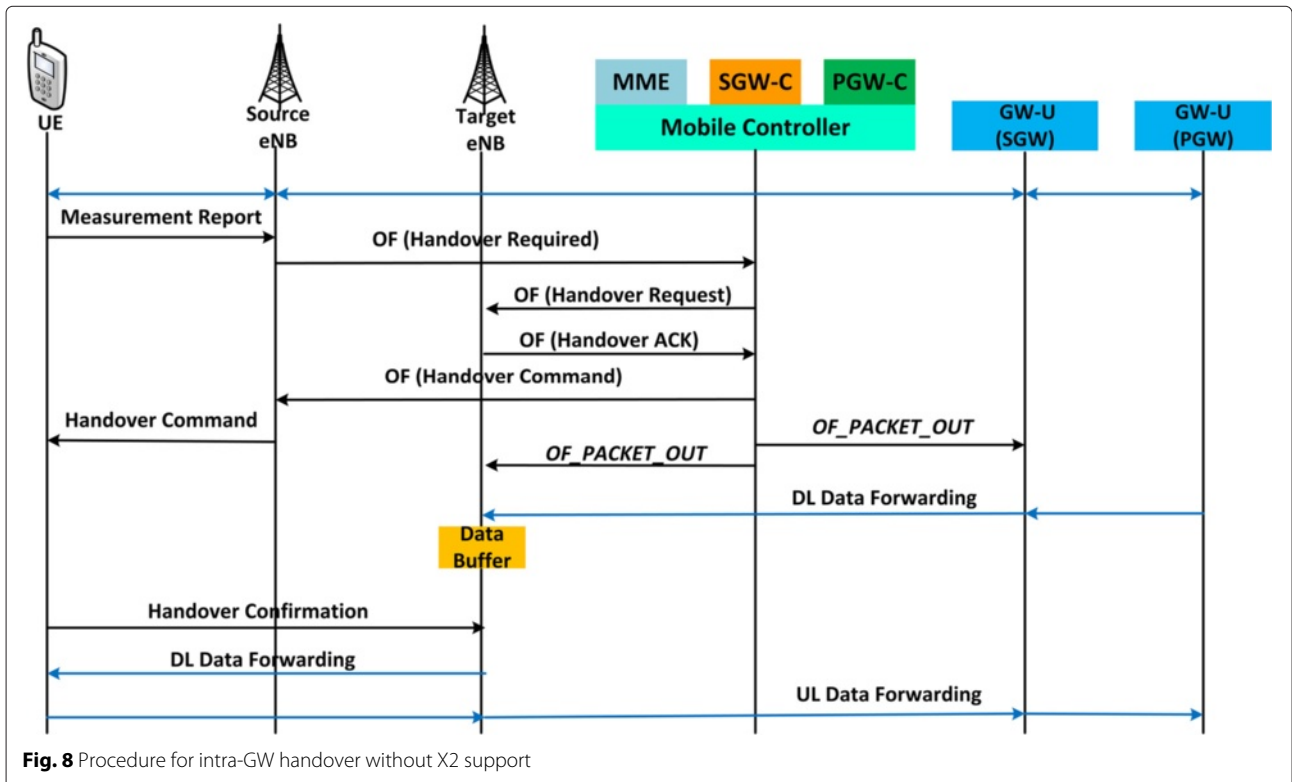
the MC prepares the resources and notifies the GW-U by connecting to both eNBs to establish a new data forwarding path. After the data forwarding path is established, the MC sends an Openflow Handover Command message to inform the source eNB to execute the handover. The source eNB reconfigures the radio resource for the UE, and meanwhile, the DL data is forwarded to the target eNB and is buffered there. After synchronization and radio connection reconfiguration are successfully completed, the DL data is then sent to the UE. Conversely, the UL data is sent from UE to the Internet through a tunnel between the target eNB and the GW-Us. Since data is no longer transferred to the source eNB, a timeout value will be set to release the resource. In the case of an inter-GW handover or gateway relocation, the MC needs to send Openflow Packet Out messages to the target GW-U (SGW) and GW-U (PGW) in order to modify their flow tables at the same time that the MC sends the Openflow Packet Out messages to the target eNB and the source GW-U (SGW).

4.5 Tracking area update

As described in [1], the tracking area update procedure occurs when a UE enters a new TA that is not in the list of TAIs allocated by the MME function at the time when the UE is attached or when the TAU timer expires. The UE will update its new TA to the network irrespective of whether it is in an Idle or a Connected state. In the traditional LTE/EPC architecture, this procedure is different depending on whether or not the MME changes location. In the OEPC architecture, however, the tracking update procedure is quite constant. Figure 9 depicts the tracking area update procedure for the OEPC architecture. Here, the UE sends a TAU Request message to the eNB when it detects a new TA. This message is embedded in an Openflow message and is then sent to the MC. Upon receiving the TAU Request message, the MC informs the MME function in order to record the location information of the

UE and then performs a location update to the HSS server. In order to complete the tracking area update procedure, the UE sends a TAU Complete message to the MC. At the end of this procedure, the new location for the UE is updated.

5 Performance evaluation

In this section, we investigate the signalling load at the control entities of a traditional LTE/EPC architecture, a reference architecture [14, 15], and our full OEPC architecture. These control entities are the MME, the Openflow controller, and a mobile controller, respectively. The signalling loads at these control entities caused by five aforementioned procedures are considered as the evaluation metric. For convenience, we called the proposed architecture described in [14, 15] as a partial OEPC architecture because this architecture was also designed by using Openflow technology but it only separated the control and data plane of the SGWs while keeping the PGW the same as that in the traditional LTE/EPC architecture. In contrast, our fully OEPC decoupled completely the control and data plane of SGW as well as PGW and put all the control functions of those gateways into the MC.

5.1 Signaling load analysis

For the three tested architectures, we refer to the signalling analysis model described in [19–23] in order to analyze the signalling load that is generated by the five common procedures. We assume that each UE is a smart phone that supports K application types, including email, web, voice, etc. Let λ_k be the average arrival rate of the type-k session at a UE and P be the probability that a session is originated by a UE, respectively. Further, let C, A, and ρ respectively denote the total number of eNBs in a region, the area of a cell, and the UE density in a region. SL_1, SL_2, and SL_3 represent the signalling load of MME, Openflow controller, and mobile controller, respectively.
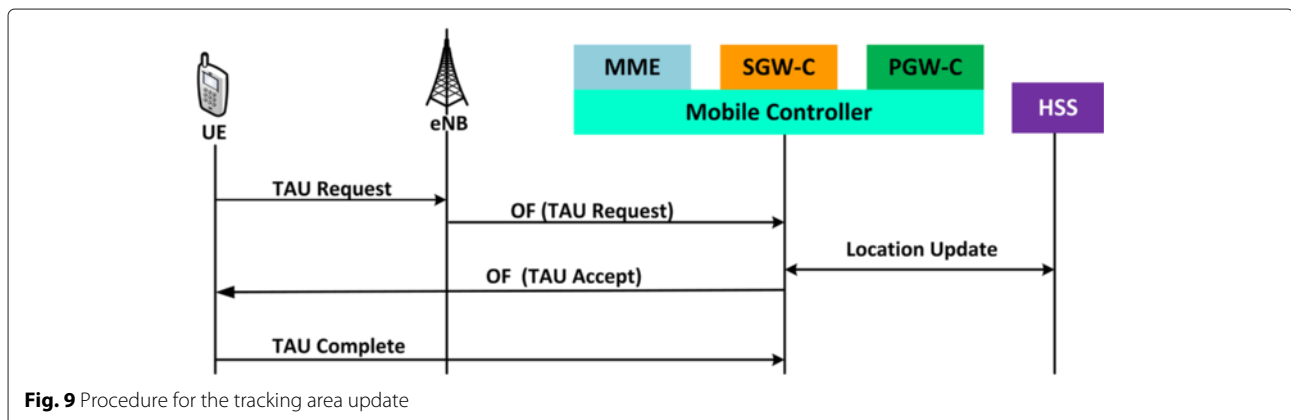

Fig. 9 Procedure for the tracking area update

5.1.1 Traditional LTE/EPC architecture

As said before, we referred the evaluation method from [19] to calculate signalling load for procedures in OEPC architecture. In order to calculate the signalling load for each procedure, we need to draw the call flows that show the message exchanging between entities in each procedure. In the traditional LTE/EPC architecture, these call flows were specified in 3GPP technical specification (TS 23.401 [1]) and partially presented in [19]. In [19], the authors analyzed four procedures for the traditional LTE/EPC architecture including UE-originated session (or UE-triggered service request), UE-terminated session (or network-triggered service request), handover with X2 support, and tracking area update (TAU). In this paper, we added an analysis for initial attachment procedure and the handover without X2 support (S1 handover) procedure for the traditional LTE/EPC architecture. Furthermore, based on these analyses in the traditional LTE/EPC architecture, we applied to analyze all procedures that occur in OEPC architectures.

From [19], the signalling load at control entities for each procedure is proportional to the number of messages entering and leaving these entities and the average arrival rate of session at a UE. The signalling load is affected by the average arrival rate of session generated by an application at a UE because we assumed that each UE

is a smart phone that can support multiple application types. However, in the special case like initial attachment procedure, the signalling load is not affected by the session arrival rate because the initial attachment procedure does not depend on what kinds of applications are used by the user. The following illustrates how to calculate the signalling load at MME entity caused by the initial attachment procedure. Let us take a look at Fig. 10 showing the procedure for initial attachment in the traditional LTE/EPC architecture. Except for the authentication and authorization step, it is clear that the total number of messages entering and leaving MME entity is ten messages. Therefore, the total signalling load at MME entity caused by the initial attachment procedure is given by
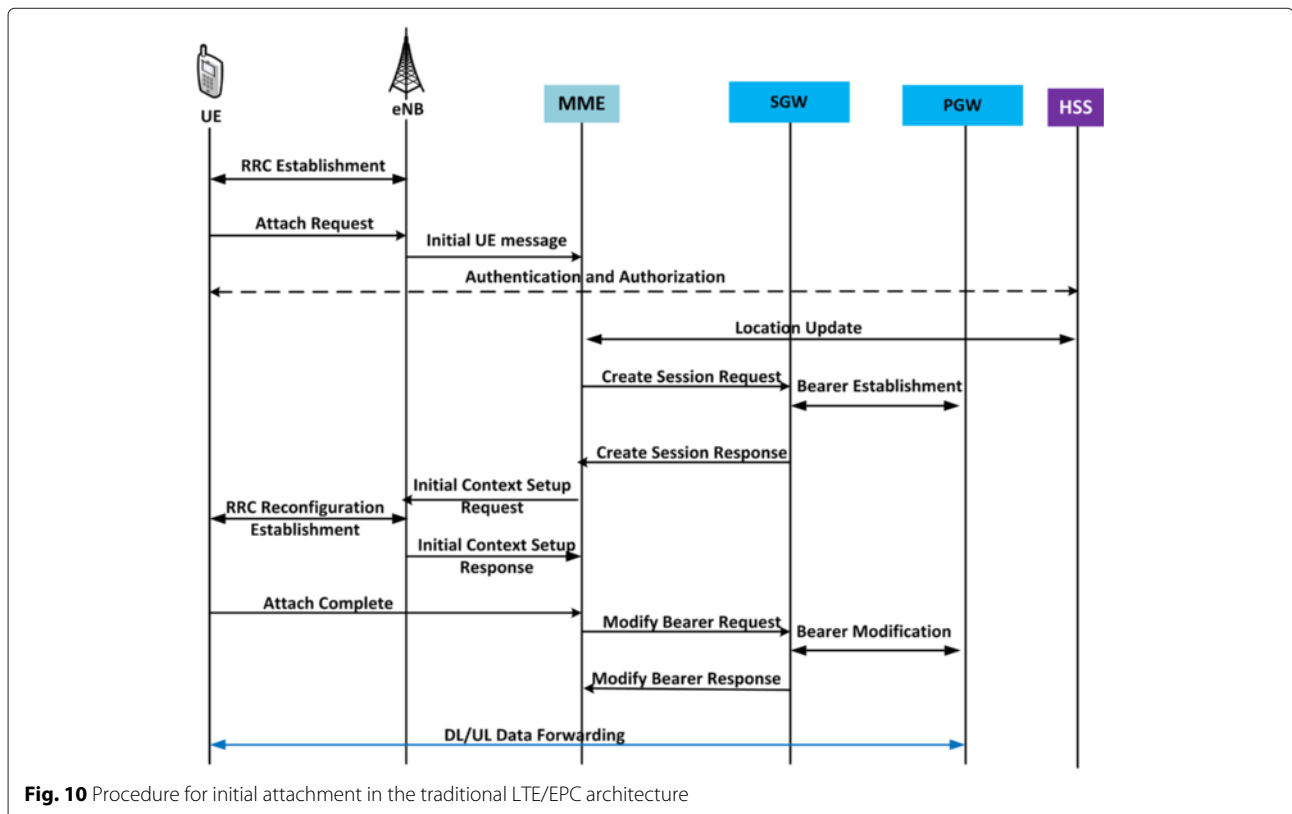
$$SL_{1IA}(k) = 10P_{\text{initial}}\rho AC \qquad (1)$$

where P_{initial} is the probability that an UE initiates an attachment procedure to the network.

Similarly, the signalling load at the MME entity caused by other procedures is calculated below.

The signalling load caused by the UE-triggered service request procedure is given as

$$SL_{1UE}(k) = 10\lambda_k P\rho AC \qquad (2)$$

Fig. 10 Procedure for initial attachment in the traditional LTE/EPC architecture

The signalling load caused by the network-triggered service request procedure with unicast and multicast paging is given as

$$SL_{1NWu}(k) = [(11+C_{\text{TAU}})R_pP_i+10(1-P_i)]\,\lambda_k(1-P)\rho AC \tag{3}$$

$$SL_{1NWm}(k) = [12R_pP_i + 10(1-P_i)]\,\lambda_k(1-P)\rho AC \tag{4}$$

where C_{TAU} is the size of tracking area and is calculated by the number of eNBs per tracking area, R_p is the average number of paging transmissions per page, and P_i is the probability that a UE is in Idle state. The reference paper [19] provided a proof that the probability that the UE in a Connected state is $(1 - P_i)$.

The signalling load caused by the handover procedure with and without X2 support is given as

$$SL_{1hX2}(k) = [4(1-P_r) + 6P_r]\,R(1-P_i)C \tag{5}$$

$$SL_{1h}(k) = [9(1-P_r) + 11P_r]\,R(1-P_i)C \tag{6}$$

where P_r is the relocation probability of SGW and can be well approximated by $\frac{1}{\sqrt{C_{\text{TAU}}}}$ [19], and R is the crossing rate of a set of UEs out of an enclosed region. We adopted a fluid-flow mobility model, which is described in [18] and is a widely used mobility model for modelling traffic in telecommunications networks, to compute the signaling load. It should be noted that UE may involve to the charging and policy control by network operator during its handover, regardless of SGW relocation as specified in 3GPP TS 23.401 [1]. We consider the case that UE does involve. It means that there are always two messages, Modify Bearer Request and Modify Bearer Response, exchanging between SGW and PGW during UE handover, regardless of SGW relocation.

The signalling load caused by the tracking area update procedure is given as

$$SL_{1\text{TAU}}(k) = \frac{3RC}{\sqrt{C_{\text{TAU}}}} \tag{7}$$

Finally, the total signalling load at the MME entity caused by five procedures in the case of unicast paging is simply the sum of Equations (1)–(3), (5) for handover with X2 support or (6) for handover without X2 support, and (7). For multicast paging support, the corresponding value will be the sum of Equations (1), (2), (4), (5) for handover with X2 support or (6) for handover without X2 support, and (7).

5.1.2 Partial OEPC architecture

In order to calculate the total signalling load in partial OEPC architecture, we need to draw call flows for each procedure occurring in this architecture. In [16], the authors presented two procedures namely initial attachment and UE-triggered service request procedure. The tracking area update procedure only required the presence of eNB, controller, and HSS entity so this procedure is the same as that described in Fig. 9. For handover procedure, there are four kinds of handovers (intra/inter-GW handover with/without X2 support) as defined in Section 4.4 but no handover procedure was drawn in [16]. Figure 11 shows the call flow for the intra-GW handover with X2 support in the partial OEPC architecture. The difference between this procedure and the procedure shown in Fig. 7 is the exchange of messages between the controller and PGW entity. Because the PGW in the partial OEPC architecture is the traditional PGW, so messages like modify bearer request and modify bearer response are still kept.

Using the same method as in the previous section, we can easily obtain the total signalling load at the Openflow controller caused by five procedures in the partial OEPC architecture. These values are given as follows.

The signalling load caused by the initial attachment procedure is given as

$$SL_{2IA}(k) = 8P_{\text{initial}}\rho AC \tag{8}$$

The signalling load caused by the UE-triggered service request procedure is given as

$$SL_{2UE}(k) = 8\lambda_k P\rho AC \tag{9}$$

The signalling load caused by the network-triggered service request procedure with unicast and multicast paging is given as

$$SL_{2NWu}(k) = [\,(9+C_{\text{TAU}})R_pP_i + 8(1-P_i)]\,\lambda_k(1-P)\rho AC \tag{10}$$

$$SL_{2NWm}(k) = [\,10R_pP_i + 8(1-P_i)]\,\lambda_k(1-P)\rho AC \tag{11}$$

The signalling load caused by the handover procedure with and without X2 support is given as

$$SL_{2hX2}(k) = [\,7(1-P_r) + 8P_r]\,R(1-P_i)C \tag{12}$$

$$SL_{2h}(k) = [\,8(1-P_r) + 9P_r]\,R(1-P_i)C \tag{13}$$

As mentioned in previous section, in the handover case, we assume that UE is involved with the charging and policy control by the network operator. Because PGW entity in partial OEPC architecture is kept the same as that in conventional one, thus in order to handle charging and policy control, SGW-C function on top of OF controller needs to exchange Modify Bearer Request and Modify Bearer Response messages with PGW.

The signalling load caused by the tracking area update procedure is given as

$$SL_{2\text{TAU}}(k) = \frac{3RC}{\sqrt{C_{\text{TAU}}}} \tag{14}$$
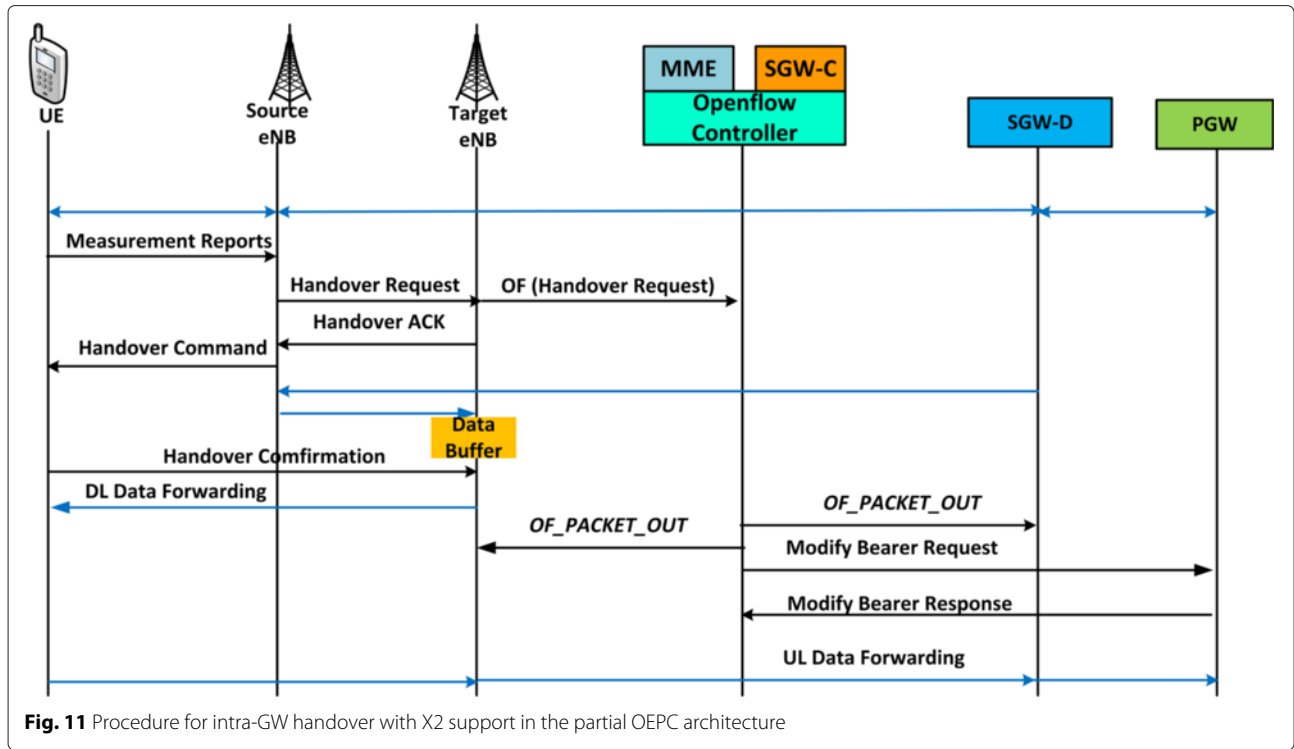
Fig. 11 Procedure for intra-GW handover with X2 support in the partial OEPC architecture

Finally, the total signalling load at the Openflow controller in the case of unicast paging is simply the sum of Equations (8)–(10), (12) for handover with X2 support or (13) for handover without X2 support, and (14). For multicast paging support, the corresponding value will be the sum of Equations (8), (9), (11), (12) for handover with X2 support or (13) for handover without X2 support, and (14).

5.1.3 Full OEPC architecture

The same method as in the previous section can be used to easily obtain the total signalling load at the mobile controller caused by the five procedures in the full OEPC architecture. The five procedures were described in detail in Section 5. The signalling load caused by each procedure is given as follows.

The signalling load caused by the initial attachment procedure is given as

$$SL_{3IA}(k) = 6P_{\text{initial}}\rho AC \tag{15}$$

The signalling load caused by the UE-triggered service request procedure is given as

$$SL_{3UE}(k) = 7\lambda_k P \rho AC \tag{16}$$

The signalling load caused by the network-triggered service request procedure with unicast and multicast paging is given as

$$SL_{3NWu}(k) = [(8 + C_{\text{TAU}})R_p P_i + 7(1 - P_i)] \lambda_k (1 - P)\rho AC \tag{17}$$

$$SL_{3NWm}(k) = [9R_p P_i + 7(1 - P_i)] \lambda_k (1 - P)\rho AC \tag{18}$$

The signalling load caused by the handover procedure with and without X2 support is given as

$$SL_{3hX2}(k) = [3(1 - P_r) + 5P_r] R(1 - P_i)C \tag{19}$$

$$SL_{3h}(k) = [6(1 - P_r) + 8P_r] R(1 - P_i)C \tag{20}$$

In this handover case, we assume that the charging and policy control-related information can be included in Openflow messages. It is not necessary to exchange Modify Bearer Request and Modify Bearer Response messages between MC and PGW. As a result, the handover signalling load in full OEPC architecture is much lower than those of traditional and partial OEPC architectures.

The signalling load caused by the tracking area update procedure is given as

$$SL_{3TAU}(k) = \frac{3RC}{\sqrt{C_{\text{TAU}}}} \tag{21}$$

Finally, the total signalling load at the mobile controller caused by five procedures in the case of unicast paging is simply the sum of Equations (15)–(17), (19) for handover with X2 support or (20) for handover without X2 support, and (21). For multicast paging support, the corresponding value will be the sum of Equations (15), (16), (18), (19) for handover with X2 support or (20) for handover without X2 support, and (21).

5.2 Numerical results

In this section, we present numerical results for all of the equations described in the previous section. The values for almost parameters in equations from (1)–(21) comes from the work in [19]. The application types with average arrival rate and duration are presented in Table one in [19]. Finally, we have a parameter table for evaluation as shown in Table 1. Some of the parameters can be varied or inferred from other parameters so they have no values in that table. We evaluate the total number of signaling messages that are processed by MME, the Openflow controller, and the mobile controller. We also show the impact of these values on the tracking size (C_{TAU}), the total number of UEs as well as the velocity of UE.

For the first scenario, we compare the signaling loads for the three control entities of the three architectures and show the impact that these values have on the tracking area size for both unicast paging and multicast paging.

Table 1 Parameter for evaluation

Parameter	Description	Value
$P_{initial}$	Probablity that a UE initiates an attachment procedure	0.2
P	Probablity that a session is generated by UE	0.5
λ_k	Average arrival rate of session type-k	0.4
$\frac{1}{\mu_k}$	Averate duration of session type-k	0.05
$P_i = \frac{1}{1 + \frac{\lambda_k}{\mu_k}}$	Probability that a UE is in Idle state	
$1 - P_i$	Probability that a UE is in Connected state	
S	Area of a considered region (km^2)	500
Nue	Total number of UEs	
$\rho = \frac{Nue}{S}$	UE density (UEs/km^2)	
C	Total number of eNBs in a considered region	500
R_p	Average number of paging transmission per page	1.1
V	Velocity of UE	
γ	Overlapping factor	1.2
$r = \gamma \sqrt{\frac{S}{C\pi}}$	Radius of a cell	
$L = 2\pi r$	Perimeter length of a cell (km)	
$R = \frac{\rho L V}{pi}$	Crossing rate out of a cell (UEs/hour)	
C_{TAU}	Tracking area size	
P_r	GW relocation probability	

The default values for this scenario are $V = 20$ km/h, Nue $= 2.10^6$ and the numerical results are shown in Fig. 12. This figure clearly indicates that the full OEPC architecture can reduce the signaling messages that are processed by the mobile controller by more than that achieved by the Openflow controller in a partial OEPC architecture in [14, 15]. This is a result of the reduction in the number of messages that are exchanged between the Openflow controller and PGW in a partial OEPC architecture. As shown in this figure, when C_{TAU} is too small (lower than 10), the signalling load for all control entities is too high for either unicast paging or multicast paging. This value gradually decreases when the tracking size increases. It should be noted that the paging type does not affect the signalling load caused by the two handover procedures (with and without X2 support). However, as shown in Equations (3) and (4) in the traditional LTE/EPC architecture, the signalling load caused by the network-triggered service request is proportional to the C_{TAU} value in the case of unicast paging while the signalling load is constant to the C_{TAU} value in the case of multicast paging. It means that this signalling load is affected by the paging type. Therefore, the total signalling load in the traditional LTE/EPC network is affected by the type of paging mechanism. It is similar to the partial OEPC and the full OEPC network. Such reason results in the difference in the signaling loads between unicast and multicast paging as shown in Fig. 12a, b.
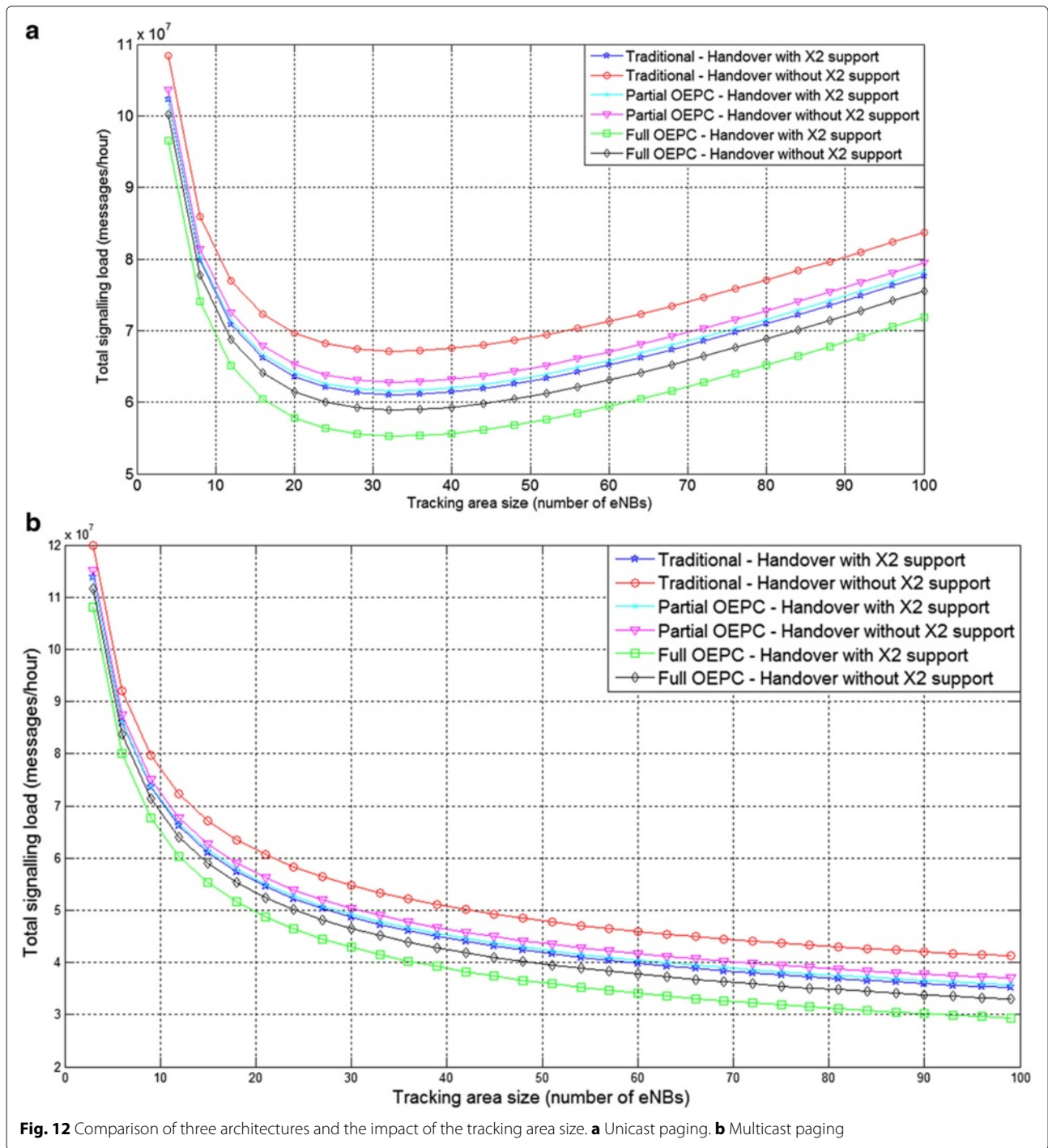
For the second scenario, we show a comparison of the signaling loads for the three architectures and the impact of the number of UEs. The results are shown in Fig. 13a, b. The number of UEs varied from 0 to 1000. Overall, the signaling loads increase in a linear manner as the number of UEs increases. As expected, the full OEPC architecture has the lowest signaling loads for either unicast or multicast paging when compared to those of the traditional LTE/EPC and partial OEPC architectures.

For the third scenario, we show the impact of the signaling loads on the velocity of UE, and we also compare the signaling loads among the three architectures. The velocity of the UE varies in a range from 0 to 100 km/h. As depicted in Fig. 14, the signaling loads increase in a manner that is directly proportional to the handover rate. Again, the total signaling loads processed by mobile controller in the OEPC architecture are lower than those of the other two architectures.
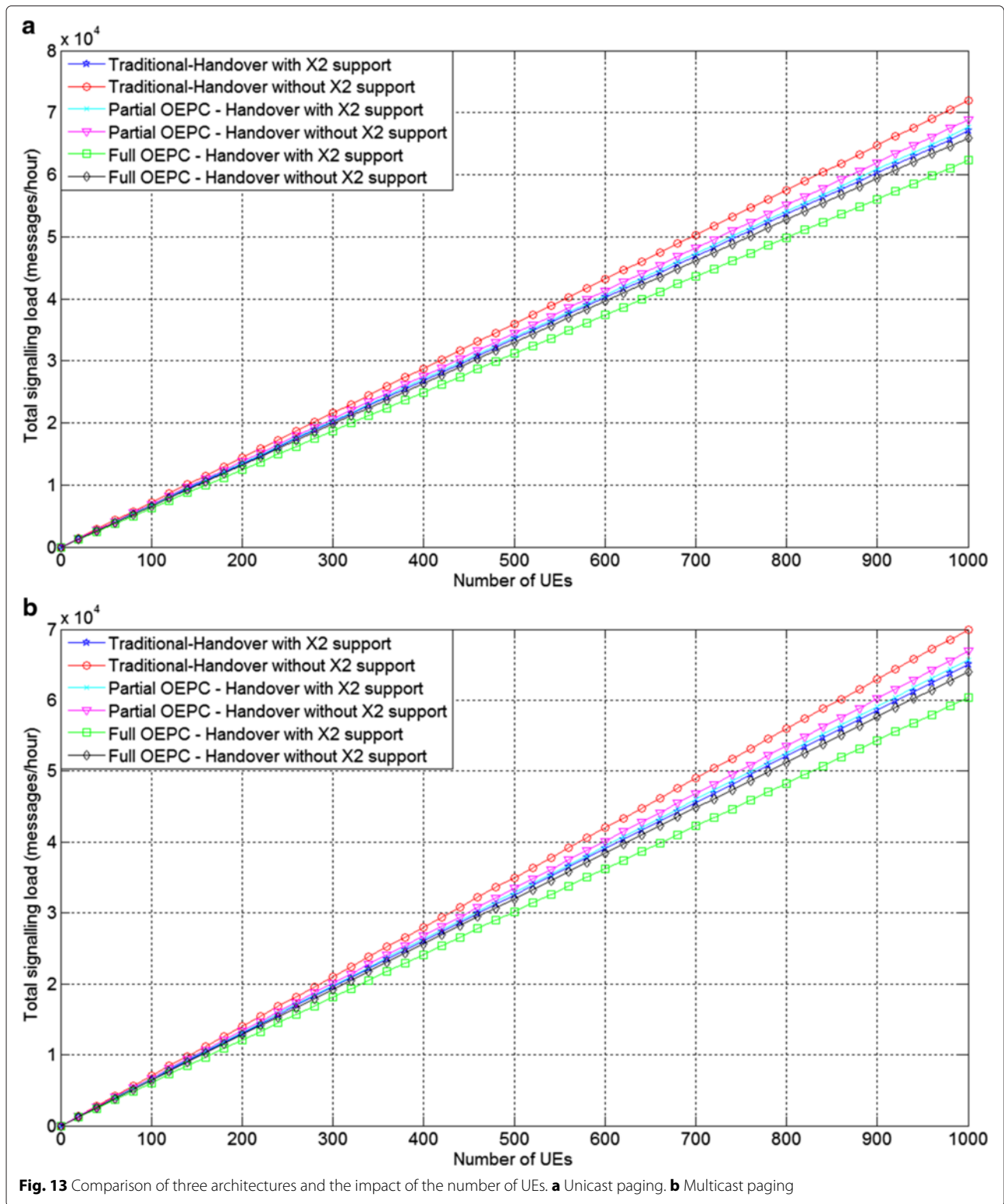
6 Conclusions

In this paper, we propose a new Openflow-enabled mobile packet core network, OEPC. Five common procedures are analyzed in detail, including the initial attachment, UE-triggered service request, network-triggered service request, handover, and the tracking area update. The numerical results of the evaluation indicate that the

Fig. 12 Comparison of three architectures and the impact of the tracking area size. **a** Unicast paging. **b** Multicast paging
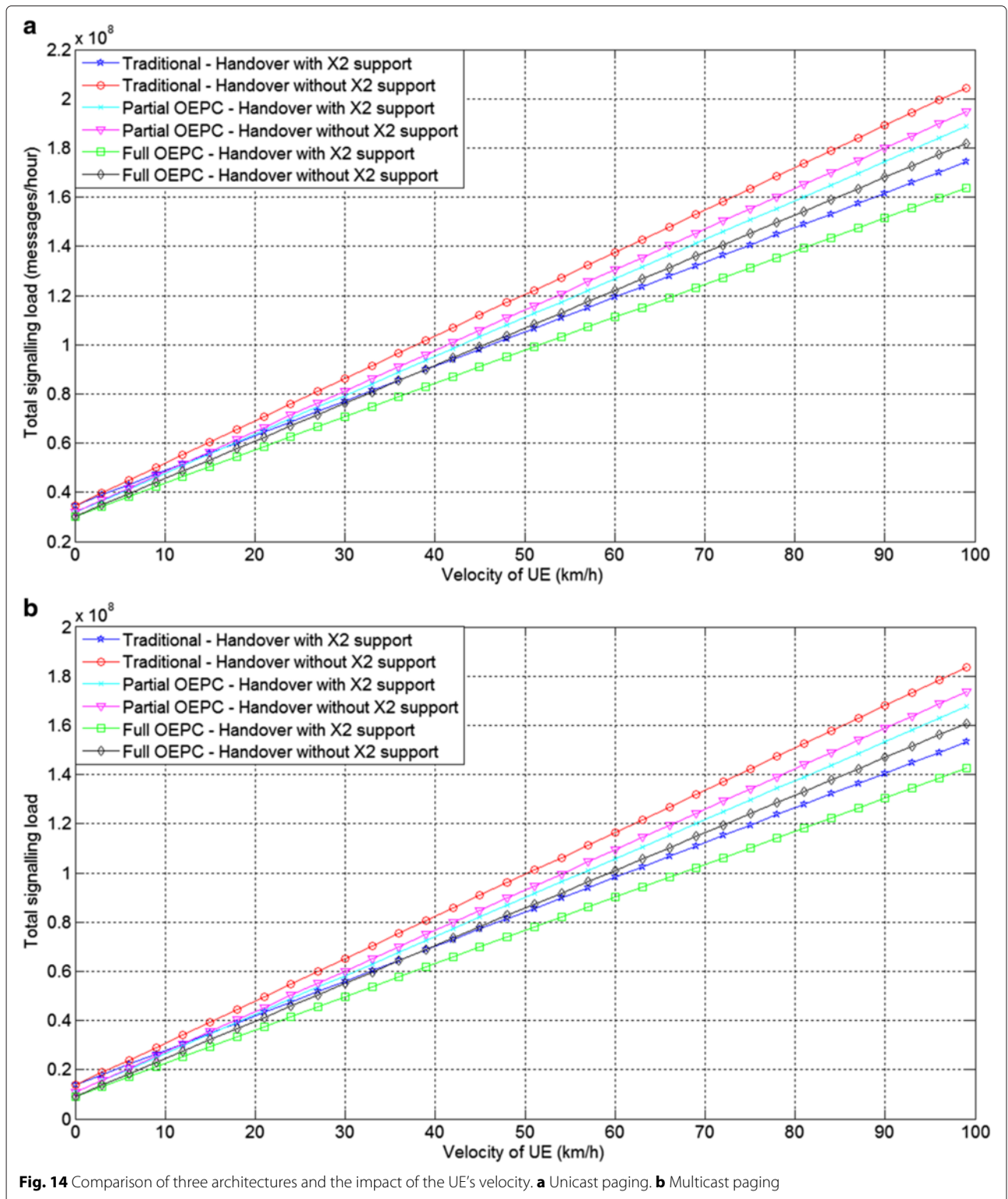
proposed architecture can reduce the signalling load relative to that of the traditional LTE/EPC architecture as well as to a reference architecture. Although these reductions are not much, other benefits of this architecture are flexibility, high rate of innovation, and ease of configuration and management. In future studies, we will continue to implement the OEPC architecture

with OpenEPC platform [24] and will enhance this architecture with a MobileVisor concept as described in another study [25]. MobileVisor enables to support the mobile virtual network operator (MVNO) concept and allows for multiple mobile operators to run their own mobile network with a shared underlying mobile packet core network. The final target of this work is

Fig. 13 Comparison of three architectures and the impact of the number of UEs. **a** Unicast paging. **b** Multicast paging

to make a real test-bed and to perform a demonstration with a real mobile phone and an LTE eNodeB. In addition, new mobility management paradigm (e.g., distributed mobility management) in the OEPC architecture will be taken into consideration as our future work.

Fig. 14 Comparison of three architectures and the impact of the UE's velocity. **a** Unicast paging. **b** Multicast paging

References
1. 3GPP, General packet radio service (GPRS) enhancements for evolved universal terrestrial radio access network (EUTRAN) access. Technical Specification (TS 23.401) (2010). [Online] http://www.3gpp.org/DynaReport/23401.htm. Accessed May 2015
2. Open Networking Foundation (ONF), Software-defined networking: The new norm for networks. ONF White paper (2012). [Online] https://www.opennetworking.org/images/stories/downloads/sdn-resources/white-papers/wp-sdn-newnorm.pdf. Accessed May 2015
3. N McKeown, T Anderson, H Balakrishnan, G Parulkar, L Peterson, J Rexford, S Shenker, J Turner, OpenFlow: enabling innovation in campus networks. ACM Comput. Commun. Rev. **38**(2), 69–74 (2008)
4. Open Networking Foundation (ONF) (2011). [Online] https://www.opennetworking.org. Accessed 28 Nov 2014
5. K Pentikousis, Y Wang, W Hu, Mobileflow: towards software-defined mobile networks. IEEE Commun. Mag. **51**(7), 44–53 (2013)
6. X Jin, LE Li, L Vanbever, J Rexford, in *Proceedings of 9th ACM International Conference on emerging Networking EXperiments and Technologies (CoNEXT)*. Softcell: scalable and flexible cellular core network architecture (ACM, California, USA, 2013), pp. 163–174
7. S Shanmugalingam, P Bertin, in *Proceedings of the nineteenth IEEE Symposium on Computers and Communications (ISCC)*. Programmable mobile core network (IEEE, Madeira, Portugal, 2014), pp. 1–7
8. G Motoyoshi, K Leibnitz, M Murata, Proposal and evaluation of a future mobile network management mechanism with attractor selection. EURASIP J. Wireless Commun. Netw. **2012**(1), 1–13 (2012)
9. SM Raza, D Kim, H Choo, in *Proceedings of ACM International Conference on Research in Adaptive and Convergent Systems (RACS)*. The proposal for sdn supported future 5G networks (ACM, Maryland, USA, 2014), pp. 180–185
10. W A Basta, M Kellerer, K Hoffmann, ED Hoffmann, A Schmidt, in *Proceedings of the first IEEE Workshop on Software Defined Network for Future Networks and Services (SDN4FNS)*. virtual sdn-enabled LTE/EPC architecture: a case study for s-/p-gateways functions (IEEE, Trento, Italy, 2013), pp. 1–7
11. MR Sama, LM Contreras, J Kaippallimalil, I Akiyoshi, H Qian, H Ni, Software-defined control of the virtualized mobile packet core. IEEE Commun. Mag. **53**(2), 107–115 (2015)
12. V Yazici, UC Kozat, MO Sunay, A new control plane for 5G network architecture with a case study on unified handoff, mobility, and routing management. IEEE Commun. Mag. **52**(11), 76–85 (2014)
13. M Yang, Y Li, B Li, D Jin, S Chen, Service-oriented 5G network architecture: an end-to-end software defining approach. J. Commun. Syst (2015). [Online]. doi:10.1002/dac.2941
14. SBH Said, MR Sama, K Guillouard, L Suciu, G Simon, X Lagrange, J-M Bonnin, in *Proceedings of second IEEE International Conference on Cloud Networking (CLOUDNET)*. New control plane in 3GPP LTE/EPC architecture for on-demand connectivity service (IEEE, San Francisco, USA, 2013), pp. 205–209
15. MR Sama, SBH Said, K Guillouard, L Suciu, in *Proceedings of 12th IEEE International Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOpt)*. Enabling network programmability in LTE/EPC architecture using openflow (IEEE, Tunisia, 2014), pp. 395–402
16. VG Nguyen, Y Kim, in *Proceedings of the fifth IEEE Conference on ICT Convergence (ICTC)*. Signaling load analysis in Openflow-enabled LTE/EPC architecture (Busan, Korea IEEE, 2014), pp. 734–735
17. A Gudipati, D Perry, LE Li, S Katti, in *Proceedings of the 2nd Workshop on Hot topics in Software Defined Networking (HotSDN)*. SoftRAN: software defined radio access networks (ACM, Hongkong, China, 2013), pp. 25–30
18. Foundation(ONF) Open Networking, Openflow specification version 1.4.0 (2013). [Online]. https://www.opennetworking.org/images/stories/downloads/sdn-resources/onf-specifications/openflow/openflow-specv1.4.0.pdf. Accessed August 2014
19. I Widjaja, P Bosch, HL Roche, in *Proceedings of the 70th IEEEE Vehicular Technology Conference Fall (VTC 2009-Fall)*. Comparison of MME signaling loads for long-term-evolution architectures (IEEE, Alaska, USA, 2009), pp. 1–5
20. PK Gunpta, RV Rajakumar, CS Kumar. Analysis of impact of network activity on energy efficient of 3GPP-LTE. in: Proceedings of IEEE Annual India Conference (INDICON) (IEEE, Kochi, India, 2012), pp. 665–669
21. X An, F Pianese, I Widjaja, UG Acer, DMME: A distributed lte mobility management entity. Bell Labs Tech J. **17**(2), 97–120 (2012)
22. I Sato, A Bouabdallah, X Lagrange, A new lte/epc control-plane based transmission procedure to cope with short data push services. Wirel Pers. Commun. **72**(3), 1723–1735 (2013)
23. D Lam, DC Cox, J Widom, Teletraffic modeling for personal communications services. IEEE Commun. Mag. **35**(2), 79–87 (1997)
24. OpenEPC, Open evolved packet core. [Online] http://www.openepc.net/en/openepc/index.html. Accessed December 2014
25. VG Nguyen, Y Kim, in *Proceedings of 11th IEEE International Symposium on Wireless Communications Systems (ISWCS)*. Slicing the next mobile packet core network (IEEE, Barcelona, Spain, 2014), pp. 901–904