BULGARIAN ACADEMY OF SCIENCES

CYBERNETICS AND INFORMATION TECHNOLOGIES • Volume **19**, No 4
Sofia • 2019 Print ISSN: 1311-9702; Online ISSN: 1314-4081
DOI: 10.2478/cait-2019-0037

Mission Critical Messaging Using Multi-Access Edge Computing

Evelina N. Pencheva, Ivaylo I. Atanasov, Vladislav G. Vladislavov

Technical University of Sofia, 8 St. Kliment Ohridski blvd, 1000 Sofia, Bulgaria
E-mails: enp@tu-sofia.bg iia@tu-sofia.bg vladislav.vladislavov@balkantel.net

Abstract: *5th Generation* (*5G*) *mobile system is expected to support the requirements of mission critical communications for ultra reliability and availability, and very low latency. With the development of messaging and data transfer in mobile networks, mission critical communication users see more and more potential in data communications. In this paper, we explore the capabilities of Multi-access Edge Computing* (*MEC*) *that appears to be a key 5G component, to provide short messaging service at the network edge. The provided use cases illustrate the capabilities for transferring mobile originating and mobile terminating short messages to and from mission critical mobile edge applications. The data model describes the service resource structure and the Application Programming Interface definitions illustrate how the mobile edge applications can use the service. Some implementation aspects related to behavioral logic of the network and applications are provided. The performance analysis enables estimation of latency introduced by the service.*

Keywords: *Mission Critical Communications, Fifth Generation Mobile System, Edge Computing, Messaging Service.*

1. Introduction

Mission Critical Communications are used by government and public security services, such as police, fire brigades, emergency assistance and a number of vertical segments such as energy, transport companies and others. The common to all of these customer segments is the need for highly reliable communications, high availability, uncompromised security, wide geographical coverage, and high quality of service [1].

Traditionally, this kind of communications has been provided by a combination of dedicated private networks and public services. The need of improvement of the existing mission critical communications requires the use of 5th-Generation (5G) public networks that have the potential to deliver high-speed connections, massive machine-type communications, ultra-high reliability and low latency [2, 3].

The architectural design of 5G mobile system exposes level of higher flexibility by connecting Network Function Virtualization (NFV), Software Defined Networking (SDN), and Cloud Computing, that can address the requirements of various applications for mission critical communications. The NFV concept addresses cost reduction, high performance and scalability by virtualizing network resources. The NFV decouples the network functions from proprietary hardware equipment so they can run as software on standardized hardware. SDN is a technology in which network hardware is separated from the control logic to simplify network management and to enable evolution. Cloud computing enables access to customized applications from anywhere having Internet connection. Centralized cloud computing solutions cannot address challenges such as low latency, limited resources and capacity constraints of mobile devices, and service continuity. Communication service providers have realized the need to provide computing and storage resources, and network infrastructure closer to where these applications are consumed. The distribution of cloud computing capabilities for data processing and storage close to the end user enables better bandwidth utilization and power consumption. Multi-access Edge Computing (MEC), which is regarded as a key component of 5G, can deliver high transmission rates, low latency and high availability. Contemporary research on mission critical communications focuses primarily on mechanisms and frameworks to ensure high reliability [4-6].

In this paper, we study the capabilities to provide mission critical messaging service at the network edge using MEC technology. We also evaluate the latency introduced by the service. The latency in next generation wireless networks is studied in [7, 8]. The vicinity to end users of MEC services enables further latency reduction.

The rest of the paper is organized as follow. Next section argues the research motivation. Section 3 presents typical use cases of open access to messaging service at the mobile network edge. Data model that enables third papery applications to access and store data in uniform way is described in Section 4. Simplified definitions of the proposed Application Programming Interfaces (API) are presented in Section 5, and some implementation aspects are considered in Section 6. Section 7 provides API performance analysis. The conclusion summarizes authors' contribution.

2. Research motivation

Messaging service plays an important role in communicating mission critical information during censorious situations. Mission critical messaging must be effective and scalable in order to adapt to different scenarios. The Short Messaging Service (SMS) which has gained a lot of popularity is still the cheapest, fast an easy to use form of communications. It is effective especially in situations with weak radio signal, which is important in censorious situations. The voice communications may be useful for warning, but text messaging service may provide more detailed information that can be reread many times.

The SMS is provided in 5G networks also [9, 10]. There are two methods for transportation of text messages over 5G access, namely SMS over

Internet Protocol (IP) using IP Multimedia Subsystem (IMS) and SMS over Non-Access Signalling (NAS). In the latter case, the short message is sent via NAS transport message, which can carry SMS messages as payload. As the SMS is non-real time service and it is based on the principle of "store and forward", in both methods the messages are stored in the network and then delivered to the destination address. Due to delays in message delivery, the traditional SMS cannot meet the requirements for ultra low latency of mission critical communications.

The necessity for open access to messaging service is recognized with the standardization of Parlay X Short Messaging service [11]. Third party application can send or receive SMS through Parlay X gateway deployed in the operator's core network. In this paper, we propose an approach to open access to messaging service at the network edge using MEC. Deployment of open access to text messaging at the network edge does not require neither SMS Centre nor Parlay X gateway, nor IMS. It may be implemented locally in the MEC platform, which hosts mobile edge applications and provides mobile edge services.

Third party text messaging at the network edge is an alternative way to communicate messages in near to real time without keeping the message for long time. The reasons to have alternative way for message communication are many and among them are core network congestion situations, cyberattacks over SMS Center leading to temporary unavailability, etc. Third party text messaging at the network edge decreases the core network signaling and saves network resources while at the same time delivers timely location based information. The text messaging service at the network edge can be provided by using NAS signaling.

There exist several MEC deployment scenarios in 5G [12, 13].

The MEC host is an entity that contains a MEC platform and a virtualization infrastructure that provides compute, storage, and network resources, for the purpose of running MEC applications. The MEC platform is the collection of essential functionality that provides MEC applications environment. It contains also security services required to secure the interworking to other MEC platforms and application isolation, integrity and resilience, secure user's access and use of MEC applications [14].

The most appropriate MEC deployment that serves mission critical communications is when the MEC host logically includes all or part of core network components (Fig. 1). This scenario enables co-location of core network services and MEC services at 5G edge site. MEC applications can share the same NFV platform with virtualized core functions such as Access and Mobility Management Function (AMF), Short Messaging Service Function (SMSF), User Plane Function (UPF) and User Data Management (UDM). The scenario improves scalability and enables better network resource utilization.

To provide messaging service at the network edge, we propose a new mobile edge service called Edge Short Messaging Service (ESMS). As a MEC service it uses the management and orchestration services provided by the MEC platform including existing security functionality.

The next section describes typical use cases illustrating the ESMS functionality.

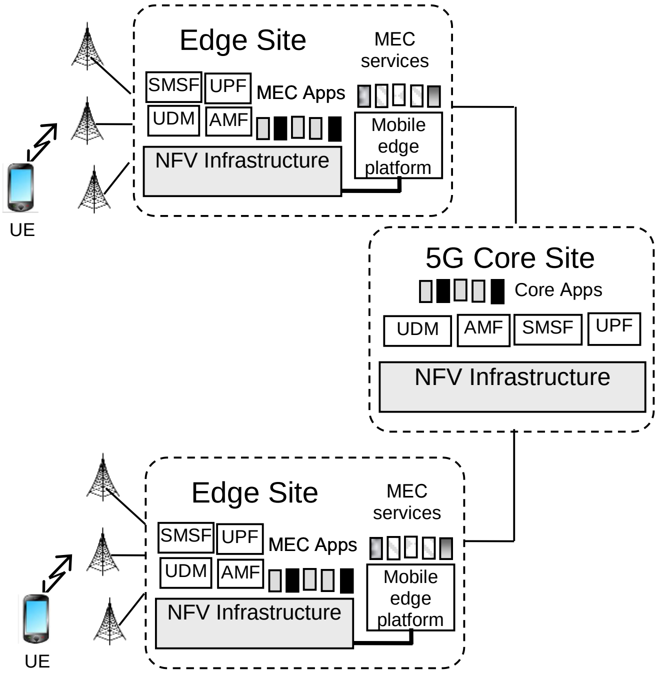

Fig. 1. MEC deployment in 5G to support mission critical applications

3. Use cases for open access to messaging service at the network edge

The ESMS provides open access to messaging service in vicinity of end users. The service supports the following functionality to support SMS over NAS:

- Relay a short message from User Equipment (UE) toward a mobile edge application;
- Relay a short message from a mobile edge application toward a UE.

To enable SMS over NAS transporting, the UE needs to indicate the SMS support during registration procedure. A mobile edge application needs to have information about registered UEs to be able to send and receive short messages to and from registered UEs.

Fig. 2 shows a scenario, where the mobile edge application sends a query to receive information about UEs registered for SMS over NAS.

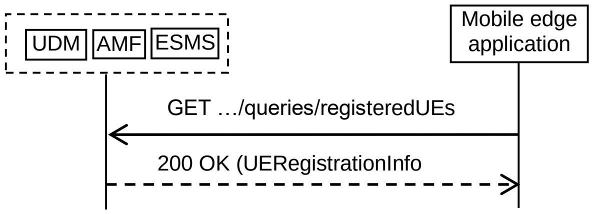

Fig. 2. Flowchart of mobile edge application requesting information about registered UEs

To receive notifications about UE registrations with support of SMS over NAS, as well as notifications about received mobile originating messages, the mobile edge application creates a subscription. Fig. 3 shows a scenario where the mobile edge application uses REST (REpresentational State Transfer) based procedure to create a subscription.

When the mobile edge application wants to receive notifications about UE registrations or received mobile originating messages, it sends a POST request with message body containing the {NotificationSubscription} data structure to the resource representing the subscription type of interest. The variable {subsriptionType} is replaced by smsRegistration, or smsDeregistration, or moMessages data type specified for different subscriptions, UE registrations, or UE de-registrations or received mobile originating messages respectively. The data structure {NotificationSubscription} is replaced with the data type specified for the different subscription, and defines the subscribed event, filtering criteria and the address where the mobile edge application wishes to receive notifications about UE registration/deregistration events. The ESMS sends “201 Created” response with the message body containing the data structure specific to that subscription. The data structure contains the address of the resource created to receive the notifications.

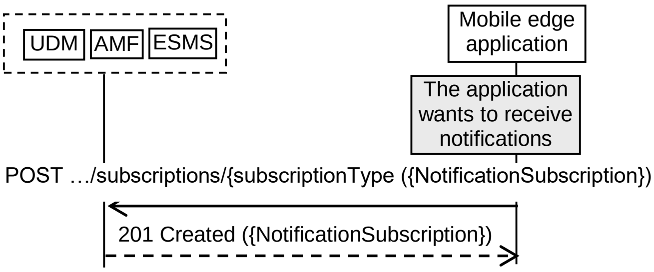

Fig. 3. Flowchart of subscribing for notifications about UE registrations

In case of active subscription, the ESMS sends a notification to the mobile edge application upon UE registration with indication for support of SMS over NAS, or upon UE de-registration, or upon receiving of mobile originating message. Fig. 4 presents the scenario where the ESMS sends notification upon UE registration to the mobile edge application. The notification contains the identifiers related to the UE and UE capabilities for SMS over NAS transport.

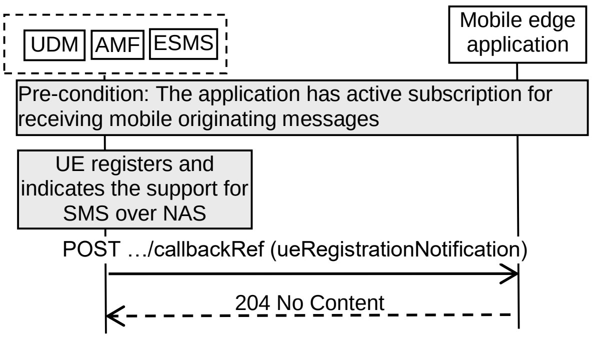

Fig. 4. Flowchart of ESMS sending notification on UE registration with indication for SMS support

The ESMS sends a POST request with message body containing the ueRegistrationNotification data structure to the call-back reference address provided by the application. The application sends “204 No Content” response to the ESMS.

Following the same pattern, when the UE indicates to the AMF that it wants no more to send and receive SMS over NAS, the mobile edge application is notified if it has active subscription for UE de-registrations.

If an UE is going to send a mobile originating message, then UE and network edge perform the UE Triggered Service Request procedure firstly to establish a NAS signalling connection to AMF. The UE builds the message to be sent and encapsulates the message in NAS message indicating the NAS message is for SMS transporting. The UE sends the NAS message to the AMF. The AMF forwards the message to the ESMS, which in turn checks the SMS subscription data in UDM. If SMS delivery is allowed, the ESMS checks whether the SMS message is dedicated for the mobile edge application or not. If it is, the ESMS notifies the mobile edge application about the incoming message. Fig. 5 shows the scenario where the mobile edge application is notified about mobile originating SMS. The moSMSNotification data structure contains information about the message and the message itself.

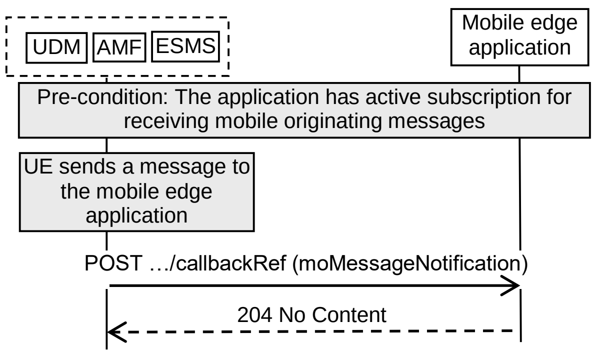

Fig. 5. Flowchart of ESMS sending notification on reception of mobile originating message intended to the mobile edge application

Fig. 6 shows a scenario where a mobile edge application sends a SMS message to UE. When the mobile edge application wants to send a short message, it sends a POST request to the resource sentMessages representing the messages sent by the application. The request body contains the message to be sent and the address where the mobile edge application wants to receive notifications about message delivery. The ESMS responds with message sending approval and message identification.

In order to be notified about message delivery, the mobile edge application needs to create a subscription, sending a POST request. An existing subscription for receiving notifications about delivery status of a message sent by a mobile edge application may be modified using PUT request or terminating using DELETE request.

Example use cases of messaging between a mobile and application server are described in [15]. The shared bike service allows a customer to rent a bike via a mobile application and drop it off anywhere for the next customer. The service includes message communications between customers and the bike via service platform, used to localize the customer, to reserve and to unlock the bike. In this use case, the customer sends a message to the server to request to unlock the bike. The server sends a message to the bike to unlock it. When unlocked, the bike sends a message to the server to report the result.

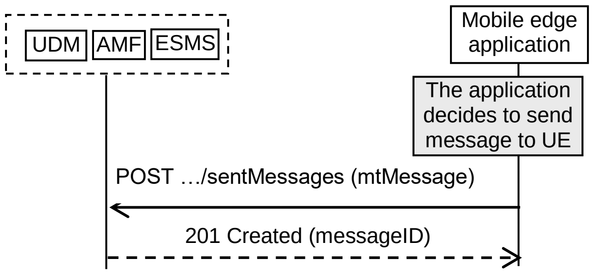

Fig. 6. Flowchart of mobile edge application sending a message to UE

4. Service data model

The data model determines explicitly the structure of information exchanged between the ESMS and mobile edge applications. The logic behind the use of the proposed data model is to enable different mobile edge applications to access and store data in a unified way.

The UERegistrationInfo type represents a subscription for notifications to UE registration for SMS from ESMS. The attributes of this type are as follows:

- callbackReference – URI provided by the mobile edge application to receive notifications on UE registrations for SMS. This attribute has to be included both in the request and in response;
- filterCriteriaReg – List of filtering criteria for the subscription. Filtering criteria are included both in the request and in response. Its structure is as follows: appInsId, plmn, cellId, regStatus. The appInsId is unique identifier for the mobile edge application instance. The plmn is a structure of mobile network code and mobile country code. The regStatus is enumeration type indicating UE registrations or UE de-registrations;
- expiryDeadline – Timestamp indicating the subscription expiry.

The smsRegistration type represents a notification from ESMS with regard to UE registration. The notification is sent by the ESMS to inform about UE registration for SMS. The attributes of this type include the following:

- timeStamp – self explained;
- cellGlobalId – Unique Cell Global Identity is a structure of mcc (mobile country code), mnc (mobile network code) and cellId (cell identity) as defined in [16];
- regStatus – Enumeration type indicating the status of registration: completed or rejected;
- tempUeId – The temporary identifier allocated for the specific UE. It is a structure of amfc (AMF code) and mtmsi (temporal mobile subscriber identifier).

The smsDeregistration type represents a notification from ESMS with regard to UE deregistration. The attributes of this type include timeStamp, cellClobalId, tempUeId.

The receivedSMSSubscription type represents a subscription to notifications about mobile originating messages intended to the mobile edge application. The attributes of this type are as follow:

- callbackReference as defined above;
- filterCriteriaMoSms – List of filter criteria for the subscription included both in the request and in response. It is a structure of appInsId (identifier of the application instance), plmn (structure of mnc, mcc and cellId as described above), and expiryDeadline.

The moSMSNotification type represents a notification from ESMS about receipt of a mobile originating message intended for the mobile edge application. The attributes of this type include timeStamp, receiverURI (indicates the address of the mobile edge application to which the message is sent), cellGlobalID, tempUeId, and message, which is the received mobile originating message.

The mtSMS type represents the information about mobile terminating message sent by a mobile edge application. The attributes of this type are as follows:

- smsReceiver – URI indicated the address to which the SMS has to be sent;
- smsSender – If preset, it indicates the name of the mobile edge application which sends the message;
- message – The content to be sent in SMS;
- timeStamp.

The messageDeliverySubscription type represents a subscription to message delivery status from ESMS. The attributes of this type are as follows:

- callbackReference as defined above;
- filterCriteriaSmsDelivery – List of filtering criteria for the subscription. It is a structure of appInsId, plmn, cellId, messageID, deliveryStatus. The messageID identifies the sent message. The deliveryStatus is of enumeration type indicating deliveredToNetwork (report that all message parts have been sent to the network), deliveryUncertain (in case of message handoff to another network), deliveryImpossible (unsuccessful delivery), messageWaiting (the message is still queued for delivery), and deliveredToUe (report that all message parts have been delivered to the UE);
- expiryDeadline.

The messageDeliveryNotification type represents a notification from ESMS with regard of message delivery status. The notification is sent by the ESMS to inform the mobile edge application about the delivery status of the message sent. The attributes of this type include timeStamp, cellGlobalIdentity, tempUeId, messagID, and message delivery status.

5. API definition

This section defines the resources and operations of the ESMS APIs.

Fig. 7 illustrates the resource URI structure of this API.

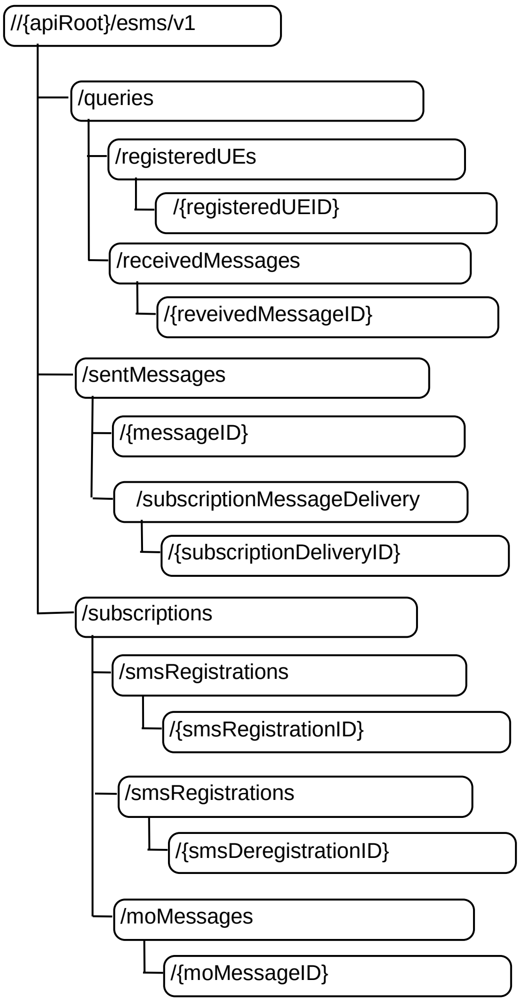

Fig. 7. Structure of resources supported by ESMS

All resource URIs of ESMS API have the following root:

{apiRoot}/{esms}/{apiVersion}/

The leading two are discovered using the service registry. It includes the scheme ("http" or "https"), host and optional port, and an optional prefix string. The "apiVersion" is set for example to "v1". The API supports HTTP over TLS (HTTPS). All resource URIs are defined relative to the above root URI.

The content format of JSON is supported. The JSON format is signalled by the content type "application/json".

The registeredUEs resource contains information about all UEs Registered for SMS, and the registeredUEID resource represents a particular UE registered for SMS over NAS. The receivedMessages resource contains information about all received messages by mobile edge applications, and the receivedMessageID resource represents a particular received mobile originating message.

The sentMessages resource contains information about all mobile terminating messages sent by applications, and the messageID resource represents a particular mobile terminating message sent by an application. The subscriptionMessageDelivery resource contains information about all subscriptions for notifications about message delivery, and the subscriptionDeliveryID resource represents a particular subscription for notifications about message delivery.

The smsRegistrations resource is a container for smsRegistrationID resources each of which represent an existing subscription for UE registration for SMS. The smsDeregistrations resource is a container for smsDeregistrationID resources each of which represent an existing subscription for UE deregistration with SMS. The moMessages is a container for moMessageID resources each of which represents an existing subscription for received mobile originating messages intended for mobile edge applications.

The container resources support HTTP methods POST and GET, while the leaf resources support HTTP methods GET, PUT, and DELETE.

6. Service feasibility

The feasibility of the proposed ESMS is illustrated by modelling the messaging status as seen by the UE and by the service. Both messaging models supported by the UE and ESMS have to be synchronized, i.e. they have to expose equivalent behaviour.

Fig. 8 shows a simplified model of the messaging status for mobile originating message sent to a mobile edge application. The model reflects the UE view.

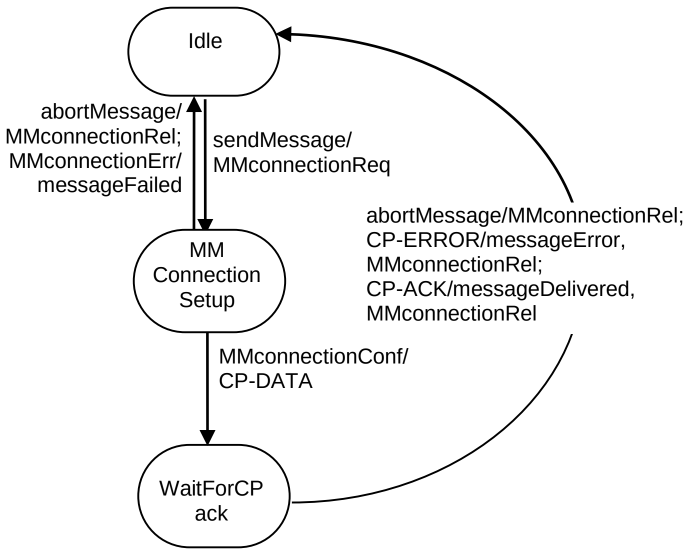

Fig. 8. Model representing the status of a mobile originating message supported by UE

In Idle state, there is no message to be sent. When the subscriber' application sends a message, the UE requests establishment of MM (Mobility Management) connection. In MMConnectionSetup state, a MM connection is setup, and upon receipt of confirmation, the UE sends the actual message as CP-DATA. In WaitForCP-ack state, the UE waits for acknowledgement. The UE informs the subscriber's application about message delivery status when an acknowledgement is received, message transfer fails, or MM connection setup fails. It is the subscriber's application responsibility to implement message retransmission. If the subscriber's application aborts the message sending, the UE requests MM connection release.

A simplified messaging status model for mobile originating text message sent to a mobile edge application supported by the ESMS is shown in Fig. 9.

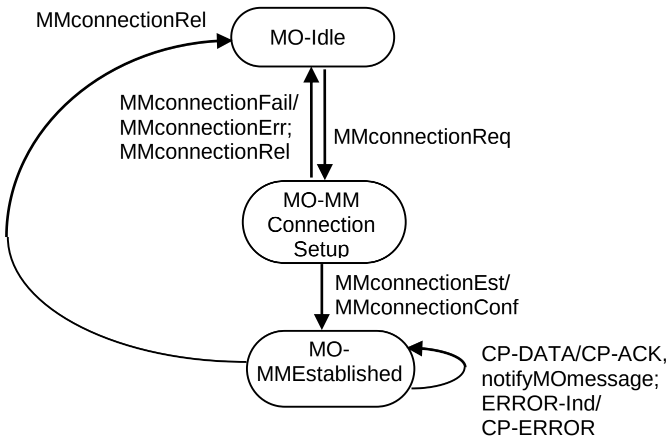

Fig. 9. Model representing the status of a mobile originating message supported by the service

In MO-Idle state, there is no message received from UE. The state is leaved on receiving a request for establishment of MM connection. In MO-MMConnectionSetup state, a MM connection is setup. In MO-MMEstablished state, a MM connection is established and the ESMS waits for CP-DATA that transfers actual message. Upon receiving CP-DATA the ESMS sends an acknowledgement and notifies the mobile edge application about incoming text message.

Both models have to expose equivalent behaviour, i.e., they have to be synchronized.

In order to prove mathematically that both models expose equivalent behaviour; we formalize the models' description. The notion of Labelled Transition System (LTS) is used to describe each model. The LTS provides a generic semantic model that can be used to capture operational semantics of concurrent processes and it is usually used in automatic software analysis based on model testing and checking [17, 18].

By $T_{\text{AppMO}}= (S_{\text{AppMO}}, \text{Act}_{\text{AppMO}}, \rightarrow_{\text{AppMO}}, s_0^{\text{AppMO}})$ is denoted an LTS, representing the model of the mobile originating message status supported by UE, where:

- $S_{\text{AppMO}} = \{\text{Idle}[s_1^{\text{AO}}], \text{MMConnectionSetup}[s_2^{\text{AO}}], \text{WaitForCPack}[s_3^{\text{AO}}]\}$;

- $\text{Act}_{\text{AppMO}} = \{\text{sendMessage}[t_1^{\text{AO}}], \text{MMconnectionConf}[t_2^{\text{AO}}], \text{CP-ACK}[t_3^{\text{AO}}], \text{CP-Error}[t_4^{\text{AO}}], \text{abortMessage}[t_5^{\text{AO}}], \text{MMconnectionErr}[t_6^{\text{AO}}]\}$;

- $\rightarrow_{\text{AppMO}}$ = {(s_1^{AO} t_1^{AO} s_2^{AO}), (s_2^{AO} t_2^{AO} s_3^{AO}), (s_3^{AO} t_3^{AO} s_1^{AO}), (s_3^{AO} t_4^{AO} s_1^{AO}), (s_3^{AO} t_5^{AO} s_1^{AO}), (s_2^{AO} t_5^{AO} s_1^{AO}), (s_2^{AO} t_6^{AO} s_1^{AO})};

- s_0^{AppMO} = { s_1^{AO} }.

By T_{SMO} = (S_{SMO}, Act$_{\text{SMO}}$ $\rightarrow_{\text{SMO}}$, s_0^{SMO}) is denoted an LTS, representing the model of the mobile originating message status supported by the ESMS, where:

- S_{SMO} = {MO-Idle[s_1^{SO}], MO-MMconnectionSetup[s_2^{SO}], MO-MMEstablished[s_3^{SO}]};

- Act$_{\text{SMO}}$ = {sendMessage[t_1^{SO}], MMconnectionConf[t_2^{ST}], CP-ACK[t_3^{SO}], MMconnectionFail[t_4^{SO}], MMconnectionRel[t_5^{SO}], ERROR-Ind[t_6^{SO}]};

- $\rightarrow_{\text{SMO}}$= {(s_1^{SO} t_1^{SO} s_2^{SO}), (s_2^{SO} t_2^{SO} s_3^{SO}), (s_3^{SO} t_3^{SO} s_3^{SO}), (s_2^{SO} t_4^{SO} s_1^{SO}), (s_3^{SO} t_5^{SO} s_1^{SO}), (s_3^{SO} t_6^{SO} s_3^{SO}), (s_2^{SO} t_5^{SO} s_1^{SO})};

- s_0^{SMO} = { s_1^{SO} }.

One of the techniques used to reason on concurrent processes is so called bisimilarity. Bisimilarity deals with task regarding behavior of the processes and more specifically about their equivalence. Two LTSs expose equivalent behaviour if they are bisimilar, i.e., one LTS displays a result and the other LTS displays the same result.

To prove that two LTS are bisimilar, it is necessary to identify a relation between their states, such as for each transition from a state in one LTS there is the respective transition from the corresponding state in the other LTS. Bisimilarity, also known as bisimulation, is of two types – strong and weak, where the strong bi-simulation requires equivalency for each and every transition and action. The weak bi-simulation is a relationship that takes into account only transitions and actions that are observable, i.e., it disregards the internal ones nevertheless they might hold essential part of the LTS structure. The mathematical definitions of strong and weak bisimulation may be found in [19, 20].

Proposition. T_{AppMO} and T_{SMO} are weakly bisimilar.

Proof: By U_{MOAppS} it is denoted a relation between the states of T_{AppMO}, and T_{SMO}, where U_{MOAppS}= {(s_1^{AO} , s_1^{SO}), (s_2^{AO} , s_2^{SO}), (s_3^{AO} , s_3^{SO})}. Then the following transitions for the states in U_{MOAppS} are identified:

1. The subscriber's application sends a mobile originating message to a mobile edge application and the UE initiates MM connection establishment procedure:

$$\text{For } (s_1^{AO}\ t_1^{AO}\ s_2^{AO}) \ \exists\ (s_1^{SO}\ t_1^{SO}\ s_2^{SO}).$$

2. The MM connection setup fails and the UE informs the subscriber's application:

$$\text{For } (s_2^{AO}\ t_6^{AO}\ s_1^{AO}) \ \exists\ (s_2^{SO}\ t_4^{SO}\ s_1^{SO}).$$

3. The MM connection is established, the UE sends the actual message using NAS signalling as CP-DATA, and waits for acknowledgement:

$$\text{For } (s_2^{AO}\ t_2^{AO}\ s_3^{AO}) \ \exists\ (s_2^{SO}\ t_2^{SO}\ s_3^{SO}).$$

4. The message is successfully delivered, the MM connection is released and the mobile edge application is notified about incoming message:

$$\text{For } (s_3^{AO}\ t_3^{AO}\ s_1^{AO}) \ \exists\ (s_3^{SO}\ t_3^{SO}\ s_3^{SO}),\ (s_3^{SO}\ t_5^{SO}\ s_1^{SO}).$$

5. There is an error in message transfer and the subscriber's application is notified and the MM connection is released:

$$\text{For } (s_3^{AO}\ t_4^{AO}\ s_1^{AO}) \ \exists\ (s_3^{SO}\ t_6^{SO}\ s_3^{SO}),\ (s_3^{SO}\ t_5^{SO}\ s_1^{SO}).$$

6. The subscriber's application aborts the message sending:

$$\text{For: } (s_2^{AO}\ t_5^{AO}\ s_1^{AO}) \sqcup (s_3^{AO}\ t_5^{AO}\ s_1^{AO}) \ \exists\ (s_2^{SO}\ t_5^{SO}\ s_1^{SO}) \sqcup (s_3^{SO}\ t_5^{SO}\ s_1^{SO}).$$

Therefore T_{AppMO} and T_{SMO} are weakly bisimilar.■

The LTS computational model can be used to describe both concurrent processes' implementations and their expected behavior specifications. Therefore, the bisimilarity concept can be regarded as a formal criterion to verify the correctness of the implementation with respect to the specification, when both are represented in the LTS model. As noted in [19], there exists a tool for automatic verification based on weak bi-similarity [21], but it seems to integrate formulae in the tool sets in a limited manner.

In addition to formal verification of the proposed ESMS, it is necessary to assess the benefits of service deployment and to evaluate the improvements as perceived by the user. The service key performance indicators affect the user perception and include latency, energy efficiency, throughput loss rate, etc., [22]. In the next section, we estimate theoretically the latency introduces by the proposed mobile edge service as it is an essential key performance indicator for mission critical services. The latency refers to the delay between any request and a consequent response.

7. Performance analysis

The round-trip time is defined as the time taken for a request to go from the source to the destination, be replied and travel back [22]. The latency introduced by the proposed ESMS may be estimated by summation of the processing times (β) in each node along the signalling path and the time required for message transfer (α).

Fig. 10 shows the signalling exchange in case of mobile terminating message.

Radio node in 5G (gNB) consists of one Centralized Unit (gNB-CU) which controls the operations of one or more Distributed Units (gNB-DU) which host lower radio protocols. The protocol between gNB-CU and gNB-DU is F1 Application Protocol (F1AP), while the protocol between gNB-CU and AMF is Next Generation Application Protocol (NGAP). The Radio Resource Control (RRC) connection between the UE and gNB has to be reconfigured in order to transfer the message over the air interface. As the AMF, ESMS and mobile edge applications are MEC hosted on the same node, the interface between them is considered to be internal. In Fig.10, short notations of signalling message are given in brackets.

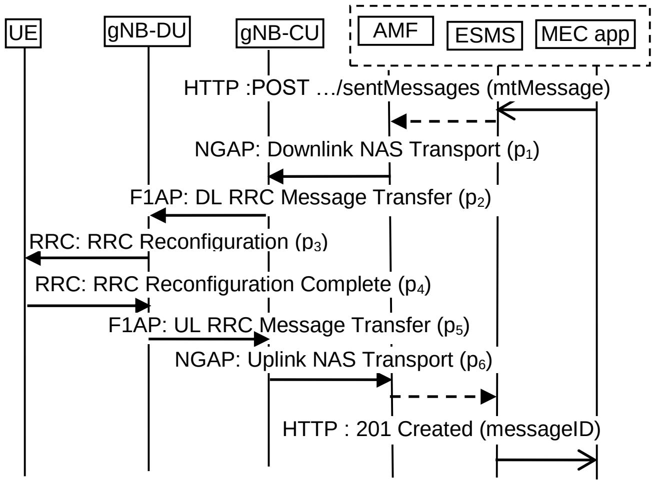

Fig. 10. Signalling flow for mobile terminating message

So, the latency introduced for mobile terminating message transfer may be evaluated as follows:

(1) $$L_{mtMessage}=\beta_{MEC}(p_1)+\beta_{AMF}(p_1)+\alpha_{NGAP}+\beta_{gNB\text{-}CU}(p_2)+\alpha_{F1AP}+ \\ +\beta_{gNB\text{-}DU}(p_3)+\alpha_{RRC}+\beta_{UE}(p_4)+\alpha_{RRC}+\beta_{gNB\text{-}DU}(p_5)+\alpha_{F1AP}+ \\ +\beta_{gNB\text{-}CU}(p_6)+\alpha_{NGAP}+\beta_{AMF}(p_6)+\beta_{MEC}(p_6).$$

As to [23] and [24] the processing time in gNB ($\beta_{gNB\text{-}CU}$ and $\beta_{gNB\text{-}DU}$) and in UE (β_{UE}) is about 0.3 ms and the time required for message transfer is 0.1 ms (feasibility studies for 5G New Radio and Next Generation Radio Access Network are not yet available). By α_{F1AP} is denoted the message transfer time between gNB-CU and gNB-DU. By α_{NGAP}, is denoted the message transfer time between gNB-CU and AMF, and α_{RRC} denotes the message transfer time over the radio interface. The time required for the message transfer between entities in the MEC host is negligible and may be omitted. The time cost for MEC execution is given as

(2) $$\beta_{MEC}=D.X/f,$$

where *D* is the information of the task input-data size *D* (in bits), *X* is the computation workload/intensity (in CPU cycles per bit) and *f* is the CPU frequency of the MEC server. The MEC parameters used are as follows: input data size D=1000 bits, the computational workflow/intensity X=1000 cycles per 1 bit, and the CPU frequency of MEC server f=1.1 GHz [25, 26].

Having this values,

(3) $$L_{mtMessage} = 7\beta + 6\alpha + 2\beta_{MEC} = 2.7 \text{ ms} + 2\beta_{MEC} = \\ = 2.7 \text{ ms} +0.91 \text{ ms} = 3.61 \text{ ms}.$$

The experimental data in [27], given by the authors provide latency evaluation for MEC in form of Round Trip Time (RTT) with fiber-wireless access. While the original data are ping-pong based, we add a summand in order to reflect the processing time in the MEC host. We use these empirical data to estimate the related theoretical distributions. Following the same distributions (Normal, LogNormal, Gamma, and Weibull) applied to MEC deployment with distributed core functions for the flow shown in Fig. 10 we fit the Cumulative Distributed Function (CDF) like the one shown in black in Fig. 11.

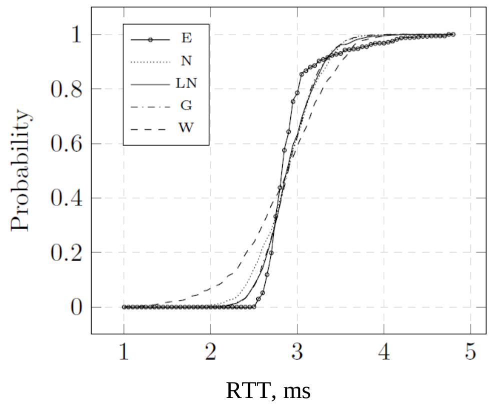

Fig. 11. Empirical and theoretical CDFs of RTT as key performance indicator for latency in provisioning radio network information in case of MEC and C-RAN co-location deployment

The notations in Fig. 11 are as follows: E is the Empirical CDF of RTT fitted by different distributions; N – Normal (μ = 2.906545; σ = 0.347701); LN – LogNormal (μ = 1.0608995; σ = 0.1059133); G – Gamma (α = 82.60575; β = 28.42070); W – Weibull (k = 6.446237; λ = 3.073411).

The comparison of the above fits to the empirical distribution is shown in Table 1 where the lower statistics the better, i.e., the fit's "discrepancy" is lower.

In Table 1 the abbreviations for the statistics are as follow: KS – Kolmogorov-Smirnov; CvM – Cramer-von Mises; AD – Anderson-Darling. All the three types put on top the LogNorm distribution as best fit to empirical data when compared to the other three distributions.

Table 1. Goodness-of-Fit Statistics

Distribution	KS	CvM	AD
LogNormal	0.1854354	8.6521321	50.1258819
Gamma	0.1951410	9.7758410	55.8919130
Normal	0.2130025	12.1537082	67.9544216
Weibull	0.2454756	19.4274263	103.2981935

The distribution could be used as a compact form of model for latency in the process of service/application development, testing or software refactoring.

8. Conclusion

In this paper, we propose an approach to open access to messaging service at the network edge. A new mobile edge messaging service is described by typical use cases illustrating the capabilities of UE to send a short message to a mobile edge application, and capabilities of a mobile edge application to send a short message to UE, including capabilities for tracking message delivery status. The service resources are modelled in a tree structure and can be manipulated by simple operations. The service interfaces follow the REST architectural style, and the behavioural models of

MEC platform and the mobile edge application logic concern some implementation details.

The latency introduced by the service is estimated by modelling the signalling between UE and network entities involved in message transfer.

The service may support mission critical communications, as the latency introduced is very low. The service provides message transfer in near to real time. It is not based on principle “store and forward”, as the in case of mobile originating short message the mission critical application has to be available, and for mobile terminating short message it is up to the mission critical application logic to retransmit the message in case the delivery fails.

Acknowledgement**:** The research is conducted under the grant of project DH07/10-2016, funded by Bulgarian National Science Fund, Ministry of Education and Science.

References

1. M u k h e r j e e, S., C. B e a r d. A Framework for Ultra-Reliable Low Latency Mission-Critical Communication. – In: Proc. of Wireless Telecommunications Symposium (WTS’17)**,** Chicago, IL, 2017, pp. 1-5.
2. P e t r o v, V., et al. Achieving End-to-End Reliability of Mission-Critical Traffic in Softwarized 5G Networks. – IEEE Journal on Selected Areas in Communications, Vol. **36**, March 2018, No 3, pp. 485-501.
3. K h o s h n e v i s a n, M., V. J o s e p h, P. G u p t a, F. M e s h k a t i, R. P r a k a s h, P. T i n n a k o r n s r i s u p h a p. 5G Industrial Networks with CoMP for URLLC and Time Sensitive Network Architecture. – IEEE Journal on Selected Areas in Communications, Vol. **37**, April 2019, No 4, pp. 947-959.
4. P o r a m b a g e, P., J. O k w u i b e, M. L i y a n a g e, M. Y l i a n t t i l a, T. T a l e b. Survey on Multi-Access Edge Computing for Internet of Things Realization. – IEEE Communications Surveys & Tutorials, Vol. **20**, Fourthquarter 2018, No 4, pp. 2961-2991.
5. B e k k o u c h e, O., T. T a l e b, M. B a g a a. – UAVs Traffic Control Based on Multi-Access Edge Computing. – In: Proc. of IEEE Global Communications Conference (GLOBECOM’18), Abu Dhabi, United Arab Emirates, 2018, pp. 1-6.
6. M a r k a k i s, E. K., et al. Efficient Next Generation Emergency Communications over Multi-Access Edge Computing. – IEEE Communications Magazine, Vol. **55**, November 2017, No 11, pp. 92-97.
7. Y a n g, J., X. J i. An Improved Inter-Domain Handover Scheme Based on Bidirectional Cooperative Relay. – Cybernetics and Information Technologies, Vol. **13**, 2013, No 4, pp.127-138.
8. K u m a r, S. A., P. I l a n g o, G. H. D i n e s h. A Modified LEACH Protocol for Increasing Lifetime of the Wireless Sensor Network. – Cybernetics and Information Technologies, Vol. **16**, 2016, No 3, pp. 154-164.
9. S h a r m a, S., et al. Secure Authentication Protocol for 5G Enabled IoT Network. – In: Proc. of 5th International Conference on Parallel, Distributed and Grid Computing (PDGC’18), Solan Himachal Pradesh, India, 2018, pp. 621-626.
10. J o v e r, R. P. The Current State of Affairs in 5G Security and the Remaining Security Challenges. – arXiv:1904.08394c2 [cs. CR\ 18 April 2019, pp. 1-8. **https://arxiv.org/pdf/1904.08394.pdf**
11. 3GPP TS 29.199-4 3rd Generation Partnership Project. Technical Specification Group Core Network and Terminals. Open Service Access (OSA). Parlay X Web Services. Part 4: Short Messaging Release 9, V9.0.0, 2009.
12. G u i s t, F., et al. MEC Deployments in 4G and Evolution towards 5G. – ETSI White Paper, No 24, 2018.
13. K e k k i, S., et al. MEC in 5G Networks. ETSI White Paper, No 28, 2018.

14. ETSI GS MEC 003 Mobile Edge Computing (MEC). Framework and Reference Architecture, V1.1.1, 2016.
15. 3GPP TR 22.824 Technical Specification Group Services and System Aspects. Feasibility Study on 5G Message Service for MIoT. Stage 1, Release 16, V16.0.0, 2018.
16. 3GPP TS 38.413 Technical Specification Group Radio Access Networks. NG-RAN. NG Application Protocol (NGAP), Release 18, V15.2.0, 2018.
17. Z h a n g, K., T. L i u, D. C h e n g. Observability of Finite Labeled Transition Systems. – IEEE Transactions on Automatic Control, Vol. **63**, June 2018, No 6, pp. 1501-1512.
18. B e e k, M. H., F. D a m i a n i, S. G n e s i, F. M a z z a n t i, L. P a o l i n i. On the Expressiveness of Modal Transition Systems with Variability Constraints. – Science of Computer Programming, Vol. **169**, 2019, pp. 1-17. 10.1016/j.scico.2018.09.006.
19. E s c r i g, D. de F., J. J. A. K e i r e n, T. A. C. W i l l e m s e. Games for Bisimulations and Abstraction. – Local Methods in Computer Science, Vol. **13**, 2017, Issue 4:15, pp.1-40.
20. Y a n g, X., J. K a t o e n, H. L i n, G. L i u, H. W u. Branching Bisimulation and Concurrent Object Verification. – In: Proc. of 48th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN), Luxembourg City, 2018, pp. 267-278.
21. G i b s o n-R o b i n s o n, T., P. A r m s t r o n g, A. B o u l g a k o v, A. W. R o s c o e. FDR3: A Parallel Refinement Checker for CSP. – International Journal on Software Tools for Technology Transfer. Vol. **2**, 2016, Issue 18, pp. 149-167.
22. ETSI GS MEC-IEG 006 Mobile Edge Computing. Market Acceleration. MEC Metrics Best Practice and Guidelines, V1.1.1, 2017.
23. ITU-R M.2410-0 Minimum Requirements Related to Technical Performance for IMT-2020 Radio Interface(S), 2017.
24. B e r t e n y i, B., et al. 5G NR Radio Interface. – Journal of ICT Standardization, Vol. **6**, May 2018, Issue 1&2, pp. 31-58.
25. C a t t a n e o, G., F. G i u s t, C. M e a n i, D. M u n a r e t t o, P. P a g l i e r a n i. Deploying CPU-Intensive Applications on MEC in NFV Systems: The Immersive Video Use Case. – Computers, Vol. **55**, 2018, Issue 7, pp.1-20. DOI:10.3390/computers7040055.
26. F i l i p p o u, M. C., D. S a b e l l a, V. R i c c o b e n e. Flexible MEC Service Consumption through Edge Host Zoning in 5G Networks. – Cornell University, Computer Science, Distributed, Parallel and Cluster Computing, 2019, arXiv:1903.01794 [cs. DC].
27. L i u, J., G. S h o u, Y. L i u, Y. H u, Z. G u o. Performance Evaluation of Integrated Multi-Access Edge Computing and Fiber-Wireless Access Networks. – IEEE Access, Vol. **6**, 2018, pp. 30269-30279.

Received: 17.09.2019; Second Version: 27.10.2019; Accepted: 10.11.2019