An Approach to Open Access to Multimedia Multicast Services at the Network Edge

Ivaylo I. Atanasov, Evelina N. Pencheva, Vladislav G. Vladislavov, and Ventsislav G. Trifonov
Technical University of Sofia, Sofia, 1756, Bulgaria
Email: {iia, enp, vgt}@tu-sofia.bg, vladislav.vladislavov@balkantel.net

Abstract —The multicast services are effective solution for content delivery to groups of subscribers. Typical multicast applications include distribution of music, video, news, advertisements, weather information, software and other data. The multicast minimizes the network traffic in comparison to unicast as the content is accessible for all users subscribed for the service simultaneously. The efforts on further evolved Multimedia Broadcast Multicast Service (MBMS) in fifth generation (5G) mobile networks enable more optimal usage of network capacity and better quality of experience for end users. The MBMS requires deployment of additional functions in the core network, which from technical point of view introduces latency. There are use case scenarios where the latency has to be minimized, e.g. for mission critical applications. In this paper, we present an approach to provide MBMS functionality at the network edge using Multi-access Edge Computing (MEC) technology. MEC reduces latency as it distributes the cloud capabilities closer to the place where they are needed. The vicinity to end users enables more efficient usage of network resources and higher data rates in addition to latency reduction. Among the others, the deployment of MBMS at the network edge enables location based content delivery, e.g. broadband multicasting of sport events at stadiums. A new MEC service is proposed, which enables third party applications to manage multicast sessions. Using the service, a MEC application may create a new multicast session, delete an existing multicast session, get information about the multicast session, and manage user participation in a multicast session. A MEC application may subscribe to receive notifications about multicast session events and may be notified when such events occur. The proposed MEC service follows the Representational State Format (REST) architectural style. We present the service data model which describes the data type and the structure of resource, as well as methods supported by the resources. To illustrate the approach feasibility, we model the multicast session state as seen by the network and by MEC application, and prove formally that both models are synchronized, i.e. they expose equivalent behavior. We also discuss service performance in terms of introduced latency. The latency "injected" by the service is evaluated theoretically through signaling for multicast session management. Based on published empirical data for MEC performance we determine the distributions which fit best to these data applied to the proposed MEC service.

Index Terms—Fifth generation mobile system, multi-access edge computing, multicast services, Representational State Transfer, state models, latency

I. Introduction

Multicast is an efficient technology to deliver the same content to a group of people or devices being served with guaranteed quality of service which addresses the requirements of low energy consumption and costs. Multicast service uses unidirectional point to multi point schemes for transmission of multimedia content (e.g. audio, video, text or picture) and it is accessible only for multicast group members that are subscribed to the service [1]. It appears to be cost effective and high quality mechanism for clients who consume the same data such as software updates, emergency messages or media content [2], [3]. The multicast/broadcast technology is a key component of fifth generation (5G) mobile system for both real-time and non-real-time services. Examples of real-time multicast scenario include television or live contents with large audience delivery, augmented and virtual reality, push-to-talk/video services, and automotive applications including autonomous driving information, infotainment, and safety applications. Example use cases of non-real-time services include software and firmware upgrades, common control messages to large number of devices, disaster and emergency alerts, news etc. [4], [5].

Multicast services will be implemented in 5G in a flexible and dynamic way for content delivery necessary for optimal system performance [6]-[9]. The standardization work on Multicast/Broadcast Multimedia Service (MBMS) is ongoing and the studies are focused at defining of framework between radio access network and core network to support MBMS in different vertical segments such as automotive applications, massive machine type communications and mission critical communications. The 5G Xcast project aims to develop architecture that enables harmonized context delivery between different networks [10]. Multicast service deployment in 5G will follow the 4G Multicast/Broadcast Multimedia Service (MBMS) architecture with enhanced software functions to support service performance characteristics such as speed, latency, and capacity. The flexible reconfigurable architectural design of the 5G network is one of the main technical challenges to

Manuscript received August 16, 2019; revised March 6, 2020.
This work was supported by the Bulgarian National Science Fund, Ministry of Education and Science under Grant No. DH07/10-2016.
Corresponding author email: enp@tu-sofia.bg.
doi:10.12720/jcm.15.4.309-317

provide reliable communications with guaranteed latency for multimedia multicast services. One of the ways to face this challenge is to distribute core network functions integrated with cloud intelligence at the network edge. Multi-access Edge Computing (MEC), former Mobile Edge Computing, which is regarded as an essential ingredient of 5G networks, aims to provide computing and storage capabilities at the network edge where these capabilities are needed [11], [12]. The MEC vicinity to end users contributes to latency minimization and optimization of network resource usage. MEC use cases include customized local content distribution, location services, video analytics, augmented reality, Internet of Things and data cashing. MEC enables third party applications to benefit from being in close proximity to end users and from tapping into local content and real-time information about radio access network conditions. A multicast service received by the client, involves one or more consecutive multicast sessions e.g. a multimedia streaming or multiple messages transmitted on long term basis such as multicasting of sport results.

In this paper, we propose a service oriented approach to implement multicast service using MEC technology. The deployment of multicast service at the network edge enables location-based content delivery, e.g. broadband multicasting of sport events at stadiums, reliable control message delivery for great number of devices without causing network congestion, and efficient usage of network resources for mission critical applications with very straighten requirements on latency, throughput and reliability.

We explore the MEC capabilities to enable third party applications deployed at the network edge to manage a multicast multimedia sessions, its members and media streams, and to obtain multicast session information.

The rest of the paper presents the proposed approach to opening the multicast session management functions by description of typical use cases, data model and application programming interfaces. In order to depict the approach essence, multicast session state models supported by MEC application and network are provided. The latency injected by the proposed service is evaluated theoretically.

II. Service Deployment

The proposed Multicast Session Management Service (MSMS) can be useful in MEC deployment scenarios with distributed core network components. This MEC deployment type suits well for mission critical communications and massive machine type communications where there is no need for communications with operator's core network entities. It is usually applied for mission critical enterprise zones, public security, emergency services, and industrial control. In this scenario, MEC applications can be hosted with 5G virtualized core network functions (VNF) sharing the same network function virtualization platform [11].

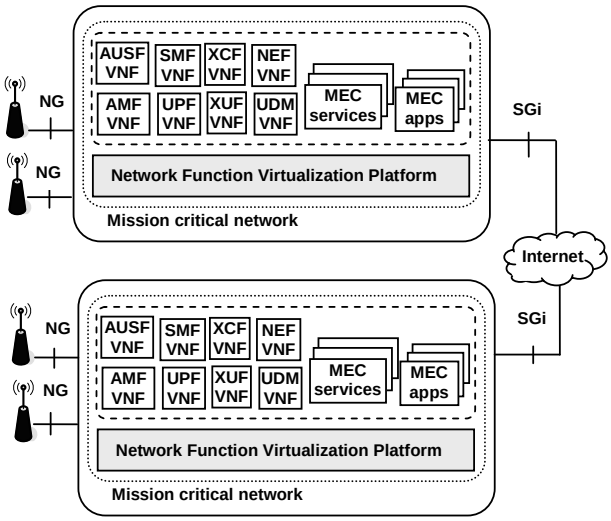

Fig. 1. Deployment of MEC with multicast functionality and distributed core in 5G

Further, distributed core network functions enable communications with customized quality of service and configurable features e.g. tailored to enterprise requirements. Fig. 1 shows the MEC integration with distributed core functions adopting 5G Xcast approach.

The distributed 5G core network tailored to enterprise requirements may consist of the following functions [14]:

- Authentication Server Functions (AUSF) for authentication of User equipments (UEs) during registration;
- Access and Mobility Management Function (AMF) for registration, access, reachability, connection and mobility management of UEs;
- Session Management Function (SMF) for session establishment, modification and release, and for policies;
- User Plane Function (UPF) for packet routing and forwarding, and policy enforcement;
- Unified Data Management (UDM) for access to subscription data;
- Network Exposure Function (NEF) for exposure of events and capabilities.

The support of multicast service requires new functionalities which may be implemented by one or more separate network functions as to 5G-Xcast project [10]. The functionality of Broadcast Multicast Service Center may be split into a control plane part (XCF) and a user plane part (XUF). The MEC platform, which provides MEC services exposed to MEC applications, takes the place of application function.

III. Overal Service Description

The MSMS allows MEC applications to manage initiation/modification/termination of multicast sessions, to control user participation in sessions, and to retrieve information about multicast sessions. The interaction

between MSMS and MEC applications follows two mechanisms:

- "Request-Response" mechanism, where the MSMS is requested by MEC application to provide a certain service, e.g. to perform an action or to provide information;
- "Subscribe-Notify" mechanism, where the MEC application subscribes to MSMS to receive notifications of interest.

A MEC application may issue the following request types:

- Requests to create/terminate a multicast session;
- Requests to retrieve information about a multicast session;
- Requests to users to join to/leave a multicast session;
- Requests to retrieve information about multicast session participants.

The MSMS supports the following subscription types:

- Subscriptions for notifications about UE registrations with multicast subscription;
- Subscriptions for notifications about joining/ leaving a multicast session.

The service design follows Representational State Transfer (REST) architectural style for distributed applications. REST takes a resource-based approach for interactions where resources represent entities (physical or logical ones).

A. Multicast Session Subscriptions and Notifications

Fig.2 shows the structure of resources related to different subscription types supported by the MSMS, where **{apiRoot}/msms/v1** is the root followed by all resource URIs (Uniform Resource Identifier) of MSMS application programming interfaces (API).

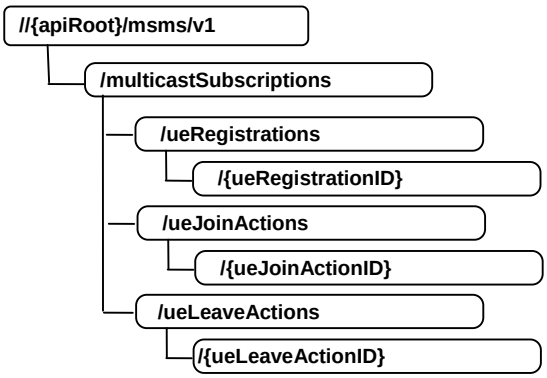

Fig. 2. Structure of resources related to subscriptions for notifications about multicast session events

The **multicastSubscriptions** resource is a container for all subscription types related to MSMS. The **ueRegistrations** resource is used to keep track of active subscriptions to registration events of users subscribed for multicast service. The {**ueRegistrationID**} resource represents a request from a MEC application to be notified about UE registrations. The **ueJoinActions** and **ueLeaveActions** resources contains all subscriptions for respectively UE joining to and leaving multicast sessions, while {**ueJoinActionID**} and {**ueLeaveActionID**} resources represent existing subscriptions of the respective type. Fig.3 illustrates the message flow for subscribing to UE registrations with multicast service subscription.

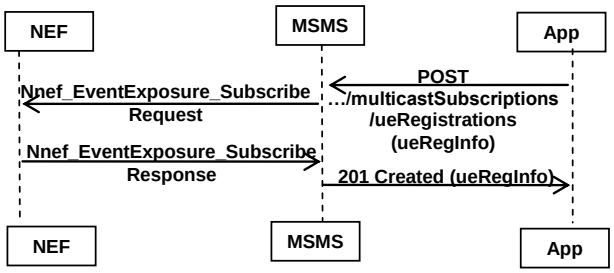

Fig. 3. Flow of subscribing for notifications about UE registrations with multicast service subscription

To receive notifications on registrations of UEs with multicast service subscription, the MEC application sends a POST request with message body containing the **ueRegSubsctiption** data structure. The **ueRegSubsctiption** data structure defines the URI where the application wants to receive notifications and the filter criteria pointing the multicast service of interest. The MSMS in turn invokes **Nnef_EventExposure_Subscribe** operation of the NEF to subscribe to receive an event on UE registration [13]. When the subscription is accepted the NEF returns the subscription correlation ID to the MSMS. The MSMS sends "201 Created" with the URI of the created resource and subscribed UE registration type.

The same template of interaction is followed for creation of other type of subscriptions where the data type exchanged represents data structure specific for the respective subscription type. The **ueJoinSubscription** and **ueLeaveSubcription** data structures in the message body of the POST request for creation a subscription for notifications on UE joining to or leaving multicast session events include the session URI and the URIs of UEs that the application wants to monitor.

The existing subscription may be updated e.g. in case of subscription expiry. Fig.4 shows the message flow for subscription updating.

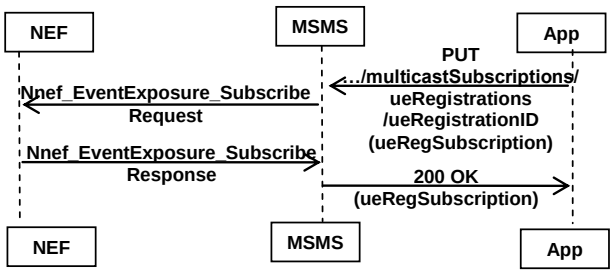

Fig. 4. Flow of updating existing subscription for notifications about UE registrations with multicast service subscription

The MEC application sends a PUT message to the resource representing the corresponding subscription with modified data structure. The MSMS answers with "200 OK" message containing the modified data.

The MEC application may terminate the subscription if it is not interested any more in receiving notifications

about specific events. To do this, the application sends a DELETE request to the resource representing the corresponding subscription, the MSMS invokes **Nnef_EventExposure_Unsubscribe** service operation of NEF and responds to the application with 204 "No Content" (Fig. 5).

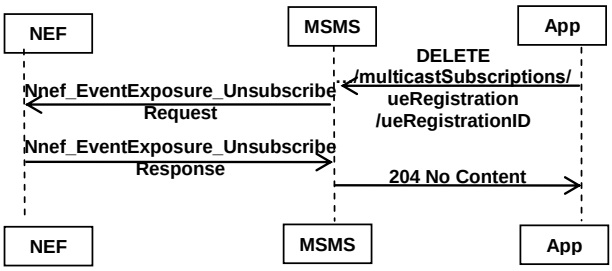

Fig. 5. Flow of termination of existing subscription for notifications about UE registrations with multicast service subscription

Upon occurrence of event of interest, the NEF performs **Nnef_EventExposure_Notify** service operation to report the event to the MSMS [15]. In case of active subscription, the MSMS sends a POST request to the URI provided by the MEC application to notify about the event. The notification contains information about the event.

Fig. 6 shows the flow of receiving notifications on UE registering with multicast service subscription. The **ueRegNotification** data structure contains the event ID, the URI of the UE, and timestamp.

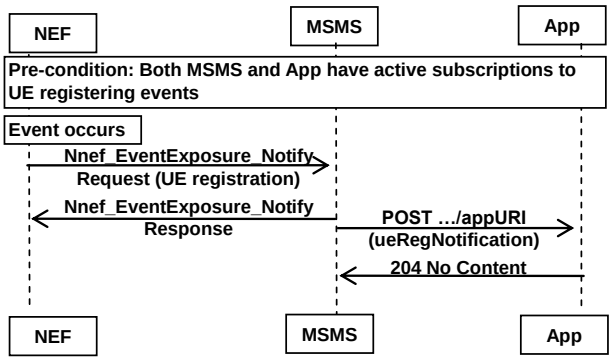

Fig. 6. Flow of MSMS event notification on UE registering with multicast service subscription

The same pattern is applied when the MSMS sends a notification on UE joining to /leaving an existing multicast session. The body of the POST request contains respectively **ueJoinNotification** data structure or **ueLeaveNotification** data structure. These data structures contain the event ID, the URI of the UE, the session URI, and timestamp.

TABLE I: METHODS SUPPORTED BY MBMS SUBCRIPTION RESOURCES

Resource name	Resource URI	HTTP method	Meaning
All subscriptions for multicast sessions	/multicastSubscriptions	GET	Retrieves multicast subscription types
All subscriptions for UE registrations	/multicastSubscriptions /{SubscriptionType}	GET	Retrieves all subscriptions related to multicast session
		POST	Creates new subscription
Existing subscription for UE registrations	/multicastSubscriptions /{SubscriptionType}/{ SubscriptionID}	GET	Retrieves information on the current subscription
		PUT	Modifies the existing subscription
		DELETE	Cancels the existing subscription

Table I summarizes the methods supported by MSMS subscription resources. In Table I, {**SubscriptionType**} may be one of the following: **ueRegistrations, ueJoinActions,** or **ueLeaveActions,** and {**Subscription-ID**} may be {**regSubscriptionID}, {ueJoinActionID),** or **{ueLeaveActionID}.**

B. *Multicast Requests*

Being notified about UE registration with subscription for multicast service, the MEC application may invite the user to join to existing multicast session. For multicast sessions in specific areas, the MEC application may use MEC Location service [16] to be notified when the user enters the area. The multicast sessions and session members are also represented as resources, and Fig.7 shows the respective resource.

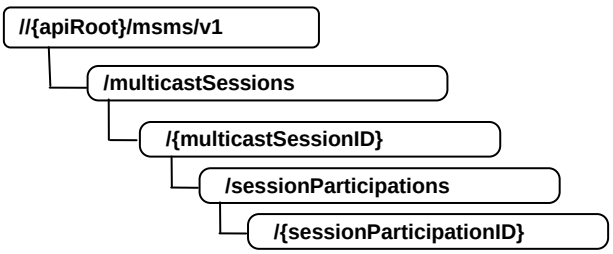

Fig. 7. Resource structure for multicast sessions and their participants

The **multicastSessions** resource is a container for all multicast sessions created by MEC application. The {**multicastSessionID**} resource represents an existing multicast session created by MEC application. The **sessionParticipations** resource contains information about UEs participating in an existing multicast session, and {**sessionParticipationID**} represents an UE participating in the session.

A MEC application can retrieve information for all multicast sessions, information for an existing multicast session, information about participants in an existing multicast session, and information about specific participant in a multicast session.

Fig. 8 shows the flow for retrieving information about an existing multicast session. The application sends a GET request to the resource representing the respective multicast session. The MSMS responds with "200 OK" with message body containing the **sessionInfo**. The **sessionInfo** data structure contains the session name, the minimum bandwidth required for receiving session

multimedia streams, media types, and the duration of the multicast session.

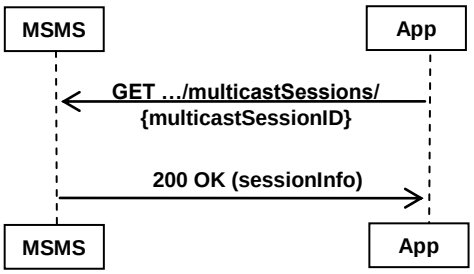

Fig. 8. Flow of retrieving information about existing multicast session

An example of HTTP request for retrieving information about existing multicast session is as follows:

```
GET /exampleAPI/msms/v1/ multicastSessions
/multicastSessionID HTTP/1.1
Host:example.com
Accept:application/json
Content-type:application/json
```

An example of HTTP response is as follows:

```
HTTP/1.1 200 OK
Host:example.com
Accept:application/json
Content-type:application/json
Content-length:164
{ “timestamp”:”2019-08-23T12:15:03.521Z”,
“sessionName”:”FootballMatch”,
“badwidth”:”800Mbps”,
“mediaType”:[“audio”,”video”]
“sessionDuration”:”7200”
}
```

To retrieve information about specific participant in a multicast session, the application sends a GET request to the resource representing the corresponding participant in the multicast session. The MSMS responds with “200 OK” with message body containing the **participantInfo**. The **participantInfo** data structure contains the URI of the UE and its status. The status of participating UE can be one of the following **AskedToJoin** (the user is asked but not joined yet), **Attached** (the user is joined to the session), **Declined** (the user rejects to join) and **AskedTo- Leave** (the user is asked but not leaved yet).

A MEC application can request a multicast session creation. To do this, the application sends a POST request to the **multicastSessions** resource with message body containing **sessionInfo** data structure for the session to be created. The MSMS responds with “201 Created” with the message body containing the data structure specific to that multicast session and the session ID.

To invite a user to join to an existing session, the MEC application sends a POST request to the **sessionParticipations** resource with parent representing the multicast session. The message body of the request contains **userInfo** data structure including URI of the UE. Paging procedure takes place in the network. The paging functionality is exposed by **Nnef_TrafficInfluence** service of NEF [15]. The MSMS invokes the **Nnef_TrafficInfluence_Create** operation on NEF pointing the UE address and the transaction ID. The user may accept or reject the invitation and the NEF uses **Nnef_TrafficInfluence_Notify** operation to notify the MSMS about the traffic event.

Fig. 9 shows the message flow for successful joining of UE to multicast session initiated by MEC application.

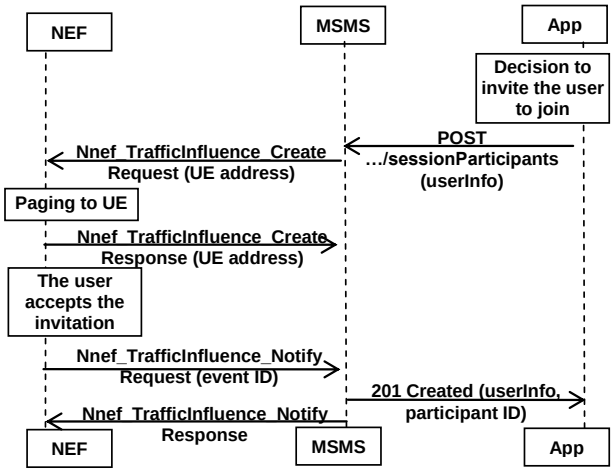

Fig. 9. Flow of application initiated UE joining to multicast session

An example of HTTP request inviting user to join to multicast session looks like the following:

```
POST /exampleAPI/msms/v1/ multicastSessions
/multicastSessionABC/sessionParticipants HTTP/1.1
Host:example.com
Accept:application/json
Content-type:application/json
Content-length:110
{ “timestamp”:”2019-08-23T12:20:00.521Z”,
“sessionAddress”:”225.100.4.13”,
“userURI”:”nai-ceae2e636f6d”
}
```

Similar interaction template occurs when the MEC application decides to invite a user to leave the multicast session. In this case, the application sends a DELETE request to the **sessionParticipationID** resource, representing the corresponding user. The MSMS invokes the **Nnef_TrafficInfluence_Delete** operation on NEF pointing the transaction ID. The MSMS is notified about the result by **Nnef_TrafficInfluence_Notify** operation.

Table II summarizes the methods supported MSMS request resources.

TABLE II: METHODS SUPPORTED BY MBMS REQUEST RESOURCES

Resource name	Resource URI	HTTP method	Meaning
All multicast sessions	/multicastSessions	GET	Retrieves list of multicast sessions
		POST	Creates a new session
An existing multicast session	/multicastSessions /multicastSessionID	GET	Retrieves information about existing session
		POST	Modifies multicast session
		DELETE	Deletes multicast session

All participants in particular multicast session	/multicastSessions /multicastSessionID /sessionParticipants	GET	Retrieves list of all session participants
		POST	Invites user to join the session
An existing participant in particular multicast session	/multicastSessions /multicastSessionID /sessionParticipants /sessionParticipantID	GET	Retrieves information about session participant
		DELETE	Invites user to leave the session

IV. State Models

The application supports the state of the multimedia session. The network also supports the multicast service bearer context state model which contains all information describing the multicast bearer service. Both states have to be synchronized.

Fig. 10 shows the multicast bearer context state model. In **Standby** state no bearer plane resources are required in the network for multicast data transfer. The network maintains this state as long as there is no a corresponding multicast session ongoing. In **Active** state, bearer plane resources are required for multicast data transfer in the network. The network maintains this state for the duration of a corresponding multicast session. Being in **Active** state, the bearer context may be modified.

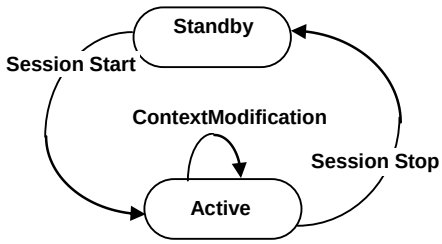

Fig. 10. Multicast bearer context state model

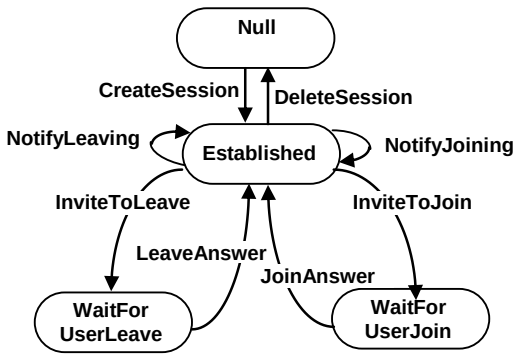

Fig. 11. Multimedia session state model supported by MEC application

Fig. 11 illustrates the proposed simplified multimedia session state model supported by MEC application. In **Null** state, the multicast session is not established. In **Established** state, the multicast session exists. Being in **Established** state, the application may be notified about user joining to or leave from multicast session. In **WaitForUserJoin** state, the application waits a user to join to an existing multicast session. In **WaitForUser-Leave** state, the application waits the user to leave the session.

To prove mathematically that both models are synchronized i.e. expose equivalent behavior we formally describe the models as Labeled Transition Systems.

The multicast bearer context state model supported by the network is formally described as $T^{bc} = (S^{bc}, Inp^{bc}, \rightarrow^{bc}, s_0^{bc})$, where

$S^{bc} = \{Standby\ [s_1^{bc})], Active\ [s_2^{bc}]\}$;

$Act^{bc} = \{SessionStart\ [t_1^{bc}], SessionStop\ [t_2^{bc}], Context\text{-}Modification\ [t_3^{bc}]\}$;

$\rightarrow^{bc} = \{(s_1^{bc}\ t_1^{bc}\ s_2^{bc}), (s_2^{bc}\ t_2^{bc}\ s_1^{bc}), (s_2^{bc}\ t_3^{bc}\ s_2^{bc})\}$;

$s_0^{bc} = \{s_1^{bc}\}$.

The multicast session model supported by MEC application is formally described as $T^{app} = (S^{app}, Act^{app}, \rightarrow^{app}, s_0^{app})$, where

$S^{app} = \{Null\ [s_1^{app})], Established\ [s_2^{app}], WatForUserJoin\ [s_3^{app}], WaitForUserLeave\ [s_4^{app})\}$;

$Act^{app} = \{createSession[t_1^{app}], deleteSession\ [t_2^{app}], notifyJoining\ [t_3^{app}], notifyLeaving\ [t_4^{app}], inviteToJoin\ [t_5^{app}], inviteToLeave\ [t_6^{app}], joinAnswer\ [t_7^{app}], leaveAnswer\ [t_8^{app}]\ \}$;

$\rightarrow^{app} = \{(s_1^{app}\ t_1^{app}\ s_2^{app}), (s_2^{app}\ t_2^{app}\ s_1^{app}), (s_2^{app}\ t_3^{app}\ s_2^{app}), (s_2^{app}\ t_4^{app}\ s_2^{app}), (s_2^{app}\ t_5^{app}\ s_3^{app}), (s_2^{app}\ t_6^{app}\ s_3^{app}), (s_3^{app}\ t_7^{app}\ s_2^{app}), (s_3^{app}\ t_8^{app}\ s_2^{app})\}$;

$s_0^{app} = \{s_1^{app}\}$.

The behavior equivalence of both models is formally proved by using the concept of weak bisimulation.

Proposition: T^{bc} and T^{app} have a weak bisimulation relationship and expose equivalent behavior.

Proof: Strong besimulation between two state machines requires strong correlation between their states and transitions such as one state machine displays the same result as the other state machine. In weak bisimulation, there may be some internal transition that can be skipped, and the results displayed are the same.

Let $R \subseteq (S^{app} \times S^{bc})$ where $R=\{(s_1^{app}, s_1^{bc}), (s_2^{app}, s_2^{bc})\}$. Then the following mapping between transitions of T^{app} and T^{bc} can be identified:

1. The MEC application creates a multicast session: for $(s_1^{app}\ t_1^{app}\ s_2^{app}) \exists (s_1^{bc}\ t_1^{bc}\ s_2^{bc})$.
2. The MEC application invites a user to join to the session: for $(s_2^{app}\ t_5^{app}\ s_3^{app}) \sqcap (s_3^{app}\ t_7^{app}\ s_2^{app}) \exists (s_2^{bc}\ t_3^{bc}\ s_2^{bc})$.
3. The MEC application invites a user to leave the session: for $(s_2^{app}\ t_6^{app}\ s_3^{app}) \sqcap (s_3^{app}\ t_8^{app}\ s_2^{app}) \exists (s_2^{bc}\ t_3^{bc}\ s_2^{bc})$.
4. The MEC application is notified about user joining the session: for $(s_2^{app}\ t_3^{app}\ s_2^{app}) \exists (s_2^{bc}\ t_3^{bc}\ s_2^{bc})$.
5. The MEC application is notified about user leaving the session: for $(s_2^{app}\ t_4^{app}\ s_2^{app}) \exists (s_2^{bc}\ t_3^{bc}\ s_2^{bc})$.
6. The MEC application terminates the multicast session: $(s_2^{app}\ t_2^{app}\ s_1^{app}) \exists (s_2^{bc}\ t_2^{bc}\ s_2^{bc})$.

Therefore, T^{bc} and T^{app} are weakly bisimilar and expose equivalent behavior.

V. Discussions on Service Performance

MEC service metrics are aimed at assessing the improvements introduced by MEC service and evaluating the benefits of service deployment. Service functional metrics are those which impact on user perception e.g. latency, energy efficiency, throughput and other quality of service key performance indicators. Non-functional metrics are related to service lifecycle, service fault tolerance and availability, service computational/ processing load and mobile edge host performance parameters.

As far as non-functional metrics are features of the MEC management system, this section focuses mainly on service functional metrics, in particular on latency.

Latency refers to a time interval between an event and a consequent target effect. Studies on latency measurements on user plane for multimedia multicast services are provided in [17], [18]. Control plane latency introduced by the proposed MEC service can be evaluated as round-trip time (RTT), set-up time, service processing time, and context-update time.

The RTT is defined as the time needed for a request generated by MEC application to go to the destination, to be replied and to go back to the MEC application. The multicast service is connection-oriented as a session is set-up between the user and the MEC application, and the initial signaling affects the latency, so the set-up time has to be minimized. The service processing time is time employed by the MEC server to process the network or application requests. Context-update time has to be considered in case of context-aware applications, such as location dependent multimedia multicast [19].

In [20], the authors provide MEC latency assessment data from an experiment where the fibber-wireless access is tested using round trip time (RTT).

The MEC service processing time to invite user to join to a multicast session can be defined as:

$$\beta_{MEC}=L.X/f=2{,}18 \text{ ms.} \quad (1)$$

where the input task size is regarded as information block of size in bits (L), the complexity of the input task is reflected as necessary CPU cycles per bit (X), and the MEC's host CPU frequency is marked as (f). Then, as to [21], [22] the parameters used are: 4800 bits for information size (for the example POST request shown above); 1000 cycles per bit for complexity; and CPU frequency at 2.2 GHz.

The set-up time introduced by the MSMS can be estimated theoretically by summing the message transfer time (α) along the signaling path and the processing time within each node (β). The processing time within each node in 5G Next Generation Radio Access Network (NGRAN) is considered to be approximately 0.3ms while the message transfer takes 0.1ms, as to [23]. As the proposed MEC service can be deployed in the vicinity of end users with distributed core functionality, the same metrics are adopted for message transfer and processing time within the involved core network functional entities. The message transfer time between core functional entities and MEC platform is assumed to be null as the communication is internal for the MEC host. To assess the set-up time between user and MEC multicast multimedia application, we consider the respective signaling procedures between UE and NG-RAN, between NG-RAN and AMF, between core network functions, and the MEC processing time.

In NG-RAN, Radio Resource Control (RRC) reconfiguration procedure takes place to establish radio bearers for the multicast session [24] and the latency introduced is:

$$L_{UE\text{-}NGRAN}= 2\alpha+2\beta=0{,}8\text{ms.} \quad (2)$$

The communication between NG-RAN and AMF is based on NGAP (Next Generation Application Protocol [25]. The user joining to existing multicast session includes PDU Session Resource Setup procedure, Paging procedure and Initial context setup procedure. The time taken by PDU Session Resource Setup procedure between NG-RAN and AMF, which is to assign resources for the multicast session, is (2α+2β)., The time taken by Initial context setup procedure aimed at establishing UE context at the NG-RAN node is (2α+2β), and the time taken by Paging procedure is (α+β). The time budget for communication between NG-RAN and AMF can be assessed as:

$$L_{NGRAN\text{-}AMF}= 5\alpha+5\beta=2\text{ms.} \quad (3)$$

The core network signaling for Network triggered service request includes Paging and Service request procedures as described in [15], [26]. In case of collocation of distributed core functional entities, the time for message transfer between them is negligible and may be omitted. The Paging procedure takes 3,3ms (11β). The Service Request procedure includes creation of session management context 0,6ms (2β), the subscription retrieval 0,6ms (2β), the PDU session authentication (13α +13β), the policy establishment and modification 1.2ms (4β), the N1N2 message transfer 0,6ms (2β), the N2 PDU session request and acknowledgement 0,6ms (2β), and the session management context update 0,6ms (2β). So, the latency introduced by Network triggered service request procedure is theoretically estimated as:

$$L_{CoreNetwork}= 38\alpha+38\beta=15{,}2\text{ms.} \quad (4)$$

The overall latency introduced by the proposed MSMS in case of application-initiated invitation to user to join to or leave the multicast session is:

$$L_{MSMS}= L_{UE\text{-}NGRAN} + L_{NGRAN\text{-}AMF} + L_{CoreNetwork} + \\ + 2\beta_{MEC} = 18\text{ms} +2\beta_{MEC}=19{,}86 \text{ ms.} \quad (5)$$

We aim to fit an accurate yet simple theoretical distribution to data in order to be able to reuse it in a compact manner upon cases of MEC-based services simulation/development/tests, or about the same when mobile applications are in question.

It seems that the reasonable number of candidates for data fitting should count on the well-known distributions like Normal, LogNormal, Gamma, and Weibull. This list of distributions is very far from exhaustive, but at least a remark might be made - fat-tail distributions are omitted, because the signaling procedures are not like heavily interlinked web traffic.

In [20] is presented an experiment, aiming estimation of MEC latency, which is based on ping-pong test, while in order to include into account the processing time within MEC, we add a component. Further, trying to find the distributions, we use the well-known distributions like Weibull, LogNormal, Gamma, and Normal, for assessment of MEC initiated secondary node addition latency, and we find fits, in form of probability density functions (PDF) of latency, shown as dotted-solid-black in Fig.12.

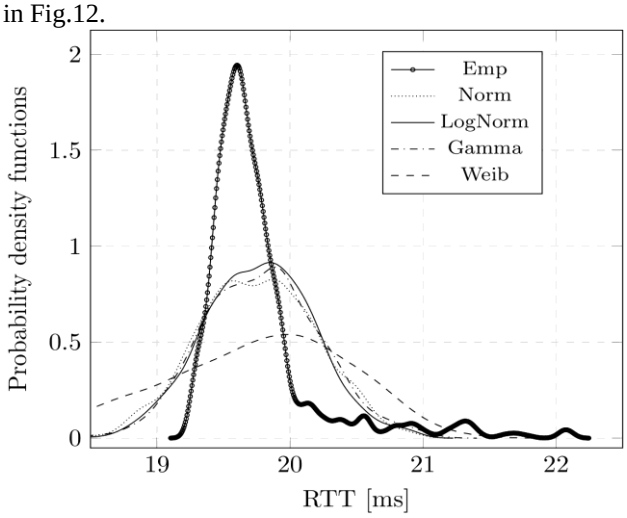

Fig. 12. Probability density functions of round trip time empirical and fitted distributions, where: Emp - the empirical one; Norm - Normal (μ = 19:765; α = 0.4404213); LogNorm - LogNormal (μ = 2:9836713; σ = 0:02167338); Gamma (α = 2090:3097; β = 105:7583); Weib – Weibull (k = 29:81125; λ= 20:01707)

VI. Conclusions

Deployment of multimedia multicast service at the network edge enables third party applications to manage multicast sessions providing location-based multimedia content with guaranteed quality of service. In this paper, we study the MEC capabilities to provide open access to multicast session management functionality which can reduce latency and improve network resource usage. We propose a new MEC service that can be deployed in MEC environment with distributed core functions. The service enables MEC applications to control multicast session establishment, modification and termination, to invite users to join to or to leave the session and to receive notifications about session related events occurred in the network. The provided typical use cases illustrate the main service functionality. The use case descriptions are accompanied by interface and data type definitions, and supported resource structure. The approach feasibility is illustrated by proving the network and application behavior equivalence. Service functional metrics in particular introduced latency are discussed.

The service may be beneficial for mission critical applications with strong low latency requirements as well as for content delivery applications requiring high bandwidth.

Conflict of Interest

The authors declare no conflict of interest.

Author Contributions

Pencheva and Vladislavov conducted the research on open access to multicast service at the network edge; Atanasov did the performance analysis; Trifonov wrote the paper; all authors had approved the final version.

References

[1] D. Gomez-Barquero, D. Navratil, S. Appleby, and M. Stagg, "Point-to-Multipoint communication enablers for the fifth generation of wireless systems," *IEEE Communications Standards Magazine*, vol. 2, no. 1, pp. 53-59, March 2018.

[2] Seth and A. Sharma, "Evolved multimedia broadcasting and multicasting services in LTE-A using device to device communication," in *Proc. International Conference on Inventive Research in Computing Applications (ICIRCA)*, Coimbatore, 2018, pp. 1250-1255.

[3] W. Guo, M. Fuentes, L. Christodoulou, and B. Mouhouche, "Roads to multimedia broadcast multicast services in 5G new radio," in *Proc. IEEE International Symposium on Broadband Multimedia Systems and Broadcasting (BMSB)*, Valencia, 2018, pp. 1-5.

[4] D. Striccoli, G. Piro, and G. Boggia, "Multicast and broadcast services over mobile networks: A survey on standardized approaches and scientific outcomes," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 2, pp. 1020-1063, Second Quarter 2019.

[5] G. Araniti, M. Condoluci, P. Scopelliti, A. Molinaro, and A. Iera, "Multicasting over emerging 5G networks: Challenges and perspectives," *IEEE Network*, vol. 31, no. 2, pp. 80-89, March-April 2017.

[6] J. J. Gimenez, *et al.*, "5G new radio for terrestrial broadcast: A forward-looking approach for NR-MBMS," *IEEE Transactions on Broadcasting*, vol. 65, no. 2, pp. 356-368, June 2019.

[7] J. Montalban, P. Scopelliti, M. Fadda, E. Iradier, C. Desogus, P. Angueira, M. Murroni, and G. Araniti,. "Multimedia multicast services in 5G networks: Subgrouping and non-orthogonal multiple access techniques," *IEEE Communications Magazine*, vol. 56, no. 3, pp. 91-95, 2018.

[8] J. W. Ryu, Q. V. Pham, H. N. T. Luan, W. J. Hwang, J. D. Kim, and J. T. Lee, "Multi-Access edge computing empowered heterogeneous networks: A novel architecture and potential works," *Symmetry*, vol. 11, no. 7, pp. 1-18.

[9] I. Atanasov and E. Pencheva, "Model aspects of open access to multimedia broadcast services in the evolved packet system," *International Journal of Digital Multimedia Broadcasting*, vol. 2016, p. 13, 2016.

[10] 5G Xcast Project. Broadcast and Multicast Communication Enablers for the Fifth Generation of Wireless Systems. *Deliverable D4.1, Mobile Core Network, CORDIS Horizon Programme*. (2018). [Online]. Available: https://cordis.europa.eu/ project/rcn/211065/factsheet/en

[11] Q. V. Pham, F. Fang, V. N. Ha, M. Le, Z. Ding, L. B. Le, and W. J. Hang, "A survey of multi-access edge computing in 5G and beyond: Fundamentals, technology integration, and State-of-the-Art," Cornell University, Computer Science, Networking and Internet Architecture, arXiv:1906.08452v1 [cs.NI], Jun 20, 2019, pp. 1-59.

[12] E. Pencheva, I. Atanasov, D. Velkova, and V. Trifonov, "Application Level User Traffic Control at the Mobile Network Edge," in *Proc. Open Innovation Association 24th FRUCT (Finnish-Russian University Cooperation in Telecommunications)*, Moscow, Russia, April 4-12, 2019, pp. 312-327.

[13] F. Giust, *et al*., "MEC deployment in 4G and evaluation towards 5G," ETSI White Paper, no. 24, pp. 1-24, 2019

[14] 3GPP TS 33.501 Technical Specification Group Services and System Aspects; System Architecture for the 5G System, Stage 2, Release 16, v16.2.0, 2019.

[15] 3GPP TS 23.502 Technical Specification Group Services and System Aspects; Procedures for the 5G System (5GS), Stage 2; Release 16, v16.2.0, 2019.

[16] ETSI GS MEC 013 Mobile Edge Computing (MEC); Location API, v1.1.1, 2017.

[17] S. Roger, *et al*., "Low-Latency Layer-2-Based multicast scheme for localized V2X communications," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 8, pp. 2962-2975, Aug. 2019.

[18] S. Natarajan and D. Das, "Demand aware edge caching architecture for evolved multimedia broadcast multicast service to reduce latency and bandwidth savings," in *Proc. IEEE International Conference on Electronics, Computing and Communication Technologies (CONECCT)*, Bangalore, 2018, pp. 1-6.

[19] ETSI GS MEC-IEG 006, Mobile Edge Computing; Market Acceleration; MEC Metrics Best Practice and Guidelines, v1.1.1, 2017

[20] Y. Liu, G. Shou, Y. Liu, Y. Hu, and Z. Guo, "Performance evaluation of integrated multi-access edge computing and fiber-wireless access networks," *IEEE Access*, 2018, vol. 6, pp. 30269-30-279.

[21] K. Cheng, Y. Teng, W. Sun, A. Liu, and X. Wang, "Energy-efficient joint offloading and wireless resource allocation strategy in Multi-MEC server systems," in *Proc. IEEE International Conference on Communications (ICC)*, Kansas City, MO, 2018, pp. 1-6.

[22] B. Bertenyi, *et al*. "5G NR radio interface," *Journal of ICT Standardization*, vol. 6, no. 1-2, pp. 31-58, May 2018.

[23] ITU-R M.2410-0 Minimum Requirements Related to Technical Performance for IMT-2020 Radio Interface(s), 2017.

[24] 3GPP TS Technical Specification Group Radio Access Network, NR, Radio Resource Control (RRC) protocol specification Release 16, v16.2.0, 2019.

[25] 3GPP TS 38.413 Technical Specification Group Radio Access Network, NG-RAN, NG Application Protocol (NGAP), v16.2.0, 2019, Release 16.

[26] 3GPP TS 33.501 Technical Specification Group Services and System Aspects, Security architecture and procedures for 5G system, Release 16, v16.0.0, 2019.

Ivaylo Atanasov is born in Sofia, Bulgaria. He has received his MSc degree in Electronics and PhD degree in Communication networks from Technical University of Sofia (TU-Sofia). He has been awarded DSc degree in communication networks in 2016 from the Faculty of Telecommunications, TU-Sofia. Since 2013, he is Professor and his scientific research area covers mobile networks, internet communications and protocols, and mobile applications.

Evelina Pencheva is with the Faculty of Telecommunications, Technical University of Sofia. She is born in Sofia. She has received her MSc degree in Mathematics from Sofia University "St. Kliment Ohridski" and PhD degree in Communication networks from TU-Sofia. She has defended her DSc thesis in 2014. Since 2010, she is Professor and her scientific research area covers multimedia networks, telecommunication protocols, and service platforms.

Vladislav Vladislavov is born in Sofia, Bulgaria in 1989. He has received the BSc. degree from the High Scholl of Transport "Todor Kableshkov" Sofia, in 2013. The MSc. degree is received at the TU-Sofia in 2015. Both degrees are in Communication Networks and Safety Critical Systems. He is currently working on his PhD thesis (start at 2019) with Faculty of Telecommunications, TU-Sofia. His research interests include mobile networks and services, artificial intelligence and intrusion detection.

Ventsislav Trifonov is born in Sofia, Bulgaria in 1968. He has received the M.Sc. degree from the High Scholl of Transport "Todor Kableshkov" Sofia, in 1991. The PhD degree has been awarded by TU-Sofia, 2014. Both degrees are in the field of Safety Critical and Fail-Safe System design and analysis in Telecommunications. He works as Associated Processor. His research interests include safety critical system, reliability theory and artificial intelligence for intrusion detection.