A Performance Evaluation of WebRTC over LTE

Giuliana Carullo, Marco Tambasco
Research Consortium on Telecommunications (Co.Ri.Tel.)
University of Salerno, 84084, Fisciano (SA) - Italy
Email: giuliana.carullo, marco.tambasco{@coritel.it}

Mario Di Mauro, Maurizio Longo
Dept. of Information Engineering,
Electrical Engineering and Applied Mathematics (DIEM)
University of Salerno, 84084, Fisciano (SA) - Italy
Email: mdimauro, longo{@unisa.it}

***Abstract*—In perspective, the effectiveness and attractiveness of multimedia communications rely both on such disruptive services as Web Real-Time Communication (WebRTC), and on such reliable mobile infrastructures as Long Term Evolution (LTE). WebRTC is the innovative protocol letting HTML5 compliant browsers to communicate in real-time using a peer-to-peer architecture. LTE on the other hand, is the current technology for mobile communication systems as standardized by the 3rd Generation Partnership Project (3GPP). In this work, we focus on WebRTC over LTE. We realized a testbed based on NS-3 framework including: *i)* some LTE customized modules *ii)* an ad hoc server providing a set of services for WebRTC call setup *iii)* two mobile clients equipped with a HTML5 browser for WebRTC audio/video call support. Several multimedia WebRTC flows between the two clients have been analyzed and empirical CDFs of typical performance figures including throughput, jitter, and packet loss have been derived under different LTE scenarios.**

***Keywords*—WebRTC, LTE, NS-3, real-time traffic.**

I. Introduction

In recent years, the evolution of both telecommunication infrastructures and mobile equipments encouraged customers to massively use multimedia services such as enriched calls, video streaming, peer-to-peer sharing and so on. Enabling the delivery of video services to mobile devices over cellular and mobile broadband networks is essential to ensure ubiquitous access to video content and services from any location, at any time, with any device and technology. The latest mobile broadband technologies including LTE and LTE-Advanced standards, exhibit theoretical data rate up to 3 Gbps downlink and up to 1.5 Gpbs uplink for Release 10. These increased data rates allow for enhanced user experience and support for multimedia services at lower transmission cost and with much richer content than 2G and 3G cellular networks [1]. According to the Ericsson Mobility Report [2] for the period 2015-2020, monthly global mobile traffic will surpass 30.5 exabytes by 2020 and a 40% of mobile LTE subscriptions growth by the same year has been foreseen. Therefore, the delivery of enriched multimedia content requires on one hand, the design of performing web applications with possibility of controlling Quality-of-Service (QoS) parameters and on the other hand, a powerful telecommunication infrastructure able to guarantee a more effective bandwidth. From an application perspective, WebRTC is a relatively recent standard making possible a multimedia session between two clients (mobile or not) equipped with a HTML5 web browser. WebRTC appears particularly promising in the area of interactive services and represents a break through in the application world since it enables web developers to build real-time multimedia applications with no need for proprietary plug-ins [3]. From an infrastructure perspective instead, LTE represents the most encouraging standard for mobile communications that exhibits some powerful features as: Multiple-Input-Multiple-Output (MIMO) schemes allowing to exploit multiple antennas at the transmitter and receiver sides intended to achieve better performance gains (in terms of array gain, spatial diversity gain and interference reduction) [4], more robust transmission schemes as OFDM and SC-FDM, improved performances in terms of handover and spectral efficiency [5]. However, testing the behaviour of a performing network is an hard task. Thus, in this work, we present a testbed whose aim is to enable some experiments about real-time multimedia sessions by using the WebRTC paradigm over a LTE-capable infrastructure with the assistance of the NS-3 simulator. Besides, some classes of NS-3 have been modified to make possible a real-time emulation over LTE modules. The paper is organized as follows. Section II concerns the analysis of some related works. In Section III, an overview of WebRTC protocol stack and essential features are described. In Section IV an overview of NS-3 with LTE support and a general perspective of our testbed is provided. Section V presents a detailed view of the realized testbed and the considered network scenarios. Then, upcoming experimental results are discussed in VI. Section VII ends this paper by drawing main conclusions and future research works.

II. Related Work

To the best of our knowledge, no work focused on evaluating the performances of WebRTC over LTE has been proposed before. However, in recent literature, several studies addressed the theme of multimedia streaming delivery over LTE infrastructure. In [6] a stochastic service model for the evaluation of the quality of real-time streaming (e.g. mobile TV) in 3GPP LTE environment has been proposed. The authors consider a multi-class Markovian process of call arrivals and take into account different radio channel conditions, requested streaming bit-rates and call-durations to characterize the outage of network resources in a real-time environment. Authors in [7] afford the theme of Quality-of-Experience (QoE) in multimedia delivering technologies, by presenting a QoE-based evaluation methodology to assess the LTE system video capacity in terms of the number of unicast video consumers that can be simultaneously served. Indeed, network operators are constantly in search of solutions which allow them to provide improved capacity services and QoE guarantees for

ISBN 978-3-901882-80-7

their video users with limited resources. A key idea for delivering QoE-based multimedia services is the adaptive streaming, aimed at optimizing the video configurations over time in order to address the best possible user experience. Authors in [8] remain on the QoE theme by proposing a functional model for the QoE assessment in real-time Internet services, with focus on VoLTE (Voice over LTE) technology. In particular, some extra nodes in LTE architecture are suggested for Video Streaming management. In [9] an analysis on the performance of downlink packet scheduling algorithms in LTE networks is presented. In this paper, the multimedia traffic over LTE infrastructure is modelled by exploiting the functionalities of NS-3 simulator, and the performances of different scheduling algorithms in various scenarios have been analyzed. Other strategies aimed at optimally managing multimedia streaming over next generation infrastructures are based on different resource allocation techniques in the LTE system as addressed in [10]. Resource allocation allows to determine the amount of radio resources that have to be assigned to each user. Such a paper focuses on the algorithms that divide packet scheduling into resource allocation and resource assignment techniques. In [11] the key idea consists in selecting the path from a video source that best meets the user satisfaction. This is done by considering two performance metrics: throughput, defined as the total number of packets successfully delivered to the users in a video session, and packet delivery delay. Authors also provide a performance analysis with the assistance of NS-3 simulations considering a pedestrian users scenario. An evaluation of the RTP circuit breaker algorithm on LTE networks is presented in [12]. Authors propose to exploit some fields of RTP Control Protocol (RTCP) in order to detect if a media flow is overusing the available capacity and causing congestion. Finally, moved by the significant growth of multicast video streaming in 4G/LTE environment, in [13], a cooperative protocol for multicast systems has been proposed. This protocol considers nodes cooperating at application level, whilst receiving video chunks from a streaming source.

III. WebRTC Protocol Stack

WebRTC [14] is an open project supported by the World Wide Web Consortium (W3C) and the Internet Engineering Task Force (IETF) that provides both browsers and mobile applications with real-time communications capabilities.

In particular, it enables web applications to establish a peer-to-peer communication channel by exploiting a set of innovative APIs. Moreover, all the multimedia functions are natively integrated by the browser makers including codecs, management of exchanged streams and so on. WebRTC extends the client-server semantic by introducing the peer-to-peer communication paradigm between the browsers. The most general WebRTC architectural model draws its inspiration from the Session Initiation Protocol (SIP) trapezoid [15] and it is shown in Fig. 1. In this model, the signaling messages involved into communications are carried by the HTTP or WebSocket protocols traversing web servers that can modify, translate, or manage them as needed. Observe that, whilst data communication between peers is performed in a peer-to-peer fashion, signaling between browser and server is not standardized. The web servers, indeed, can communicate using any signaling protocol, including SIP, Jingle [16] or any proprietary one.

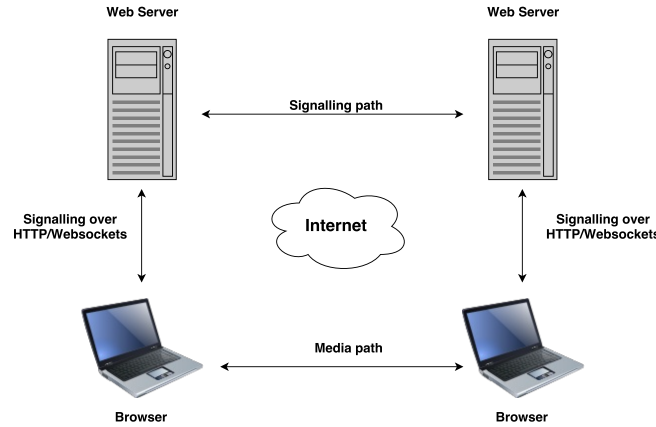

Fig. 1: The WebRTC model

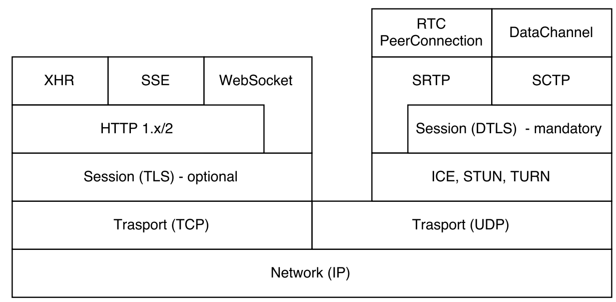

Fig. 2: WebRTC protocol stack

The WebRTC protocol stack depicted in Fig. 2, typically relies on the UDP protocol in order to carry multimedia traffic as fast as possible. Besides, several protocols and services above UDP are required to traverse layers of NATs and firewalls, to negotiate the parameters for each stream, to provide data encryption and to implement congestion and flow control. These services include Interactive Connectivity Establishment (ICE) [17], Session Traversal Utilities for NAT (STUN) [18], and Traversal Using Relays around NAT (TURN) [19] required to establish and maintain a peer-to-peer connection over UDP. Security of transferred data is guaranteed by Datagram Transport Layer Security protocol (DTLS) [20], being encryption an essential feature of WebRTC. Stream Control Transmission Protocol (SCTP) [21] and Secure Real-time Transport Protocol (SRTP) [22] are used to multiplex the different streams, to provide congestion and flow control and to ensure partially reliable delivery as well as other additional services.

Basically, the WebRTC APIs are designed around three main concepts [23]: *i) Mediastream:* an abstract representation of audio/video data streams serving as a handle for managing actions such as displaying the stream's content, recording, or sending it to a remote peer. A *MediaStream* can contain zero or multiple tracks identified by a *MediaStreamTrack* object representing a specific media source on the user agent; *ii) RTCPeerConnection:* an interface on top of WebRTC protocol stack responsible of transferring streaming data between browser peers by providing an abstraction for bidirectional multimedia communication channel. Fig. 3 shows the creation of a *RTCPeerConnection* object being input/output for a par-

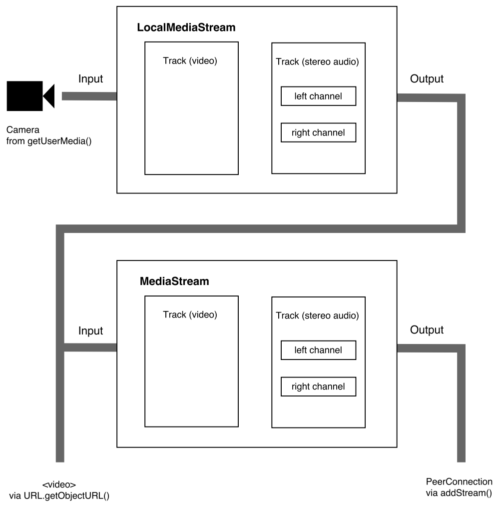

Fig. 3: PeerConnection and MediaStream objects

ticular *MediaStream*; *iii) DataChannel*: an entity providing a generic transport service allowing web browsers to exchange data in a bidirectional peer-to-peer fashion.

IV. Simulated LTE Environment within NS-3

In this Section, we present our experimental's setup. In particular, we first introduce in IV-A some basic facts about NS-3 simulator and the provided modules for the LTE environment support and then, in IV-B, we detail the structure of the realized testbed.

A. NS-3 Simulator and LTE support

NS-3 is a discrete-event network simulator for Internet systems [24] released under the GNU GPLv2 license. It is developed and distributed completely in C++, whilst simulation codes can be written as either C++ or Python programs. NS-3 enables simulation experiments by providing realistic models on how packet data networks work and perform. NS-3 also provides support for several models and protocols as Wi-Fi, WiMAX and LTE, just to mention some. In particular, the LTE module [25] has been designed to support the evaluation of several aspects of LTE elements, namely Radio Resource Management, QoS-aware Packet Scheduling, Intercell Interference Coordination and Dynamic Spectrum Access. It includes the LTE Radio Protocol stack (PDCP, RLC, MAC, PHY), residing entirely within the User Equipment (UE) and the Evolved Node B (eNB) as depicted in Fig. 4.

Furthermore, the Serving Gateway (SGW), allowing the user data packets to be routed and forwarded, and the Packet Data Network Gateway (PGW), authorizing the interconnection with external networks, belong to the so-called Evolved Packet Core (EPC). EPC is the all-IP mobile core network for LTE infrastructure defined around three paradigms: *i) Mobility* involving modules as the Mobility Management Entity (MME) aimed at controlling the signaling plane (paging, handover, bearer activation/deactivation etc.); *ii) Policy Management* involving modules as the Policy and Charging Rules Function (PCRF) in charge of providing interfaces for billing systems;

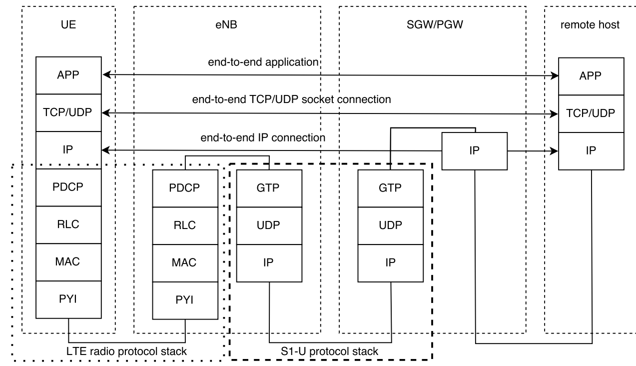

Fig. 4: LTE-EPC protocol stack

iii) Security involving modules as Home Subscriber Server (HSS) acting as authentication and authorization point for the mobile users. Besides, the LTE module embedded in NS-3 environment supports fading and mobility models too. In particular, it includes three different 3GPP-approved scenarios for fading modeling: pedestrian, vehicular and urban, each one considering different users' speeds. All these scenarios use precomputed traces embedded into NS-3.

B. Testbed View

The realized testbed consists of 4 PCs: the first one (Desktop, QuadCore at 2.4 GHz, 8 GBytes of RAM) hosting an NS-3 simulator with LTE support, two laptops (DualCore at 2.5 GHz, 4 GBytes of RAM) serving as WebRTC clients, and the last one (Desktop, QuadCore at 2.4 GHz, 8 GBytes of RAM) acting as WebRTC server. A logical view of testbed is shown in Fig. 5 and includes the following elements:

- Two mobile UEs, each one equipped with HTML5 compliant browser (on the top of TCP/IP stack) letting to set up a WebRTC audio/video call.
- A simulated LTE environment including two eNBs (serving separately the two UEs) and the EPC entity allowing IPv4 networking with LTE devices.
- An element hosting the WebRTC server - realized with *Node.js* - that acts as a signaling server aimed at coordinating the communication between the UEs.

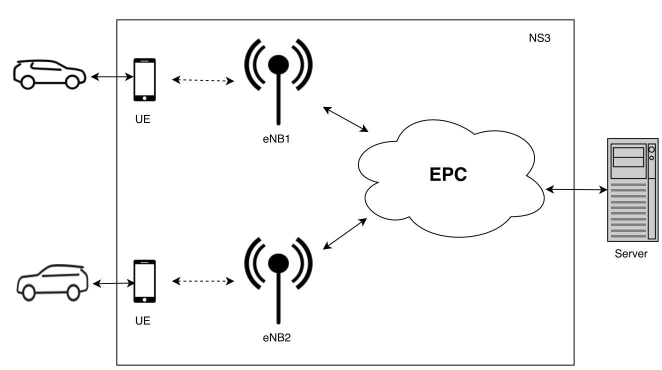

Fig. 5: General view of testbed

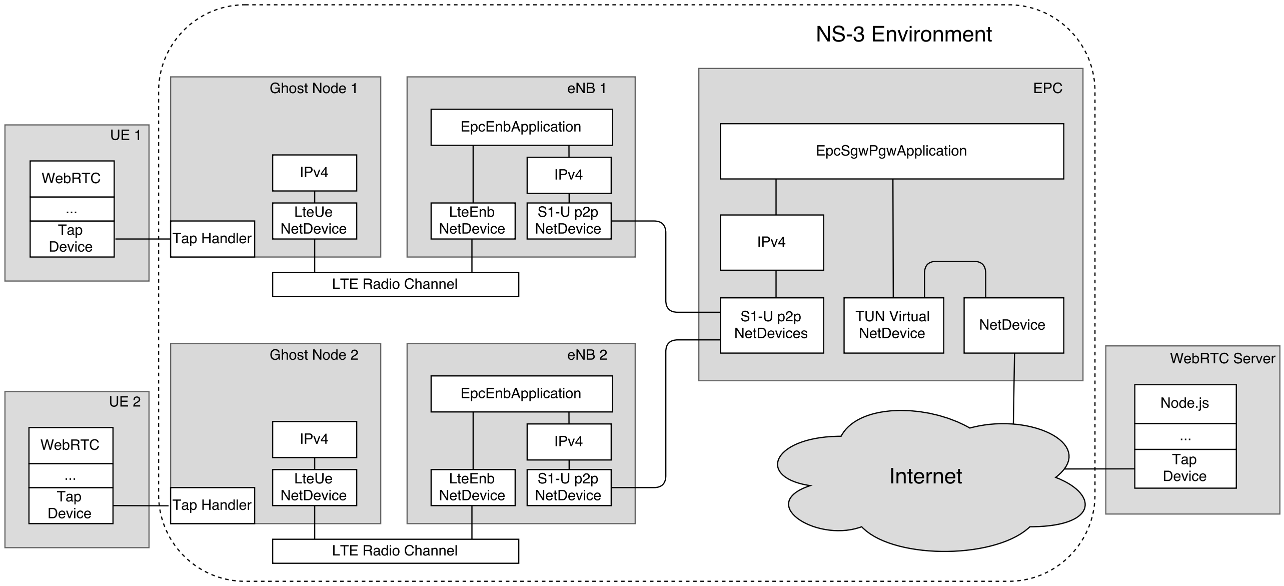

Fig. 6: The realized testbed architecture

This node has a dual purpose: on one hand, it takes care of the information exchange process involved in the initial phase of a multimedia call. On the other hand, it takes advantage of STUN and TURN functionalities (exploited by ICE), in order to manage NAT-traversal and firewall issues [26].

V. Experiment Setup

In this Section, we first present more in details the simulation setup and then the scenarios we considered during our simulations. The testbed architecture is depicted in Fig. 6, and is organized as follows. The two mobile UEs are split in two parts: an emulated one (outside of NS-3 environment) including the physical device (laptop) equipped with HTML5 compliant browser (on the top of TCP/IP stack) for the set up of a WebRTC audio/video call; this element incorporates a Tap Device (at bottom of TCP/IP stack) allowing to communicate with the simulated part (inside of NS-3 environment) called ghost node. The ghost nodes are connected to eNBs via LTE radio channel and are devoted to simulate the classic radio procedures. In particular, the ghost nodes exploit a module called *LteUeNetDevice* performing the packets delivery to eNBs over LTE radio protocol stack. At eNB level instead, a module called *EpcEnbApplication* is mainly responsible to send packets to the the SGW/PGW node via the UDP socket. The two eNBs are connected to the EPC entity on behalf of S1-U interface. The main module of EPC is named *EpcSgw-PgwApplication* and is in charge of determining the eNB node the UE is attached to. Moreover, it is involved in forwarding the data packets to the *VirtualNetDevice* representing a contact point towards external networks (as Internet). The WebRTC server, at last, is hosted on a PC with *Node.js* (as mentioned before) and, similar to the UEs, consists of an emulated part (outside the NS-3 environment) and a simulated one (inside the NS-3 environment).

General Setup	
Simulator	NS-3 version 3.22
Number of eNB	2
Number of UE	2
Mobility Model	Custom model
Distance between eNB	10 km
Init. distance UE-eNB	700 mt
Number of multimedia calls	120
Video and Audio Codecs	VP8 and Opus
Average calls duration	300 seconds

TABLE I: General Simulation Parameters

A. Parameter Selection

The general parameters we define for all our simulations are reported in Table I. All the performed NS-3 simulations consist in 120 WebRTC calls (intended as 30 calls in 4 operative scenarios described in the next section) analyzed in a 300 second long temporal window. Indeed, two mobile UEs are considered, each one attached to a different eNB. The two eNBs are 10 km away from each other, and, initially, each UE is 700 meters away from the eNB it is associated with. During the simulation, UEs move following a custom random path generated in Matlab and imported in the simulator with an average speed of $20m/s$.

B. Scenarios

In our experiments we considered four different scenarios that differ each other depending on the fading model, channel error, queue limitations and traffic interferences generated by surrounding nodes. Those scenarios, including progressively realistic features, are as follows:

Scenario #1
This is a pure ideal case used for benchmarking purposes. The two mobile equipments communicate each other in the free space (no assumptions about channel propagation conditions) and no packet errors or queue limitations are taken into account.

Scenario #2
In this model, both fading and channel error have been contemplated. In particular, a random variable uniformly distributed between 0 and 1 is compared against an error rate of 10% to simulate the packet losses. The fading effect takes into account a vehicular model and it has been derived on the basis of [27].

Scenario #3
Both fading and channel error are modeled as in the previous scenario. Besides, in order to obtain a more realistic setup, we also imposed limitation on the packets queue defined on each mobile node. In particular, nodes are enforced to use a finite queue of 100 packets managed by a FIFO mechanism. This expedient circumvents the inability of NS-3 to natively support finite packet queues on LTE embedded nodes.

Scenario #4
In addition to the Scenario #3, six interfering nodes are introduced (three for each eNB). Those nodes are independent from the two terminal nodes and they send various traffic towards an extra server attached to the PGW.

VI. Results

In this Section, we present some numerical results obtained by performing tests carried out taking into account the different proposed scenarios. All the experiments have been led in a controlled environment in order to guarantee as much correctness as possible. The considered use case involves WebRTC audio/video calls between two mobile users as clarified in the sequence diagram shown in Fig. 7 containing a simplified adaptation of radio network attach procedures. Before setting up an end-to-end WebRTC call, the terminals perform, at first, a radio attach procedure towards the respective eNB and then, a connection towards the WebRTC server in order to initiate the multimedia call. All the experimental data have been gathered at UE level (considering a one-way flow) and analyzed with the goal of extracting some typical performance figures (conveniently averaged on the basis of the number of considered calls for each scenario), namely, the user throughput, the packet loss ratio and the jitter.

In Fig. 8 the empirical CDF (Cumulative Distribution Function) of the User Throughput is depicted. The figure shows that more than 50% of the WebRTC flow reaches a throughput of 170 Kbit/s in the Scenario #1 where no interferences, fading phaenomena or finite queues on the nodes are present. By considering more realistic setups, the percentage of the WebRTC flow exhibiting higher throughput values is decreasing.

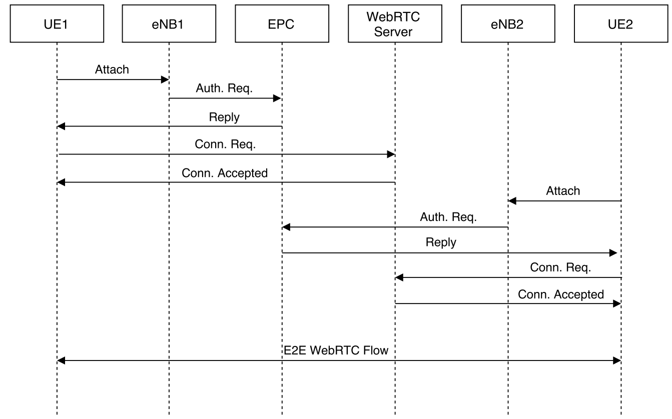

Fig. 7: Sequence diagram of the proposed use case

Fig. 9 instead, shows the empirical CDF of the jitter, namely, the variance of data packet inter-arrival times. As shown in figure, the 50% of jitter parameter lies around 4.6 milliseconds in the case of the ideal scenario and tends to grow in the rest of cases. The most critical setup appears to be the Scenario #4 where some interfering data traffic causes an increase in the jitter value for the test flow; in this case, indeed, 50% of jitter parameter lies around 6.3 milliseconds. Another parameter influenced by the communication quality is the packet loss ratio. The most favourable setup, as shown in Fig. 10, turns out to be the ideal case where a very limited packet loss ratio can be observed: for example, 50% of the whole WebRTC flow suffers of about 2.3% of packet loss

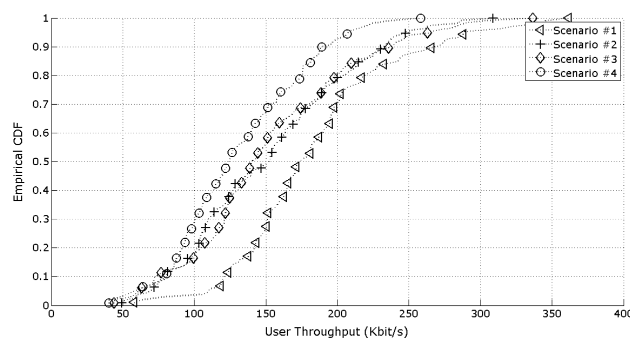

Fig. 8: Empirical CDF of the User Throughput

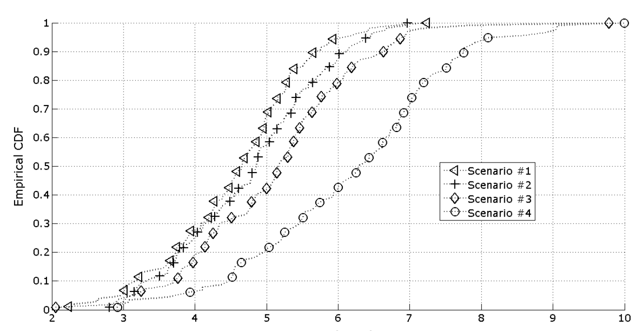

Fig. 9: Empirical CDF of the Jitter

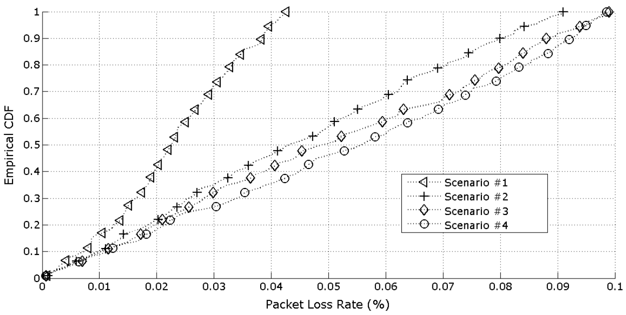

Fig. 10: Empirical CDF of the Packet Loss

whereas the case of Scenario #4 exhibits a packet loss of about 5.6%.

The obtained results show how the typical parameters are influenced by different network setups and exhibit a progressive decay from the ideal setting, up to more realistic ones.

VII. Conclusions

Nowadays, LTE and WebRTC represent two key elements in the arena of the mobile-based multimedia services. The former is the recent 3GPP radio access standard defined with a vision of an all-IP network letting to achieve theoretical data rates up to 3 Gbps downlink whereas the latter is an open technology project born to enable web-centric services by providing both browsers and mobile applications with real-time communications capabilities such as P2P video, audio voice, and data communication via simple APIs. In this paper, we present a setup considering these two powerful technologies aimed at analyzing some WebRTC audio/video flows between two mobile users forwarded inside an LTE simulated environment. A performance analysis has been carried out by taking into account some typical performance figures of multimedia calls such as throughput, jitter and packet loss. In order to analyze the flow calls in different conditions, four scenarios have been considered: a first ideal one, acting as a benchmark, where no interference effects nor packet queueing processes are contemplated and other three scenarios where some realistic features are progressively added (finite packet queues, fading phaenomena etc.). The experimental results, expressed in terms of empirical CDFs, showed the deterioration of multimedia flows quality when more realistic setups are considered. Future works will be devoted at examining more sophisticated user mobility models in order to analyze the behaviour of the aforementioned parameters. Finally, we have also in mind to test scalability properties of the whole system when more nodes are considered.

References

[1] O. Oyman, J. Foerster, Y.-J. Tcha, and S.-C. Lee, "Toward enhanced mobile video services over WiMAX and LTE," *IEEE Communications Magazine*, vol. 48, no. 8, pp. 68–76, 2010.

[2] "Ericsson mobility report, June 2015," tech. rep., Ericsson, 2015.

[3] M. Di Mauro and M. Longo, "Revealing encrypted WebRTC traffic via machine learning tools," in *SECRYPT 2015 - Proceedings of the 12th International Conference on Security and Cryptography, Colmar, Alsace, France, 20-22 July, 2015.*, pp. 259–266.

[4] E. Biglieri, R. Calderbank, A. Constantinides, A. Goldsmith, A. Paulraj, and H. V. Poor, *MIMO Wireless Communications*. New York, NY, USA: Cambridge University Press, 2007.

[5] A. Elnashar and M. El-Saidny, "Looking at LTE in practice: A performance analysis of the LTE system based on field test results," *IEEE Vehicular Technology Magazine*, vol. 8, no. 3, pp. 81–92, 2013.

[6] B. Baszczyszyn, M. Jovanovic, and M. Karray, "Quality of real-time streaming in wireless cellular networks - stochastic modeling and analysis," *IEEE Transactions on Wireless Communications*, vol. 13, no. 6, pp. 3124–3136, 2014.

[7] S. Singh, O. Oyman, A. Papathanassiou, D. Chatterjee, and J. Andrews, "Video capacity and QoE enhancements over LTE," *IEEE International Conference on Communications*, pp. 7071–7076, 2012.

[8] S. Forconi and M. Vaser, "4G LTE architectural and functional models of video streaming and volte services," in *Ubiquitous and Future Networks, 2015 International Conference on*, pp. 787–792, July 2015.

[9] T. Nandu, N. Krishnan, K. Theja, B. Pattnaik, P. Anilal, and S. Sai, "Simulative analysis of multimedia traffic over LTE with real time scenarios in network simulator-3," in *Advanced Networks and Telecommuncations Systems, 2014 IEEE International Conference on*, pp. 1–6, Dec 2014.

[10] H. Ramli, K. Sandrasegaran, R. Basukala, R. Patachaianand, M. Xue, and C.-C. Lin, "Resource allocation technique for video streaming applications in the LTE system," *WOCC2010 Technical Program - The 19th Annual Wireless and Optical Communications Conference: Converging Communications Around the Pacific*, 2010.

[11] D. Munaretto, M. Zanforlin, and M. Zorzi, "Performance evaluation in NS-3 of a video delivery framework for next generation cellular networks," in *Communications Workshops (ICC), 2013 IEEE International Conference on*, pp. 642–646, June 2013.

[12] Z. Sarker, V. Singh, and C. Perkins, "An evaluation of rtp circuit breaker performance on lte networks," *Proceedings - IEEE INFOCOM*, pp. 251–256, 2014.

[13] A. Russo and R. Lo Cigno, "Pullcast: Peer-assisted video multicasting for wireless mesh networks," *2013 10th Annual Conference on Wireless On-Demand Network Systems and Services*, pp. 60–67, 2013.

[14] A. Bergkvist, D. Burnett, C. Jennings, and A. Narayanan, "Webrtc 1.0: Real-time communication between browsers," 2013. W3C Working Draft.

[15] J. Rosenberg, H. Schulzrinne, G. Camarillo, A. Johnston, J. Peterson, R. Sparks, M. Handley, and E. Schooler, "Rfc 3261: Sip: Session initiation protocol," 2002. RFC 3261.

[16] S. Ludwig, J. Beda, P. Saint-Andre, R. McQueen, S. Egan, and J. Hildebrand, "Xep-0166: Jingle," *XMPP Standards Foundation*, 2009.

[17] J. Rosenberg, "Interactive connectivity establishment (ICE): A protocol for network address translator traversal for offer," 2010. RFC 5245.

[18] J. Rosenberg, R. Mahy, P. Matthews, and D. Wing, "Session traversal utilities for NAT (STUN)," 2008. RFC 5389.

[19] R. Mahy, P. Matthews, and J. Rosenberg, "Traversal using relays around NAT (TURN): Relay extensions to session traversal utilities for NAT (STUN)," 2010. RFC 5766.

[20] E. Rescorla and N. Modadugu, "Datagram transport layer security version 1.2," 2012. RFC 6347.

[21] M. Tuexen and R. Stewart, "Stream control transmission protocol (SCTP) chunk flags registration," 2011. RFC 6096.

[22] J. Ott and E. Carrara, "Extended secure RTP profile for real-time transport control protocol (rtcp)-based feedback," 2008. RFC 5124.

[23] S. Loreto and S. Romano, "Real-time communications in the web: Issues, achievements, and ongoing standardization efforts," *IEEE Internet Computing*, vol. 16, no. 5, pp. 68–73, 2012.

[24] G. F. Riley and T. R. Henderson, "The NS-3 network simulator," in *Modeling and Tools for Network Simulation*, pp. 15–34, Springer, 2010.

[25] G. Piro, N. Baldo, and M. Miozzo, "An LTE module for the NS-3 network simulator," pp. 415–422, 2011.

[26] Y.-D. Lin, C.-C. Tseng, C.-Y. Ho, and Y.-H. Wu, "How NAT-compatible are VoIP applications?," *IEEE Communications Magazine*, vol. 48, no. 12, pp. 58–65, 2010.

[27] "ETSI technical specification n. 136104 v10.2.0 (2011-05)," tech. rep., European Telecommunications Standards Institute, 2011.