A Performance Evaluation Of Peer-To-Peer Media Communication Over LTE

Sanabil A. Mahmood[1] , Ali Tariq Kalil Al-khayyat[2]

[1]College of Computer Sciences and Mathematics. The University of Mosul, Mosul, Iraq.

[2]College of Dentistry. The University of Mosul, Mosul, Iraq.

Abstract

Using the Internet instead of Public Switched Telephone Network (PSTN) to exchange audio and video has taken massive consideration while it gives better quality and low cost. Thus, Web Real-Time Communication (WebRTC) standard has become the most environment for building audio and video applications. The main aim of this contribution is to assess the performance of peer-to-peer (peer-to-peer) video communication using Chrome and the 4th generation (4G) internet. Likewise, an evaluation of resources, such as CPU performance and bandwidth consumption were completed. Furthermore, an evaluation of signalling channels between peers (browsers) using the Node.js platform has been produced. This work will support web researchers a chance to realize the advantages and disadvantages of a video conferencing based WebRTC.

Keywords:

The Web Real-Time Communication (WebRTC); Node.JS. and 4G.

Introduction

The "Web Real-Time Communication" group, such as "World Wide Web Consortium" and "Internet Engineering Task Force" established an original ordinary free-source context as (P2P) that is deliberated as a group of procedures and JavaScript that sustained by Opera, Firefox and son [1][2]. WebRTC was announced in the middle of various users and devices, On the other hand, the WebRTC group have not decided on an absolute communication mechanism to examine WebRTC [3][4]. Consequently, signalling is the foremost noteworthy operation matter with WebRTC [5]. On the peer's side, signalling is utilised to find a system that locates any existing peers and controls communication between them. [6]. The main goals of this project are to analyze the performance of a conceptual implementation of WebRTC video communication via the Node.js framework. As well, an evaluation of the bandwidth bitrate and CPU was done. Besides, in this research users will get a massive opportunity to understand WebRTC and communication of a connection between the client and server. This is a short way to state the usage of WebRTC video conferencing between several communications.

Node.JS: is built as a server platform to be offered as an open-source high presentation JavaScript machine. Also, is a suitable platform for construction network web submission. In addition, it can hold a great quantity and show since it is a non-blocking Input/Output prototypical [7]. Figure (1), shows the WebRTC video and audio communication, and figure (2) shows MediaStream and PeerConnection.
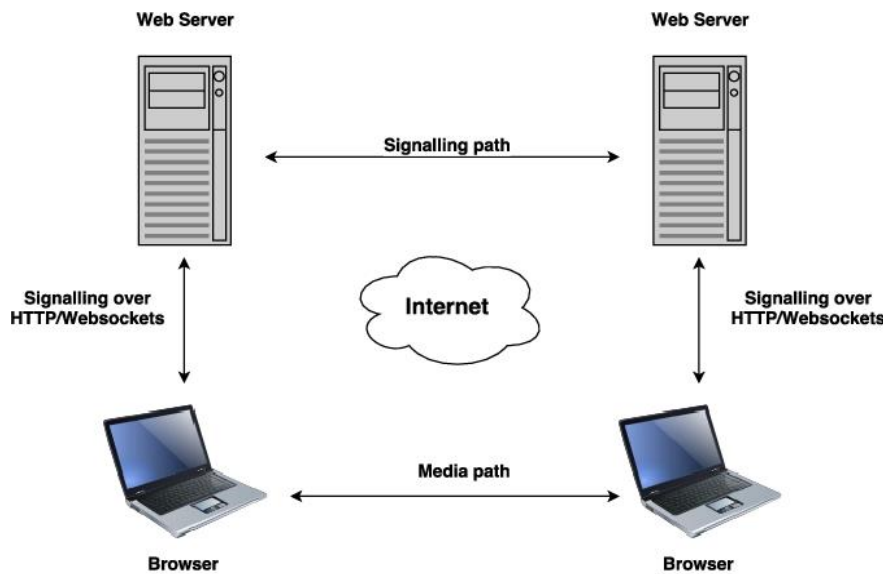

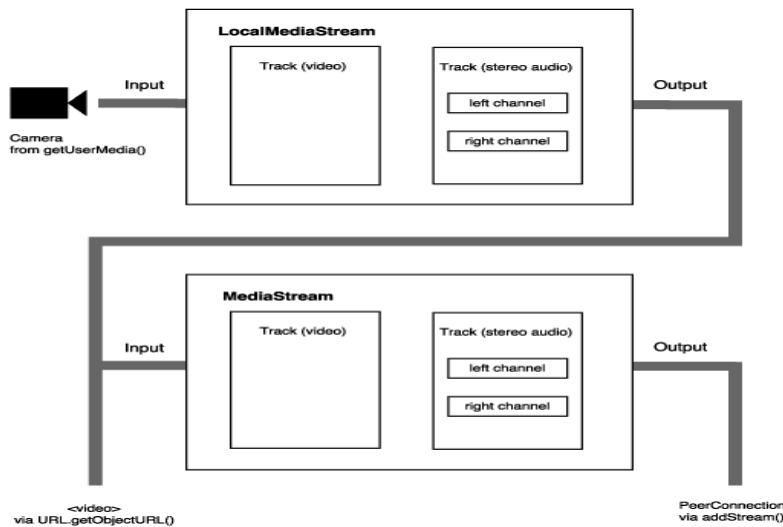**Figure 1 the WebRTC peer-to-peer communication model**

Figure 2 WebRTC MediaStream objects and PeerConnection

The following is a breakdown of the effort: Section II explains the evaluation process. Section III concludes with a discussion of future work.

Literature Review

Based on the research through published papers, it was a little difficult to find a focus for evaluating the performances of WebRTC over LTE (4G). Nevertheless, some literature were discussed a few subjects of multimedia communication supplying over LTE arrangement as the following: in [8] illustrated that a service model for evaluating real-time media stream in 3GPP situation was proposed. However, the authors applied the proposed system on mobile TV using a Markovian process, conditions of the radio channel, and bit rates for network resources. Besides, [9] demonstrated multimedia distributing technologies using Quality-of-Experience (QoE) to assess video capacity depending on the LTE system, intending to improve video plans over time to provide the best possible user experience. Including, [10] used the NS-3 simulator as an environment of LTE networks. But, the performance of downlink package preparation measures in LTE networks is available. As well as, [11] used NS-3 simulation to obtain a performance and analysis of the user's situation. Additionally, an enhancement of multimedia streaming over LTE infrastructures using different techniques was achieved. In contrast, packet should be divided and scheduled into resource assignment techniques and assigned to all users [12]. Also, [13] deliberated a selection of the pathway based on video source according to the user satisfactions. Conversely, it used two metrics to find out a performance of packet delay and its number.

Implementation of WebRTC Video Conferencing

A lab was generated to plan and the appliance of WebRTC video between browsers over a 4G network (Ethernet and Wireless). As demonstrated in Figure (3), the signalling network was created based on the customer and server establishing and hanging out communication in the middle of peers. This experiment was completed as described below.
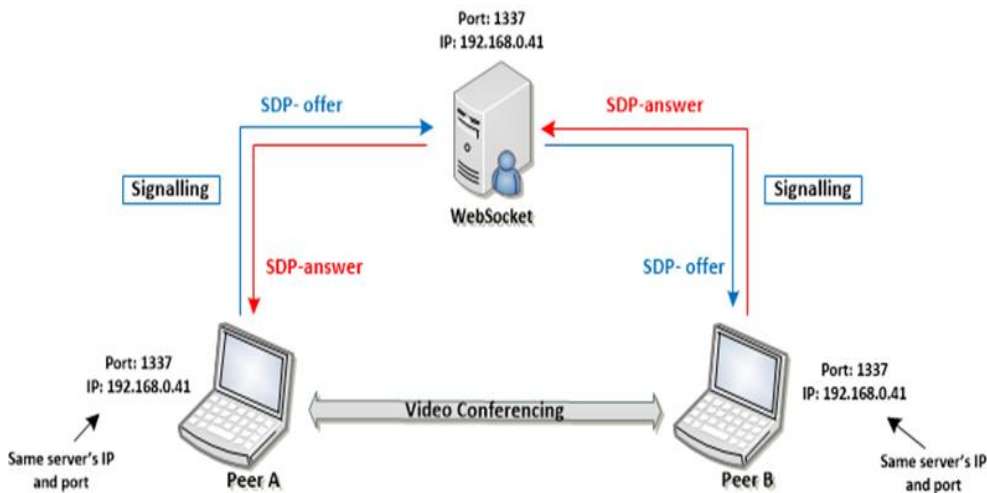

Figure 3 The architecture of video conferencing

The primary web browser was built with the JavaScript programming language and Notepad++. Peer 1 maintains the "local description using MediaStream" at first, which can be gotten with the "navigator.getUserMedia" technique, next a web browser will ask for authorization to contact the camera and microphone. When the authorization has been established, a camera will be activated, and the operation will be ready for peer 2 to connect. Peer 2 also requests "getUserMedia and share the

camera and microphone.". It has been arranged "RTC Peer Connection method for peers" together separately to hold the flooding of video and to display the peers' connection. Therefore, the caller uses "peerConn.create Offer" by "peerConn.set Local Description (offer)" methods to make an offer and direct it to another peer over a signalling service. Furthermore, the receiver receives the offer; so it needs to produce "an answer using peer Conn.set Remote Description, peer Conn.create Answer by peer Conn.set Local Description (answer) to reply it back to the server to be sent to the caller". Finally, the caller uses"peerConn.set Remote Description" method to catch the distant explanation correlated to a connection. Additionally, create "RTC Data Channel" object for multi-connection data between peers using "RTC Peer Connection. create Data Channel () method". Figures (4,5,6&7), show screenshots from the real-time implementation.
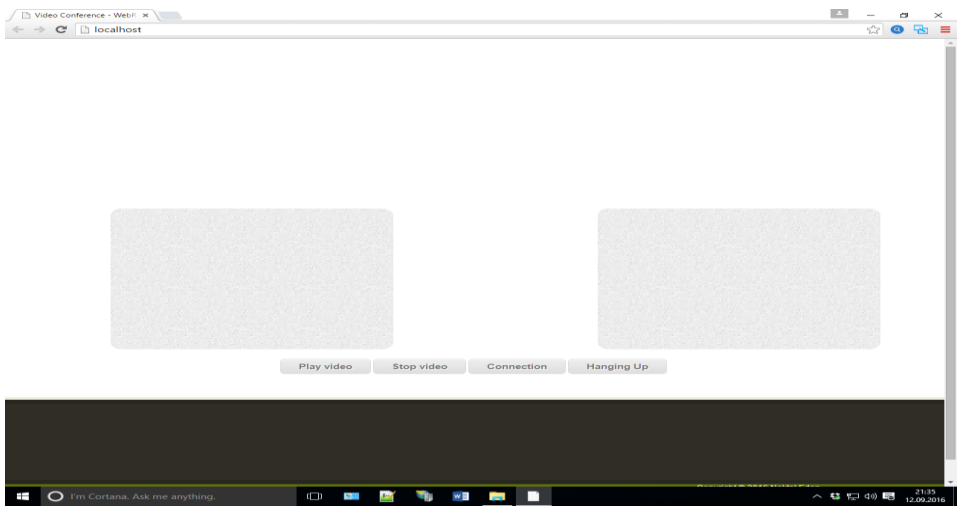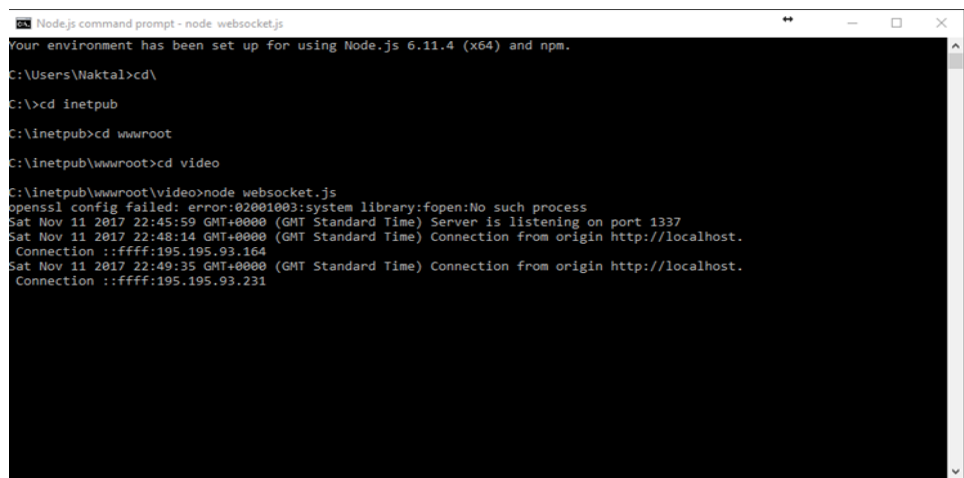

Figure 4 the main browser.

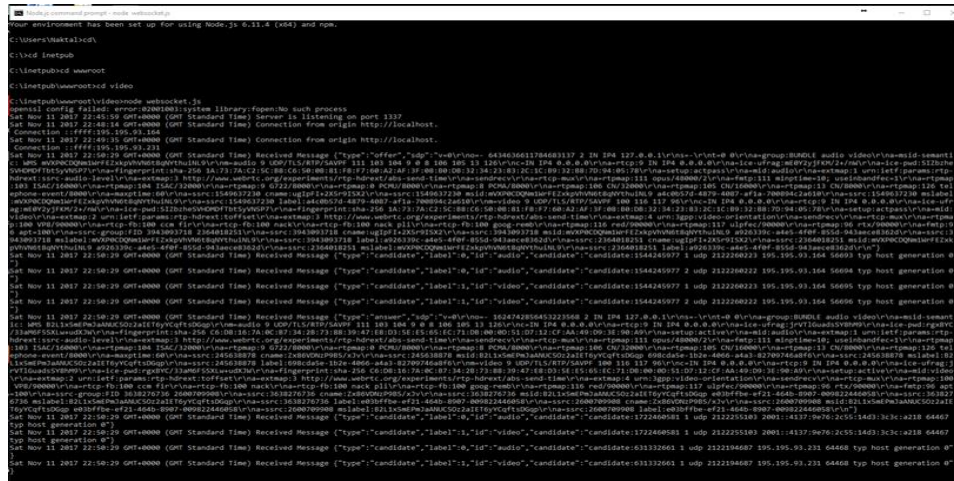**Figure 5 Construction of 1 & 2 peers and the WebSocket server**

Figure 6 A construction establishing between peers using WebSocket server
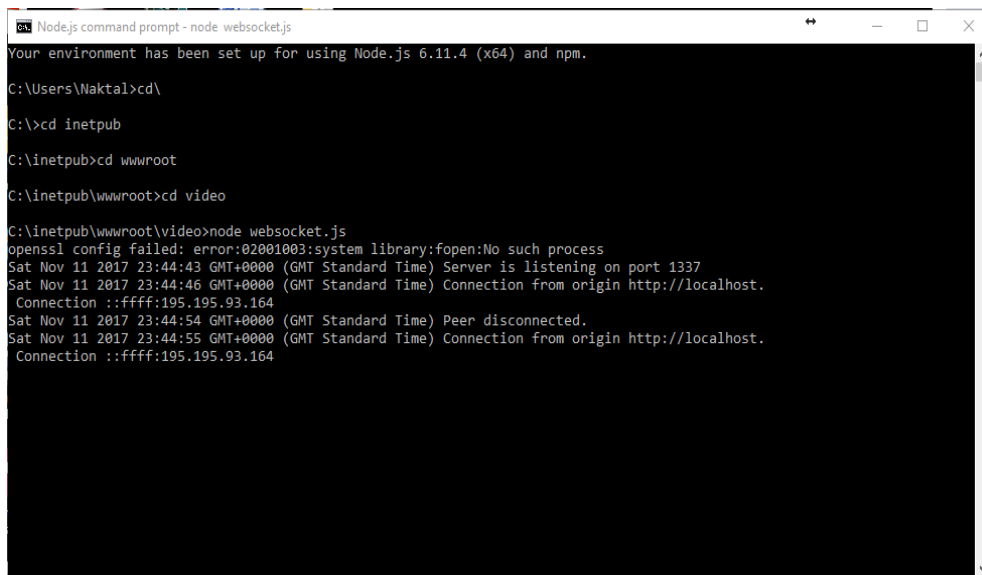

Figure 7 Disconnection between two peer and WebSocket server

Evaluation

Evaluating information supports researchers to reflect censoriously about the consistency, rationality, correctness, consultant, and suitability, the fact of view and unfairness of information foundations. As long as, books, articles, or website competitions author search conditions does not mean that it is essentially a dependable source of information. Based on the requirement of "CPU performance and bandwidth consumption, which can influence video running" [14], it has attentive on their valuation which can be brief as follows:

1. Bandwidth Consumption

The contact was obtained at various durations, such as five and ten minutes over (Wired and WiFi) of 4G network, as indicated in the table (1), and the announcement has demonstrated the great quality of speech and video.

Table (1), demonstrates the bitrate between peers by Wired and WiFi of 4G network. "The unit of bandwidth is kbit/s".

Network	Through	Bandwidth	
		Time	**Audio**
4G	Wired	Five minutes	76.793
		Ten minutes	91.578
	WiFi	Five minutes	87.423
		Ten minutes	98.849

2. CPU Presentation

CPU has a key outcome on WebRTC video conferencing. It switches a great capacity outstanding to numerous foundations distribution and getting the videos at a similar period. The CPU restrictions have an emotional impact simply the consumer with the compact CPU practice [15]. The CPU job was restrained by the "task manager tool" of "Windows 10" on the peer, which is together of user and server. The study is focused on five minutes through (Wired & Wifi) of 4G network, The performance can be exposed in diagrams (1 and 2).
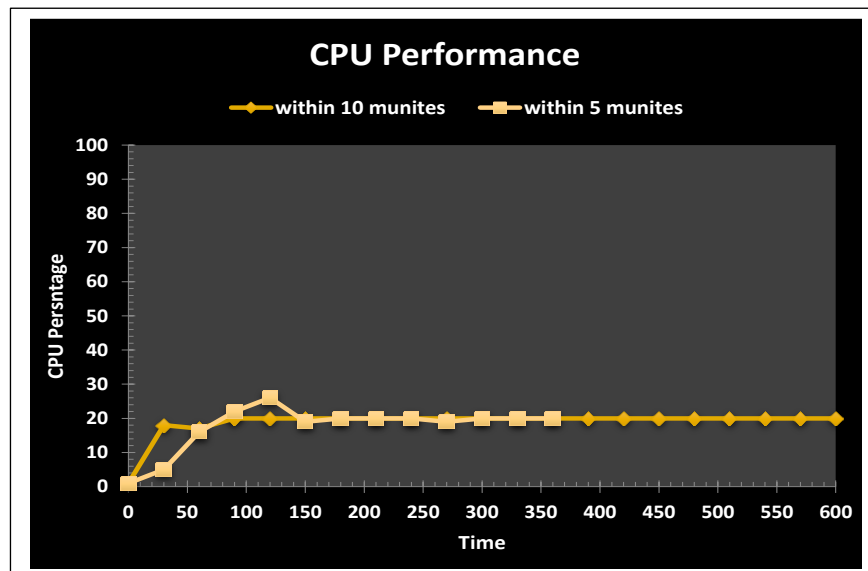
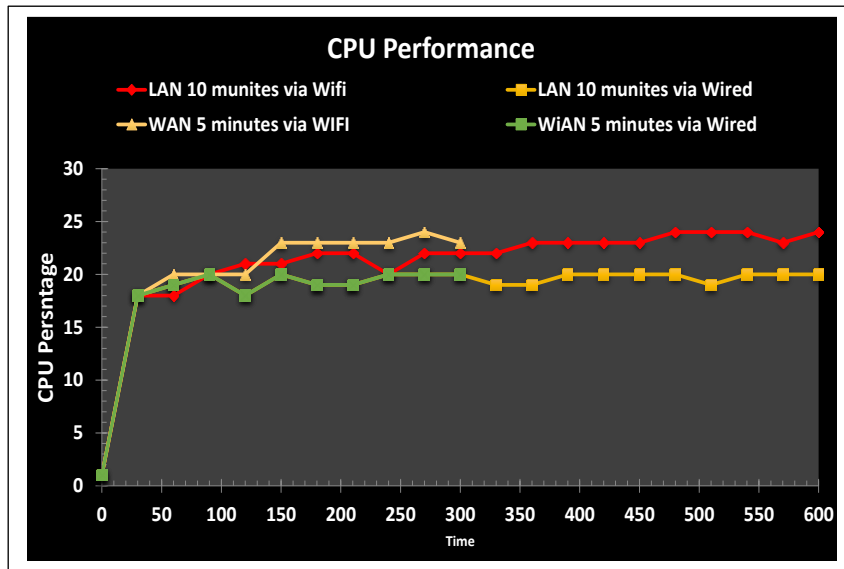**Diagram 1 the CPU performance over (wire) of 4G for 5 & 10 minutes**

Diagram 2 the CPU performance over (Wired and WiFi) for 5 & 10 minutes

3. Signalling Mechanism

The used signalling mechanism based on the Jode.js platform has been analysed and drowned. So, figure (2), shows the data exchange between peer 1 and peer 2 from the beginning to the end step by step. Thus, the applied mechanism was effective and operating excellent. As stated before, the signalling operation expenditures the WebSocket mechanism. Also, it prepared four categories of the regulator posts: "creator", "sending media stream", "peerChannel" and "exchange SDP". In the establishment, user 1 sends the "invitation" to the server and next the message controller "will get access media stream,", so now it will produce the peer's media stream from the server. Now user 1 waits till user B replies. Next, user 2 will direct an answer signalling communication to ensure that is it the initiator or not. Then, it will grow "peer Channel" and "contracted access media stream". Following that, everyone can begin launching a peer-to-peer structure. The "SDP (Session Description Protocol)" setup allows information such as "RTCP (Real-Time Control Protocol)", and very related things that can be employed in the broadcasting meeting to be transmitted. Figure (2) shows how communication between peers is validated from the time peer A initiates a request to the time peer 2 responds. It also shows how the plan is put together, with user A sending the SDP offer and ICE candidates to the server, which then passes them on to user 2. User 2 will direct the SDP response and ICE applicants to the server, which will then be forwarded to user 1.
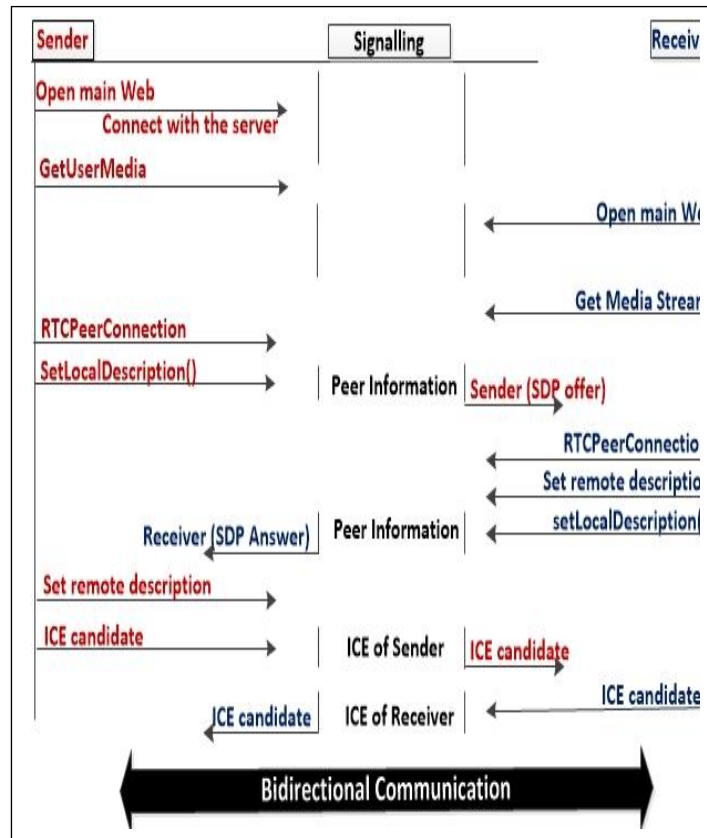
Figure 2 the data flow of the connection of two users

This work's signalling channel was created on both the client and server sides to enable peer communication, resulting in a successful job. The created signalling channel, on the other hand, describes the process of establishing, controlling, and disconnecting a communication, as well as allowing two peers to start video conferencing with one another. This signaling was used to send four different kinds of data: Information on how to open, close, and change a communication session is stored in the session control information. Session control messages are used to notify errors when one of the peers disconnects.

1. Network data: it discloses the location of peers using IP address and port so that the caller can locate the callee.
2. Determined media data: to limit which codecs and media kind the caller and the callee require in common.

In the SDP architecture, this signalling transmitted information between peers utilizing offer and answer. Thus, it displays any peer able to start or finish a meeting, ending self/distant stream, contribution bi-directional video conferencing and exit or replying to the meeting. However, it guarantees that this organization of statement is prepared to make a WebRTC presentation.

Conclusions

WebRTC was proposed as a new typical for enabling RTC between users who use various browsers devoid of the need for additional software. Nonetheless, adopting WebRTC faces a big challenge: it

does not provide a statement average in the middle of different browsers, which stops WebRTC from operating properly to discordancy concerns with multi connections. This work used the WebSocket protocol as a server to totally interact between two separate browsers, overcoming the aforementioned challenge. It also developed and implemented WebRTC video connection, which allows for bi-directional communication over a variety of networks, including LAN and WAN networks (Wired and Wi-Fi). The physical implementation was thoroughly examined in terms of CPU performance, bandwidth rate, and QoE. In the future, we'll try to use Socket.io to construct a WebRTC signalling mechanism for an unlimited number of peers and spread on diverse topologies like mesh and star. In addition, WebRTC video will be compared to the most used procedures in Video calling.

Acknowledgements

References

[1] M. Phankokkruad and P. Jaturawat, "An Evaluation of Technical Study and Performance for Real-Time Face Detection Using Web Real-Time Communication," in International Conference on Computer, Communication, and Control Technology (I4CT), 2015, no. I4, pp. 162–166.

[2] T. Sandholm, B. Magnusson, and B. A. Johnsson, "An on-demand WebRTC and IoT device tunneling service for hospitals," in International Conference on Future Internet of Things and Cloud (FiCloud), 2014, pp. 53–60.

[3] H. V. Cola. Cristian, "On multi-user web conference using WebRTC," in 18th International Conference on System Theory, Control and Computing (ICSTCC), 2014, pp. 430–433.

[4] A. Albas and G. Auguets, "WebRTC," Politècnica de Catalunya, 2016.

[5] L.-L. Tsahi, "WebRTC in the Enterprise," 2016.

[6] S. N. Malik and M. U. Saeed. Kamran, Sabir. Hussain, "A Comparison of SIP with IAX an Efficient new IP Telephony Protocol," in International Conference of Engineering and Emerging Technology, 2015, no. August, p. 7.

[7] A. S. Karl. Bissereth, Billy B. L. Lim, "An Interactive Video Conferencing Module for e-Learning using WebRTC," in Internetional Conferences, 2014, pp. 1–4.

[8] B. Błaszczyszyn, M. Jovanovic, and M. K. Karray, "Quality of real-time streaming in wireless cellular networks - Stochastic modeling and analysis," IEEE Trans. Wirel. Commun., vol. 13, no. 6, pp. 3124–3136, 2014.

[9] S. Singh, O. Oyman, A. Papathanassiou, D. Chatterjee, and J. G. Andrews, "Video capacity and QoE enhancements over LTE," IEEE Int. Conf. Commun., no. March, pp. 7071–7076, 2012.

[10] S. F. M. Vaser, "4G LTE architectural and functional models of Video Streaming and VoLTE services," in 2015 Seventh International Conference on Ubiquitous and Future Networks, 2015.

[11] Zaheduzzaman Sarker; Varun Singh; Colin Perkins, "An evaluation of RTP circuit breaker performance on LTE networks," in 2014 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), 2014.

[12] Huda Adibah Mohd Ramli; Kumbesan Sandrasegaran; Riyaj Basukala; Rachod Patachaian, "Resource allocation technique for video streaming applications in the LTE system," in The 19th Annual Wireless and Optical Communications Conference (WOCC 2010), 2010.

[13] Daniele Munaretto; Marco Zanforlin; Michele Zorzi, "Performance evaluation in ns-3 of a video delivery framework for next generation cellular networks," in 2013 IEEE International Conference on Communications Workshops (ICC), 2013.

[14] J. Nurminen, A. Meyn, and Eetu. Jalonen, "P2P media streaming with HTML5 and WebRTC," in Conference on Computer Communications Workshops (INFOCOM WKSHPS), 2013, no. 4, pp. 1–2.

[15] L. O. D. N. Eirik. Fosser, "Quality of Experience of WebRTC based video communication," Norwegian University of Science and Technology, 2016.