WEBRTC ON MULTI-PARTY COMMUNICATION TO LOWER VIDEO STREAMING TRAFFIC

YI-LUN CHEN, JEANNE CHEN, TUNG-SHOU CHEN

Departments of Computer Science and Information Engineering, National Taichung University of Science and Technology
E-mail: 1s18023104@nutc.edu.tw, 2jeanne@nutc.edu.tw, 3tschen@nutc.edu.tw

Abstract— Most media stream servers use peer to peer (P2P) network infrastructure to resolve heavy load problem such as in WebRTC. WebRTC is characterized by a flexible signaling protocol method. To implement multi-party communication on this platform is relatively complex and difficult. In WebRTC, multi-party communication requires more bandwidth which increases by new peers. In this paper, we proposed a new WebRTC flow control mechanism, called the adaptive peer traffic control WebRTC (APTC WebRTC) for multi-party communications. Experimental results showed that APTC WebRTC can reduce the traffic for each peer in the multi-party communication to reduce local peer bandwidth hogging. At the same time, there is no video traffic consumption at the server-side of APTC WebRTC. This reduces the need for a large number of servers and costs are also recused.

Keywords— WebRTC, Peer to Peer, Multi-party Communication, Adaptive Traffic Control.

I. INTRODUCTION

Due to advancement in internet technology, most video communication software has been developed, such as Zoom, BigBlueButton (BBB), and WebRTC [1][3][4][5]. WEBRTC supports a variety of modules, which may be used to handle different network environment signaling connection management modules [8]. It does not provide signaling protocol specification. Therefore, it is combined with signaling protocol in order to improve compatibility. Furthermore, multi-party communication is difficult and complex to process. Examples of which include XMPP[6], SIP[2] and others.

There are several ways to use signaling protocol in WebRTC. Adeyeeye et al. [9] implemented a three party communication for WebRTC. Signaling is accomplished by using SIP via WebSocket and JSON via XMLHttpRequest (XHR). After comparing the overheads for these two methods, JSON via XHR showed less overhead. The current browser supports WebSocket using simple and easy method to achieve multi-party communication, such as JSON via WebSocket. Video and voice communications will generate increasing amounts of streaming data. In multi-party communications, these streaming data will consume a large amount of network bandwidth [7]. Ng. et al. [10] proposed the P2P-Multipoint Control Unit (P2P-MCU) server method to share the bandwidth and CPU usage when each traditional WebRTC participates in the multi-party video chat. However, P2P-MCU results in some delays on the client side. The server requires higher computing resource and bigger network bandwidth. When there is a new peer connection added to the network, the P2P-MCU server will require more CPU and network bandwidth.

This study proposed an adaptive peer flow control mechanism, called Adaptive Peer Traffic Control WebRTC (APTC WebRTC), to reduce the multi-party communication traffic consumption of traditional WebRTC, and to implement the signaling protocol through a javascript plugin. Comparisons on the multi-party communication traffic will be made with Zoom and BBB. This also includes the relationship between the number of peer connections and traffic consumption in APTC WebRTC.

II. THE PROPOSED APTC WEBRTC METHOD

WebRTC video chat is exchanging video streaming data between peers in the network. For the existing peers, newly added peers will consume more traffic for all peers.

In this paper, we propose an adaptive peer traffic control based multi-party communication WebRTC (APTC WebRTC). In APTC WebRTC, the network resolution and bandwidth are adjustable based on increase or decrease upload/download traffic. Fig. 1 shows the comparison diagram of bandwidth requirement for multi-party communication between the traditional WebRTC and APTC WebRTC. Each shows four peer connections. The arrow paths displayed the streaming data transmissions. The thicker arrow paths showed that the traditional WebRTC requires significantly larger bandwidth for its streaming data.

Fig. 1. Multi-party communication streaming data transmission.
The thinner arrow paths in APTC WebRTC imply smaller bandwidth requirement for the streaming data. The directional arrows pointing towards each peer will be noted as branches. For example, Fig. 1 showed the number of branches for each peer is 3 and the total upload/download bandwidth is the sum of the three upload/download bandwidth paths. As shown in Fig. 1, the number of branches in traditional WebRTC is increased to 3 by adding a new remote peer to the network. The total bandwidth of the local peer is increased three times from the origin. The changes in the bandwidth are as shown in Fig. 2.

Fig. 2. Traditional WebRTC local side bandwidth changes.

If remote peers are increased to \( n \) then the number of branches for each peers will be increased to \( n \). When the value of \( n \) increases, the upload and download traffic will grow up to \( n \) times. The limited bandwidth network environment for general user, cannot afford high traffic consumption. Fig. 3 shows the diagram traditional WebRTC change to \( n \) remote peers.

Fig. 3. Traditional WebRTC change to \( n \) remote peers.

Fig. 4 shows the diagram of APTC WebRTC changing bandwidth and resolution. Remote peer in (1) showed maximum resolution with one peer only. However, when (2) increases to two remote peers to start a multi-party communication, the resolution of each peer have to be readjusted.

Fig. 4. Adjusting resolution of each remote peer with new peer addition.

After readjusting resolution, upload/download bandwidth is also reduced for each remote peer so as to reduce network traffic consumption. Fig. 5 shows APTC WebRTC adjustments of bandwidth when there are three remote peers.

Fig. 5. APTC WebRTC adjustments of remote peer bandwidths with new peer additions.

Fig. 6 shows the system control flow for APTC WebRTC.

Fig. 6. Flow of adaptive peer traffic control WebRTC (APTC WebRTC).

When WebRTC is multi-party communication, the local peer traffic consumption is related to the branch of remote peers. Fig. 7 shows the relationship between traffic and branches. In comparison to the other methods, the line curve for traditional WebRTC (line tagged with triangles) reveals that it consumes the most amount of traffic. The diamond tagged line curve of trial APTC WebRTC is generated by the simulation with fixed traffic. The rectangle tagged line curve is the empirical APTC WebRTC results of possible traffic consumption in relation to the number of branches.

Fig. 7. Relationship between branches and bandwidths.

In this study, implementation of multi-party communication APTC WebRTC is based on RTCMulticonnection. Fig. 8 shows a flow chart for this system. The numbered markings show the
sequence of processing the communication for each peer. (1) shows any peer request to Web server for web content. After the page is loaded to the peer, (2) shows the establishment of peer communication through the signaling server. When peer in (3) received the location of the remote peer, the streaming media transmission channel is established via a signaling protocol.

![Diagram](image)

**Fig. 8. System flow of APTC WebRTC.**

### III. EXPERIMENTAL RESULTS AND ANALYSIS

In this study, APTC WebRTC is the agent for video communication traffic consumption reduction via RTCMulticonnection library to handle signaling protocol of multi-party communication. Fig. 9 shows the communication process.

![Image](image)

**Fig. 9. APTC WebRTC multi-party communication.**

In the experimental testing a total of five PCs were used and the network environment as shown in Fig. 10. The network environment is Ethernet network through switches connected to the router and then to the server.

![Image](image)

**Fig. 10. Network experimental environment.**

The experimental testing was divided into three parts. The test time is 30 minutes for each part. At different stages of the experiment each PC as a peer was added to the network to begin communication. First is the test for traffic consumption of peer participation in multi-party communication relationship between the number of communicating peers to traditional WebRTC and APTC WebRTC. Second is the test of the traffic consumption relationship to the number of peer participation in client-side of multi-party communication to Zoom, BBB and APTC WebRTC. Third and final part is the test traffic consumption relationship to the number of client connected at server-side. Part 2 and part 3 of experimental testing will be not performed for two peers. Therefore, testing starts at three peers.

Part 1 of the experimental results is as shown in Fig. 11. When communicating peers are increased to five, traditional WebRTC showed upload/download traffic increased to 5569 Kbps and 5558 Kbps. APTC WebRTC upload/download traffic was 1680Kbps and 1632Kbps because the number remote peers detected in the network requires adjustment to bandwidth and resolution. The new peer addition reduces traffic increment proportion. On the other hand the local peer has less total bandwidth change.

![Graph](image)

**Fig. 11. Relationship between peers and traffic of multi-party communications for the traditional WebRTC vs APTC WebRTC.**

Part 2 of peer and traffic consumption at client-side of Zoom, BBB and APTC WebRTC is as shown in Fig. 12. The number of the peer participation for Zoom, BBB and APTC WebRTC is increased from three to five with download traffic proportional changes as 49.36%, 125.28% and 33.44%. The total download bandwidth of APTC WebRTC is significantly lower than the other two.

![Graph](image)

**Fig. 12. Zoom, BBB and APTC WebRTC relation between multi-party communications and traffic.**
The third and final part testing traffic relationship between the number of clients at server-side of Zoom, BBB and APTC WebRTC is as shown in Fig. 13. When clients of Zoom and BBB are increase to 5 the upload traffic was 5353Kbps, and 5213Kbps. This is due to the client-server based network which require transfer for the streaming data clients through the server. Peers content request/response or signaling packets pass through the server so that server side only has a small flow. The reason is due to APTC WebRTC based P2P network architecture.

![Fig. 13. The number of clients connected and traffic on the server-side.](image)

APTC WebRTC improves traffic consumption of multi-party communication. There is less bandwidth increases when new peer is added to the network. APTC WebRTC has lower download increase proportion of traffic then Zoom and BBB. The server-side has less traffic consumption for the content request/response and signaling.

**CONCLUSIONS**

It is relatively difficult and complex to implement multi-party communication protocol for the traditional WebRTC. More connected peers required higher network bandwidth in the traditional WebRTC in multi-party communication. This study proposed the APTC WebRTC to reduce the video chat traffic, and to process the multi-party communication signaling protocol via a javascript plugin. APTC WebRTC reduces network traffic by adjusting the resolution and bandwidth of all connected peers accordingly to the number of peers in the network. Experimental results showed that APTC WebRTC bandwidth consumption is significantly less than the traditional WebRTC at a local peer with additional peers. Comparisons with Zoom and BigBlueButton, when the client-side connected peers are increased from three to five, showed the proportion of traffic increase is much smaller than the other two for APTC WebRTC. APTC WebRTC generated less traffic consumption at server-side to reduce the device costs of server-side.

In conclusion, APTC WebRTC can reduce traffic consumption, and is suitable for use in multi-party video communication.

**REFERENCES**