Enhancing VoIP BW Utilization over ITTP Protocol

AbdelRahman H. Hussein[1]*, Mahran Al-Zyoud[2]
Dept. of Computer Science
Al-Ahliyya Amman University (AAU)
Amman, Jordan

Kholoud Nairoukh[3], Sumaya N. Al-Khatib[4]
Dept. of Networks and Information Security
Al-Ahliyya Amman University
Amman, Jordan

Abstract—The revolution of Voice over Internet Protocol (VoIP) technology has propagated everywhere and replaced the conventional telecommunication technology (e.g. landline). Nevertheless, several enhancements need to be done on VoIP technology to improve its performance. One of the main issues is to improve the VoIP network bandwidth (BW) utilization. VoIP packet payload compression is one of the key approaches to do that. This paper proposes a new method to compress VoIP packet payload. The suggested method works over internet telephony transport protocol (ITTP) and named Delta-ITTP method. The core idea of the Delta-ITTP method is to find and transmit the delta between the successive VoIP packet payloads, which is typically smaller than the original VoIP packet payload. The suggested Delta-ITTP method implements VoIP packet payload compression at the sender side and decompression at the receiver side. During the compression process, the Delta-ITTP method needs to keep some values to restore the original VoIP packet payload at the receiver side. For this, the Delta-ITTP method utilizes some of the IP protocol fields and no additional header is needed. The Delta-ITTP method has been deployed and compared with the traditional ITTP protocol without compression. The result showed that up to 19% BW saving was achieved in the tested cases leading to the desired enhancement in the VoIP network BW utilization.

Keywords—Voice over IP (VoIP); VoIP protocols; Internet Telephony Transport Protocol (ITTP); payload compression; bandwidth utilization

I. INTRODUCTION

Voice over Internet Protocol (VoIP) is everywhere. There is a very high growing demand for VoIP in just about all companies and even by the normal people [1][2]. However, two main dilemmas need to be addressed for VoIP to replace the traditional telecommunication systems. The first one is the faint quality of the VoIP calls when compared with the traditional telecommunication system calls. This is because VoIP works over IP networks which, as opposite to the traditional telecommunication systems, do not provide a dedicated channel for the VoIP call [3][4]. The second dilemma is the failure to make the best use of VoIP network bandwidth (BW) [4][5]. This is mainly because the VoIP codecs produce a very small voice frame (VoIP packet payload) in order to avoid exceeding the acceptable 150ms delay of VoIP calls. The codec voice frame is usually 10 to 30 bytes in length while the size of the protocols (VoIP packet header) that used to carry the voice data can reach up to 40 bytes. Therefore, only 20% to 43% of the VoIP network BW is used to carry the VoIP data while the remaining share of the BW is wasted to carry the VoIP packet header. Clearly, this is a very high ratio of waste of the VoIP network BW [6][7].

*Corresponding Author

There are several protocols used by VoIP technology to carry the VoIP data. They are typically classified into signaling protocols and media transfer protocols. The signaling protocols are used to establish and negotiate the parameters to be used in a call. The two main examples of VoIP signaling protocols are SIP and H.323 protocol [8][9]. The media transfer protocol is used to convey the voice data, after establishing the call, from point to point over the VoIP network. The 12-byte Real-time Transport Protocol (RTP), 6-byte Internet Telephony Transport Protocol (ITTP), and 4-byte Inter-Asterisk eXchange (IAX) are the main examples of media transfer protocols. Both RTP and IAX take the help of the 8-byte User Datagram Protocol (UDP) to be able to convey the voice data, while ITTP is able to carry the voice data by itself [10][11][12]. As mentioned earlier, adding these protocols along with the 20-byte IP protocol to the small VoIP packet payload leads to a considerable amount of BW waste. Table I shows the wasted BW, caused by media transfer protocols along with the IP protocol. The wasted BW is calculated by dividing the protocol size over packet size (payload and protocol).

The VoIP researchers have followed several ways to deal with the wasted BW resulting from the VoIP applications. These approaches include VoIP packet grouping, VoIP packet payload compression, and VoIP packet header compression [13][14][15]. In this research paper, we propose a new method to save the VoIP network BW by compressing the VoIP packet payload. The suggested method computes the delta of the consecutive VoIP packets payload with a specific mechanism to compress VoIP packet payload. Additionally, the suggested method utilizes some of the IP protocol fields to serve and save the values needed by the compression algorithm. Thus, no extra header needs to be added to the VoIP packet.

TABLE I.   CONSUMED BW BY MEDIA TRANSFER PROTOCOLS AND IP PROTOCOL

Protocol Name	Protocol Size	Wasted BW(%)		
		10-bytes payload	20-bytes payload	30-bytes payload
IAX/UDP/IP	32	76.1%	61.5%	51.6%
RTP/UDP/IP	40	80%	66.6%	57.1%
ITTP/IP	26	72.2%	56.5%	46.5%

The hierarchy of the paper is as follows: Section 2 spotlights one of the grouping methods that addresses the BW utilization problem. Section 3 discusses the internal operations and procedures of the suggested method. Section 4 shows the evaluation parameters of the suggested method and investigates its robustness against the traditional ITTP method. In the end, Section 5 summarizes the findings.

II. RELATED WORKS

The issue of VoIP BW utilization has been investigated by many researchers. This section discusses some of the methods that were suggested to improve VoIP network BW utilization.

Teymoori, Peyman, et al. suggested a packet grouping method to improve the VoIP network BW in wireless LAN (WLAN). To achieve this, the suggested method grouped the incoming VoIP frames in one layer 2 header. Besides packet grouping, the suggested method focused on the ideal size of the resulted grouped packet, particularly for the delay-sensitive applications in the high speed WLAN. Thinking about the impact of packet grouping on delay, Teymoori, Peyman, et al. suggested an investigative model to get the ideal grouped packet size with respect to delay limitations forced by real-time applications. Specifically, the authors defined a packet grouping scheme as a constrained convex optimization issue to expand the overall WLAN throughput. First, Teymoori, Peyman, et al. displayed the channel access delay of nodes using the IEEE 802.11n packet grouping scheme. Second, through a general strictly concave network utility function, the optimization problem is constructed and the distributed solution is suggested to reach at its ideal point. The result proved that the suggested model reached the ideal point within little iterations in long term with no extra overhead. Moreover, delay requirements of nodes are actually fulfilled showing the precision of the suggested investigative model [16].

Sze et al. suggested a method that performs both header compression and packet grouping. The suggested method involved a module located at the sender device and another one located at the receiver device. The first module stripes the header from the data, compresses the RTP header based on certain mechanism, attaches the compressed RTP to the data, groups the resulted frames in one UDP/IP header, and transmits the resulted large packet to the receiver. The frames are grouped in one large packet until reaching a specific delay limit. The receiver side module de-groups the received large packet, separates the compressed RTP from the data, restores the original RTP header, attaches the RTP header along with UDP/IP header to the data, and sends the resulted original packet to its final destination. Besides header compression and packet grouping, the suggested method modifies the initial process of establishing the VoIP call. This modification is necessary for the sender to inform the receiver that the transmitted packets are grouped together and not the normal VoIP packet. The suggested method was tested and evaluated against the traditional method without grouping. The results showed an improvement in the BW utilization that reached up to 300% in the tested scenarios [17].

All of the previously mentioned methods were deployed over RTP/UDP/IP protocols header. In contrary to those methods, Abualhaj, M. et al suggested a grouping method, called ITTP-Mux that works over the ITTP protocol header. The method, involved two modules located at the sender side gateway (named S-Mux) and the receiver side gateway (named S-DMux). The main purpose of the S-Mux is to group the packets that share the same path to the same gateway together in one IP header rather that a separate IP header to each packet. Before being grouped in one large packet with one IP header, a small header is attached to each of the mini-packets to be grouped in the large packet. The main purpose of the S-DMux is to segregate the large packet to the mini-packets, by inspecting the small header, and restore the original VoIP packet. The restored VoIP packets are then transmitted to their final destinations. The suggested method was tested and evaluated against the traditional ITTP method without grouping. The result showed an improvement in the BW utilization by up to 29.1% in the tested scenarios [18]. Another grouping method that works over ITTP protocol was in 2019. However, the suggested method, named CA-ITTP, combines between VoIP packets grouping and VoIP packets payload compression. Similar to the ITTP-Mux method, CA-ITTP method involved two modules located at the sender side gateway (named SCA) and the receiver side gateway (named RCD). The SCA module performs the same functions of the S-Mux in addition to payload compression based on a certain algorithm. The RCD module performs the same functions of the S-DMux in addition to payload decompression. The compression algorithm reduces the size of the VoIP packet payload based on a specific mathematical mechanism, which produces a smaller payload than the original payload. The decompression algorithm reinstates the size of the VoIP packet payload at the receiver side gateway. The suggested CA-ITTP method was tested and evaluated against the traditional ITTP method without grouping or compression. The result showed that the BW waste was reduced by half in the tested scenarios [19]. In this paper, we propose a new method to compress the VoIP packet payload, called Delta-ITTP. The suggested Delta-ITTP method computes and sends the variation of the consecutive VoIP packets payload, which is usually smaller than the original VoIP packet payload, using a specific delta algorithm. In addition, unlike the afore-discussed method, no additional header is needed since the suggested method utilizes some of the IP protocol fields to serve the purpose of the compression algorithm. The following section discusses the details of the Delta-ITTP method.

III. THE SUGGESTED DELTA-ITTP METHOD

This section discusses the internal algorithm of the suggested Delta-ITTP method. The main purpose of the Delta-ITTP method is to enhance the VoIP network BW utilization. The desired utilization is achieved by shrinking the VoIP packet payload. The core shrinking idea used by the Delta-ITTP method is to consider the VoIP packet payload a digital number. By doing so, the VoIP packet payload of the consecutive packets can be subtracted from each other. The result of the subtraction is usually less in size than the VoIP packet payload. Then, the subtraction result replaces the VoIP packet payload and is sent to the receiver, saving the BW due to its smaller size. For demonstration, assume that the payload

of the first VoIP packet is X and the payload of the second VoIP packet is Y. The size of payload X is fifteen bits "111001100110101" and the size of payload Y is fifteen bits "110011101001001". Then the result of subtracting the payload Y from the payload X is thirteen bits 101111101100, which is smaller than the normal payload size (110011101001001), thus lessens the consumed BW. There is a set of operations that are performed at the sender side to shrink the VoIP packet payload, as discussed in section (III.A). At the receiver side, the VoIP packet payload needs to be filled in order to recover its initial shape and size as produced by the codec at the sender side. Section 3.2 discusses the operations to recover the VoIP packet payload initial shape and size.

A. Delta-ITTP Operations at Sender Side

The main aim of the suggested Delta-ITTP method is to shrink the VoIP packet payload and, thus, save the VoIP network BW. Several operations must occur to achieve this. First, dissociation of the VoIP packets payload from the VoIP packets header. Second, subtraction of the VoIP packets payload that follow each other. For example, assume there are two payloads x and y, where x is the payload of packet n and y is the payload of packet n+1. If the value, as a digital number, of x is greater than y, then subtract y from x. Otherwise, subtract x from y. If the subtraction result is less than y, then replace y by the subtraction result, otherwise, keep y value as it is. Clearly, the first packet's payload remains unchanged because no packets had arrived before it. Third, update the flag fields of the IP protocol in the VoIP packet header. The value of these fields is used to recover the VoIP packet payload initial shape and size as produced by the codec at the sender side. The process of updating these fields will be discussed below. Fourth, reconstruct the VoIP packet by associating the y packet payload to its header. Fifth, send the resulted VoIP packet to its destination. Fig. 1 illustrates the set of operations performed by the suggested Delta-ITTP at the sender side.

As mentioned above (Third operation), the flag fields of the IP protocol in the VoIP packet header need to be updated. There are three 1-bit flag fields within the IP protocol header. These fields are typically used to control the fragmentation process and identify the fragments of the large IP packets. However, the VoIP packets size is typically between 36 bytes to 56 bytes when ITTP protocol is used; thus, no fragmentation of the ITTP VoIP packets occur. Therefore, the three 1-bit flag fields within the IP protocol header are unused in case of VoIP over ITTP and remain zeros. Accordingly, the suggested Delta-ITTP method will use them to denote i) whether x is greater than y or the opposite and ii) whether y value is replaced by the subtraction result or kept as it is. Only two bits are needed to denote these two cases; therefore, the leftmost two bits of the three will be used. The first leftmost bit is used to denote whether x is greater than y or the opposite. Setting it to one denotes that x is greater than y and setting it to zero denotes the opposite. The second leftmost bit is used to denote whether y value is replaced by the subtraction result or kept as it is. Setting it to one denotes that y value kept as it is and setting it to 0 denotes that y value is replaced by the subtraction result. These values are used to

recover the VoIP packet payload initial shape and size as produced by the codec at the sender side. Fig. 2 shows the pseudocode of the subtraction process (Second operation) and updating of the IP protocol header flag fields (Third operation).
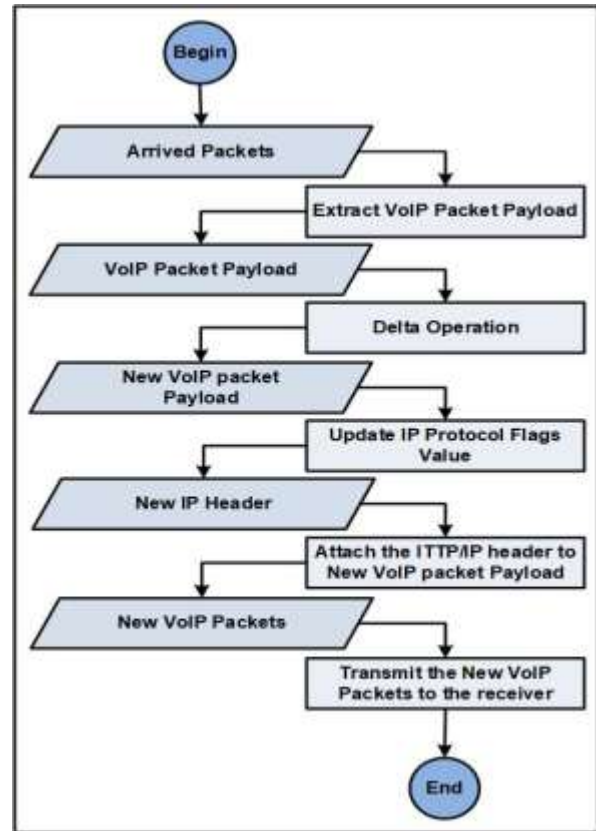

Fig. 1. Delta-ITTP Operations (Sender Side).

Delta Algorithm @ Sender Side

```
1   // X is packet n payload
2   // Y is packet n+1 payload
3   // FF First leftmost flag bit
4   // SF Second leftmost flag bit

5   if (X > Y) {
6       result= X – Y
7       FF = 1
8   else
9       result= Y – X
10      FF = 0
11  }
12  if (result < Y) {
13      Y= result
14      SF = 0
15  else
16      Y= Y
17      SF = 1
18  }
```

Fig. 2. Delta Algorithm (Sender Side).

B. Delta-ITTP Operations at Receiver

The VoIP packet payload must be recovered to its initial shape and size as produced by the codec at the sender side. The receiver side performs a set of operations to do so. First, dissociation of the VoIP packets payload from the VoIP packets header. Second, recovering the VoIP packet payload initial shape and size as produced by the codec at the sender side. This is achieved by performing the addition or subtraction operation with the previous VoIP packet payload, based on the IP protocol header flag field's value. For example, assume that there are two payloads x and y, where x is the payload of packet n and y is the payload of packet n+1. If the second leftmost bit of the IP protocol header flag fields is equal to one, then the VoIP packet payload equals y. However, if the second leftmost bit of the IP protocol header flag fields is equal to zero and the first leftmost bit of the IP protocol header flag fields is equal to zero, then add y to x to produce the VoIP packet payload initial shape and size. Otherwise, subtract y from x to produce the VoIP packet payload initial shape and size. Clearly, no operations are performed on the first packet's payload because it did not change at the sender side. Third, set the IP protocol header flag field's value to zero to avoid any misinterpretation by the receiver. Fig. 3 shows the pseudocode of recovering the VoIP packet payload initial shape and size (Second operation) and updating of the IP protocol header flag fields (Third operation). Fourth, reconstruct the VoIP packet by associating the y packet payload to its header. Fig. 4 illustrates the set of operations performed by the suggested Delta-ITTP at receiver side.

Delta Algorithm @ Receiver Side

```
1   // X is packet n payload
2   // Y is packet n+1 payload
3   // FF First leftmost flag bit
4   // SF Second leftmost flag bit

5   if (SF is equal to 1) {
6       Payload-Size= Y
7   else if (SF is equal to 0) {
8       if (FF is equal to 0) {
9           Payload-Size= Y + X
10      else if (FF is equal to 1)
11          Payload-Size= X - Y
12  }
```
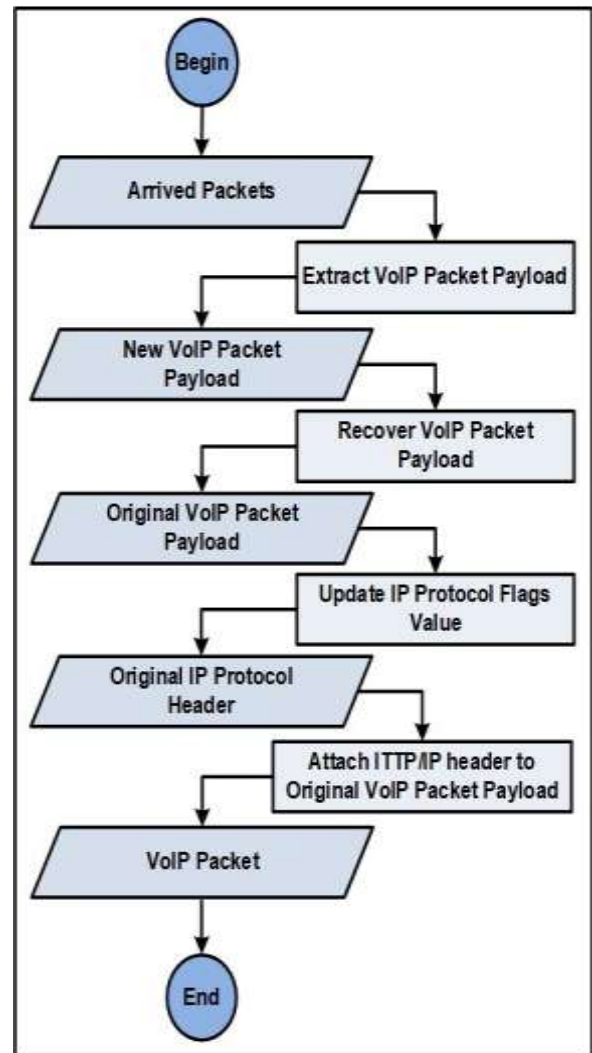
Fig. 3.   Delta Algorithm (Receiver Side).

Fig. 4.   Delta-ITTP Operations (Receiver Side).

IV. DELTA-ITTP RESULTS AND PERFORMANCE EVALUATION

This section discusses the suggested Delta-ITTP performance evaluation. The main aim of the suggested Delta-ITTP method is to shrink the VoIP packet payload; thus, saving the VoIP network BW. The VoIP network BW of the suggested Delta-ITTP method was investigated on the basis of three different metrics, namely, shrunk bits per VoIP packet payload, saved BW, and capacity improvement.

A. Shrunk Bits Per VoIP Packet Payload

This section discusses the payload shrunk bits when using the suggested Delta-ITTP method. The payload shrunk bits were investigated for five different streams, with a sample of seven packets for each, in comparison to the original VoIP packet payload size (O_Size). The size of the first VoIP packet payload of each stream is equal to the O_Size because no packets had arrived before it and, thus, no shrinking occurred. The rest of the VoIP packets' payloads, starting from the second packet, were shrunk with different values. The difference of the shrunk bits' value is due to the different distribution style of each VoIP packet payload. The more the

number of shrunk bits, the better the BW utilization [20]. Fig. 5 shows the payload shrunk bits of the suggested Delta-ITTP method.

B. Saved BW

This section discusses the saved BW ratio when using the suggested Delta-ITTP method. The saved BW ratio was investigated for 10 different streams. The BW saving was improved by approximately 12% to 19%; thus, enhancing the BW utilization [21]. The difference of the saved BW ratio between the 10 streams is due to the different distribution style of each VoIP packet payload. Fig. 6 shows the saved BW ratio of the suggested Delta-ITTP method.

C. Capacity Improvement

This section discusses the capacity improvement ratio when using the suggested Delta-ITTP method. Capacity is the number of calls running concurrently at a specific period. The capacity improvement ratio was investigated for 10 different streams. The capacity improvement was approximately between 3% and 7%; thus, enhancing the BW utilization. Again, the difference of the capacity improvement ratio between the 10 streams is due to the different distribution style of each VoIP packet payload. Fig. 7 shows the capacity improvement ratio of the suggested Delta-ITTP method.
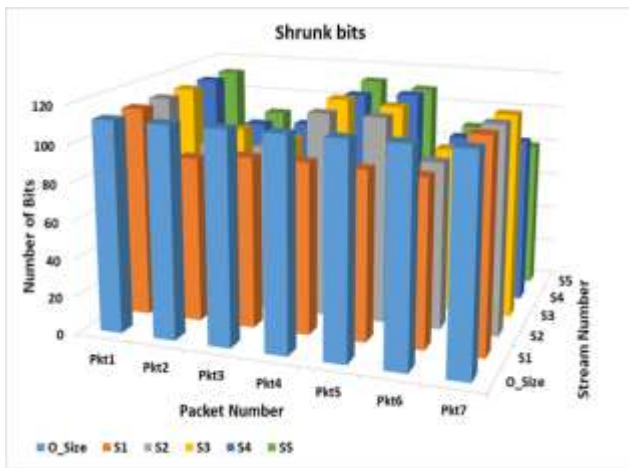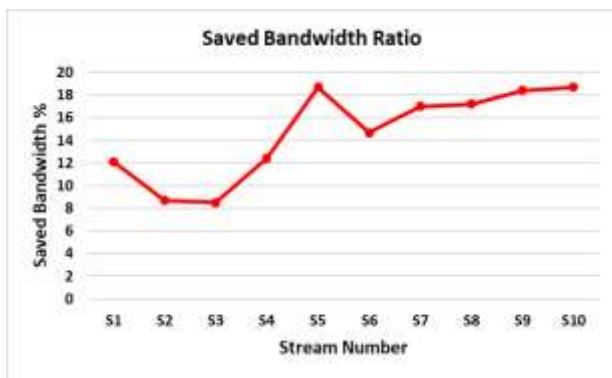

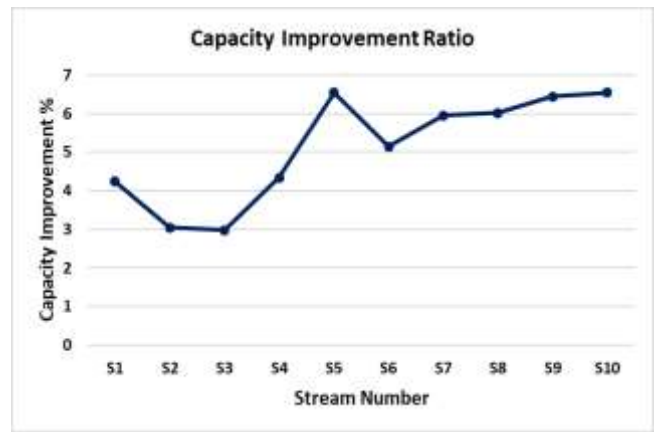Fig. 5. VoIP Packet Payload Shrunk Bits.

Fig. 6. Saved Bandwidth Ratio.

Fig. 7. Capacity Improvement Ratio.

V. CONCLUSION

VoIP technology has evolved rapidly to serve most of the big companies in private and public sectors. However, the BW utilization of VoIP networks needs to be improved. This paper suggests a new method, called Delta-ITTP, to improve VoIP networks BW utilization. The Delta-ITTP method achieves this by shrinking the VoIP packet payload using the suggested delta algorithm. The core idea of the delta algorithm is to find the delta of the VoIP packets payload that follow each other and send this delta, instead of the full VoIP packet payload. The Delta-ITTP method utilizes the IP protocol flag fields to keep the needed information by the delta algorithm; thus, no additional header is needed. The Delta-ITTP method was deployed and compared with the traditional ITTP protocol without compression. The results show that the number of shrunk bits varies, between the VoIP packets payload, based on the bits' pattern. The improvement in BW saving and capacity reached up to 19% and 7%, respectively. Therefore, the suggested Delta-ITTP method enhances the VoIP network BW utilization and achieves its design goal. In the future, we will deploy the suggested Delta-ITTP method with other VoIP network BW utilization methods, such as VoIP packets aggregation methods.

REFERENCES

[1] Y. Zhang, K. Xie, and O. Ruan, "An improved and efficient mutual authentication scheme for session initiation protocol," PloS one, vol. 14(3), pp. e0213688, 2019.

[2] M. M. Abualhaj, M. M. Al-Tahrawi, and S. N. Al-Khatib, "Performance Evaluation of VOIP Systems in Cloud Computing," Journal of Engineering Science and Technology, vol. 14(3), pp. 1398-1405, 2019.

[3] H.M. Mahdi, A.M. Suhail, A. Maaruf, H. H. AbdelRahman, and A.G. Muzafar, "Simulation and Analysis of Quality of Service (QoS) Parameters of Voice over IP (VoIP) Traffic through Heterogeneous Networks", International Journal of Advanced Computer Science and Applications (IJACSA), vol. 8, no. 7, 2017.

[4] T. T. Nguyen, and G. Armitage, "A survey of techniques for internet traffic classification using machine learning," IEEE communications surveys \& tutorials, vol. 10(4), pp. 56–76, 2008.

[5] M. M. Abualhaj, S. N. Al-Khatib, M. Kolhar, A. Munther, and Y. Alraba'nah, "Effective Voice Frame Pruning Method to Increase VoIP Call Capacity," TEM Journal, vol. 9(1), pp. 48-54, 2020.

[6] M. O. Ortega, G. C. Altamirano, C. L. Barros, and M. F. Abad, "Comparison between the real and theoretical values of the technical parameters of the VoIP codecs," 2019 IEEE Colombian Conference on Communications and Computing (COLCOM), pp. 1-6, 2019.

[7] J. M. Valin, T. B. Terriberry, C. Montgomery, and G. T. Maxwell, "A high-quality speech and audio codec with less than 10-ms delay," IEEE transactions on audio, speech, and language processing, vol. 18(1), pp. 58-67, 2009.

[8] A. Irshad, M. Sher, E. Rehman, S. A. Ch, M. U. Hassan, and A. Ghani, "A single round-trip sip authentication scheme for voice over internet protocol using smart card, "Multimedia Tools and Applications, vol. 74(11), pp. 3967-3984, 2015.

[9] S. Stelios Antoniou, Introducing VoIP Signaling - https://www.pluralsight.com/blog/it-ops/voip-signaling-protocols, July 2020.

[10] M. M. Abualhaj, M. M. Al-Tahrawi, and S. N. Al-Khatib, "A New Method to Improve Voice over IP (VoIP) Bandwidth Utilization over Internet Telephony Transport Protocol (ITTP)," Proceedings of the 2019 8th International Conference on Software and Information Engineering, pp. 192-195, 2019.

[11] G. Kaur, J. Kaur, S. Aggarwal, C. Singla, N. Mahajan, S. Kaushal, and Sangaiah, A.K, "An optimized hardware calibration technique for transmission of real-time applications in VoIP network,", Multimedia Tools and Applications, vol. 78(5), pp. 5537-5570, 2019.

[12] M. Spencer, B. Capouch, E. Guy, F. Miller, and K. Shumard, "Iax: Inter-asterisk exchange version 2 ," Internet Request for Comments, 2010.

[13] P. Fortuna, and M. Ricardo, "Header compressed VoIP in IEEE 802.11," IEEE Wireless communications, vol. 16(3), pp. 69-75, 2009.

[14] R. F. Aji, S. Yazidb, H. Suhartantoc, and I. Budid, "Throughput Fairness Using Packet Aggregation on 802.11 g Networks," 2016.

[15] Shambour, Qusai, et al. " Effective Voice Frame Shrinking Method to Enhance VoIP bandwidth Exploitation." International Journal of Advanced Computer Science and Applications (IJACSA), IEEE, vol. 11(7), pp. 313-319, 2020.

[16] Teymoori, Peyman, et al. "An optimal packet aggregation scheme in delay-constrained IEEE 802.11 n WLANs." 2012 8th International Conference on Wireless Communications, Networking and Mobile Computing. IEEE, 2012.

[17] H. P. Sze, S. C. Liew, J. Y. Lee, and D. C. Yip, "A multiplexing scheme for H. 323 voice-over-IP applications," IEEE Journal on Selected Areas in Communications, vol. 20(7), pp. 1360-1368, 2002.

[18] M. M. Abualhaj, "ITTP-MUX: An efficient multiplexing mechanism to improve voip applications bandwidth utilization," International Journal of Innovative Computing Information and Control, vol. 11(6), pp. 2063-2073, 2015.

[19] M. M. Abualhaj, "CA-ITTP: An efficient method to aggregate voip packets over ittp protocol," International Journal of Innovative Computing Information and Control, vol. 15(3), pp. 1067-1077, 2019.

[20] M. M. Abualhaj, et al. "PS-PC: An effective method to improve voip technology bandwidth utilization over ittp protocol." Cybernetics and Information Technologies (CIT) 20(3), pp. 147-158, 2020.

[21] M. M. Abualhaj, et al. "Multiplexing voip packets over wireless mesh networks: A survey." KSII Transactions on Internet & Information Systems 10(8), pp. 3728-3752, 2016.