

An innovative approach for enhancing capacity utilization in point-to-point voice over internet protocol calls

Mosleh M. Abualhaj¹, Ahmad Adel Abu-Shareha², Sumaya Nabil Al-Khatib³, Mohammad O. Hiari¹, Layth Al-Mahadeen⁴

¹Department of Networks and Cybersecurity, Faculty of Information Technology, Al-Ahliyya Amman University, Amman, Jordan

²Department of Data Science and Artificial Intelligence, Faculty of Information Technology, Al-Ahliyya Amman University, Amman, Jordan

³Department of Computer Science, Faculty of Information Technology, Al-Ahliyya Amman University, Amman, Jordan

⁴Department of Financial and Administrative Sciences, Al-Balqa' Applied University, Amman, Jordan

Article Info

Article history:

Received Jul 4, 2023

Revised Jul 20, 2023

Accepted Sep 6, 2023

Keywords:

Capacity utilization

Codecs

IP networks

Point-to-point calls

Voice over internet protocol

ABSTRACT

Voice over internet protocol (VoIP) calls are increasingly transported over computer-based networking due to several factors, such as low call rates. However, point-to-point (P-P) calls, as a division of VoIP, are encountering a capacity utilization issue. The main reason for that is the giant packet header, especially when compared to the runt P-P calls packet payload. Therefore, this research article introduced a method to solve the liability of the giant packet header of the P-P calls. The introduced method is named voice segment compaction (VSC). The VSC method employs the unneeded P-P calls packet header elements to carry the voice packet payload. This, in turn, reduces the size of the voice payload and improves network capacity utilization. The preliminary results demonstrated the importance of the introduced VSC method, while network capacity improved by up to 38.33%.

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



Corresponding Author:

Mosleh M. Abualhaj

Department of Networks and Cybersecurity, Faculty of Information Technology, Al-Ahliyya Amman University

Amman, Jordan

Email: m.abualhaj@ammanu.edu.jo

1. INTRODUCTION

Due to several factors, voice over internet protocol (VoIP) data is increasingly transported over computer-based networking, both wired and wireless networks. These factors include the advanced voice signal processing methods, the proliferation of mobile devices, the noticeable low call rate, and the wide spread of the Internet service [1]. Although there is no guarantee of call quality, numerous degradations in the wired and wireless network infrastructure can affect the voice signal. Additionally, VoIP calls waste significant network capacity [2], [3]. This work focuses on the network capacity of VoIP calls. The main reason for the poor network capacity utilization is the runt payload size of the VoIP packet produced by the VoIP codecs. A codec transforms the audio signal from analog to digital. Then, the digital audio is split into runt segments (the packet payload) to avoid the long packetization delay that might degrade the call quality. The codecs produce different sizes of the audio segment, e.g., the G.728 codec produces an audio segment with 10 bytes, linear predictive coding (LPC) with 14 bytes, G.723.1 with 20 bytes, and G.726 with 30 bytes. On the other hand, the size of protocols used to convey these runt audio segments is 40 bytes. These protocols are 12 bytes of real-time transfer protocol (RTP), 8 bytes of user diagram protocol (UDP), and 20 bytes of IP protocol (40 bytes of RTP/UDP/IP). Therefore, the wasted network capacity, based on the size of the codec audio segment and RTP/UDP/IP protocols, can reach up to 80% [4]–[6].

Generally, VoIP calls can be divided into point-to-point (P-P), point-to-multipoint, and multipoint-to-multipoint. P-P calls have only two participants, the caller and the callee. The same 40-byte RTP/UDP/IP protocols carry the voice segments for the three VoIP call types. In addition, these RTP/UDP/IP protocols are used to carry many applications data, including VoIP, video conferencing, webcasting, and TV distribution. Even UDP and IP are utilized to carry broader application data [7]–[10]. Accordingly, the RTP/UDP/IP protocols contain data (elements) in their headers to serve all these apps. As a result, the P-P calls, the focus of this work, do not require many headers, if not the majority. In this paper, we will investigate all the RTP/UDP/IP protocol header elements, locate the elements unneeded by the P-P calls, and utilize these elements to benefit the P-P calls. These elements will be used to keep all or at least a portion of the VoIP packet's voice segment (packet payload). Therefore, the P-P call packet size will be reduced due to shortening the payload. The network's capacity utilization will be improved.

This paper is organized as follows: section 2 discusses the related works. Section 3 discusses the proposed method. Section 4 discusses the results of the proposed method. Section 5 concludes the paper.

2. RELATED WORKS

Numerous studies have treated the network capacity utilization of P-P calls from different perspectives. On top of these perspectives are what P-P calls packet merging methods. As the name implies, these methods merge the voice segment of numerous P-P call packets from different sessions into a single header rather than one for each packet. Doing this, the header size consumes less space, improving network capacity utilization. Teymoori *et al.* [11] introduced one of the packet merging methods. In their paper, the method merges numerous VoIP packets over high-speed 802.11 wireless networks, focusing on finding the best-merged packet size. An analytical model was introduced to find the best packet size while accounting for the delay imposed by the merging process. NS2 simulation findings showed that the introduced analytical model is accurate to a great extent in the tested scenarios. Camps-Mur *et al.* [12] introduced yet another packet merging method. The merging method in their paper concentrates on improving the quality of data transmission over Wi-Fi networks and the efficiency of power-saving protocols. The simulation showed that the merging method performed better with 802.11 power save mode than with the 802.11e U-APSD protocol. Regardless of the merging method used, they all improved network capacity utilization. However, VoIP packet merging methods cause numerous dilemmas. In general, i) VoIP call quality is deteriorating, ii) packet merging methods need a good number (a high number) of simultaneous VoIP sessions to work, and iii) the merging process might make network devices work harder because of the processing that comes from merging [1], [13], [14].

Another vital perspective to improve network capacity utilization for P-P calls is packet header compaction. Packet header compaction accomplishes impressive reduction of the 40 bytes that P-P calls packet headers to only 2 or 4 bytes. This is accomplished based on two characteristics of the P-P calls packet header. The first characteristic is that numerous elements of the packet header contain the same value in all the succeeding P-P calls' packets. e.g., the source IP address element in the IP protocol header and the source port element in the UDP protocol header. The second characteristic is that numerous elements of the packet header are increasing with foreseeable values in all the succeeding P-P calls' packets. e.g., the RTP protocol header's time-stamp and sequence number elements. The elements that fulfill one of these characteristics can be sent in the first packet of the P-P call and dropped from all the remaining call packets. The callee side can then put these element values in all the incoming packets before sending them to the callee application [15]–[17]. Niu *et al.* [16] introduced one of the packet compacting methods. They have introduced a compacting method for the OpenFlow network for packets transmitted through satellite links. Unlike the other compacting methods, the introduced method compacts the packet header along with the layer 2 header. The results showed that capacity utilization has improved by more than 86% and 89% with IPv4 and IPv6, respectively. Garg *et al.* [17] introduced another packet compacting method. The newly developed method relies on the correlation of packet headers transferred from a source node to a destination node. The suggested mechanism, modified and improved IPv6 header compression (MIHC), performs better than implementation and evaluation of the enhanced header compression (IPHC) and NO_COMP, with 20% and 76% higher throughput, 13% and 38% less latency, 12% and 37% less round-trip time, and 13% and 39% smaller packet sizes, respectively. Regardless of the merging method used, they all improved network capacity utilization. However, the packet header compacting methods cause numerous dilemmas. Generally, i) the compacting methods encompass a variety of operations and processes that affect and decrease the efficiency of the network devices, and ii) the compacting methods are undesirable in situations where multiple packets are lost, as the header values of the subsequent packets will be incorrect [1], [14], [18].

Accordingly, the current perspectives, packet merging and header compaction, must be improved to treat P-P calls' network capacity utilization liability. Therefore, this research article introduces a method to solve the liability of RTP/UDP/IP header elements that are unnecessary for P-P calls. Generally, these superfluous elements will be utilized to carry all or at least a portion of the speech segment of the VoIP packet.

Therefore, the p-p call packet size will be reduced due to shortening the payload, and the network capacity utilization will be improved. The introduced method is named voice segment compaction (VSC).

3. METHOD

This section discusses the newly introduced VSC method. The VSC technique will be Applied on the VoIP device attached to the WAN on both the dispatcher and recipient sides. At the dispatcher-side VoIP appliance, the VSC method makes the payload compaction of the voice segment of the VoIP packet. The unit of the VSC method at the dispatcher is called D-VSC. The VSC technique constructs the compacted voice segment on the recipient side of the VoIP device. The unit of the VSC method at the recipient is called R-VSC. Sections 3.3 and 3.5 explain the D-VSC and R-VSC units, respectively. Figure 1 clarifies the place of the VSC method units.

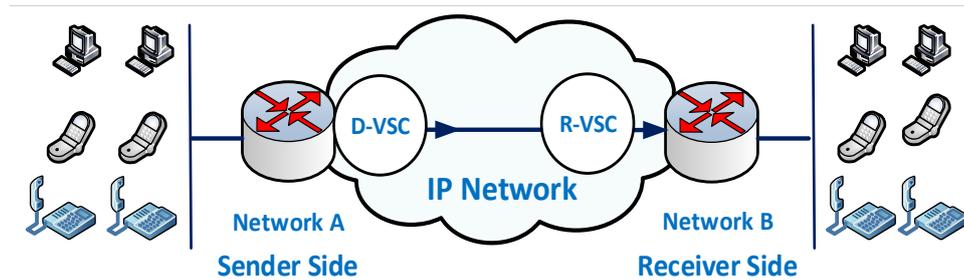


Figure 1. Place of the VSC method units

The VSC method employs the RTP/UDP/IP header elements to store and transmit the speech segment of the VoIP packet. These elements are called buffering elements. Several elements of the RTP/UDP/IP header will be employed as buffering elements on the basis of particular conditions (Cond.).

Cond. 1: The element has to be unneeded in P-P calls to convey the voice segment to the callee.

Cond. 2: Storing new data in place of the original data in the element does not result in misunderstanding the P-P call packet by the callee clients.

Cond. 3: The value of particular elements is constant and does not alter at any time during any call. The R-VSC unit can set the value of such an element on the recipient side.

Cond. 4: The value of particular elements is determined at the outset of the P-P call and does not change throughout the whole call packet. The R-VSC unit can save the value of such elements at the recipient's side appliance. After that, the R-VSC unit can set these elements' values before sending the packet to the addressee.

Cond. 5: If the R-VSC unit at the recipient side appliance creates a state table (ST), inferring the value of particular elements from other elements is possible.

Cond. 6: As detailed, the header compaction methods load the layer 3 equipment between the P-P call ends by building ST [19], [20]. Thus, no context or state table should be built on this equipment to reinstate any value of RTP/UDP/IP header elements.

Based on these conditions, buffering elements are: i) total length (TLen), identification (ID), source IP address (SIPA), protocol (Prt) in the IP protocol header; ii) checksum (Chk), length (Len), and source port (SrcP) in the UDP protocol header; and iii) timestamp (Ti) and synchronization source (SSRC) in the RTP protocol header. The complete size of the buffering elements is 184 bits (23 bytes). The next section explains the buffering elements in detail.

3.1. Buffering elements

The IP protocol header encompasses a number of unnecessary elements to move the P-P call conversation and are considered buffering elements. The first element is the TLen element. The TLen element keeps the size of the entire IP packet, equivalent to the length of the unchanged 40-byte RTP/UDP/IP header and the size of the codec voice segment. In a VoIP call, the utilized codec is determined at the outset of the P-P call and does not change throughout the whole call packet [14], [21], [22]. As a result, the D-VSC unit can maintain the size of the codec segment in the ST table. After that, the TLen element can be calculated from the segment size in the table plus the 40-byte header size. Hence, the TLen element is deemed a buffering element.

The next unnecessary element to move the P-P call conversation into the IP header is the ID element. The ID element helps in reassembling the segmented packet on the addressee side. The P-P call packet sizes are usually extremely small (less than 80 bytes). Hence, the P-P call packets are not segmented because their size is smaller than the existing layer 2 technologies' maximum transmission unit (MTU) [14], [21], [23]. Therefore, the ID element is unnecessary for P-P calls and can be deemed a buffering element. The Prt is another unnecessary element in the IP header to move the P-P call conversation. The Prt element specifies the transport layer protocol in the packet, which is always the UDP protocol with a value of 17 in the case of P-P calls [14], [21]. As a result, the R-VSC unit can assign the Prt element the number 17 before dispatching it to the call addressee. The SIPA element is the last element in the IP header that is unnecessary to move the P-P call conversation. The SIPA element is used to differentiate the data source for the addressee. However, the P-P calls establish two unidirectional sessions (one for each end of the call) [14], [21], [22]. Hence, the SIPA element is not needed to differentiate the data source by the addressee in the case of a P-P call. Therefore, the SIPA element can carry a portion of the voice segment.

In addition to the IP header, the UDP header encompasses a number of unnecessary elements to move the P-P call conversation and are considered buffering elements. The first element is the SrcP element. The SrcP is not mandatory for all applications, such as P-P calls. This is because there are two unidirectional sessions and, thus, no need to distinguish the source [14], [23], [24]. Hence, the SrcP element can carry a portion of the voice segment. The next unnecessary element to move the P-P call conversation in the UDP header is the Len element [14], [23], [24]. The Len element maintains the size of the codec voice segment. Therefore, its value can be retrieved from the ST table at the D-VSC unit. The Chk element is the last element in the UDP header that is unnecessary to move the P-P call conversation. Chk is not a compulsory element; it often has zeros for deactivation. A lot of modern techniques are utilized to reconstruct the corrupted voice segment of the P-P call. This means that deactivating the non-compulsory Chk element allows the applications to re-form the wrecked voice segment of the P-P call, which improves the precision of P-P calls [19], [24], [25]. As a result, the P-P calls benefit from deactivating the Chk. Hence, the Chk element can be deemed a buffering element.

Lastly, two elements in the RTP header are unnecessary to move the P-P call conversation and are considered buffering elements, namely the Ti and SSRC elements. The Ti and sequence number (SN) elements are increased by a constant value in consecutive packets [26]–[28]. Thus, the R-VSC unit may drive the Ti from the SN based on their existing numerical relationship, as shown in the below section. The SSRC element is essential to recognize the source in multicast sessions and when the translator or mixer is used in P-P sessions. But it is unnecessary with the P-P calls [14], [27], [28].

3.2. Ti element and ST table

As previously stated, successive packets increase the Ti and SN elements by a constant value. Assume that the Ti and SN elements for five consecutive packets are (3, 10), (6, 20), (9, 30), (12, 40), and (15, 50), respectively. Clearly, depending on the numerical relation between them, the values of the Ti element can be determined from the corresponding values of the SN element. Any P-P call must save the Ti and SN element values of just the first packet in the ST table. In addition, the recipient client socket is saved in the ST table to differentiate between Ti and SN values for each call. Then, the delta (De) between the two elements is obtained and stored in the ST table. A sample of the ST table is shown in Table 1. The R-VSC unit will use De to obtain the original value of the Ti element. By adding De to the SN, one can obtain the Ti element. Moreover, as presented in Table 1, the ST table keeps track of the size of the speech segment throughout the call arrangement. The R-VSC unit then uses this value to determine the values of the TLen and Len elements, as stated above.

Table 1. ST table

Recipient socket	De	Codec segment size
13.13.13.1:10101	5035274231	20
13.13.13.2:10120	6035367218	10
13.13.13.3:11110	5135253252	10
13.13.13.4:10210	5035562325	30
13.13.13.5:11121	6035664214	30

3.3. VSC method: D-VSC unit

The R-VSC unit executes many operations on the dispatcher's WAN appliance. At first, the payload of a P-P call packet is deduced from the header. Next, this payload segment is saved in the buffering elements based on the size of each element. At first, portions of the segment are saved in the TLen element, then in the ID element, the Prt element, the SIPA element, the SrcP element, the Len element, the Chk element, the SSRC

element, and the Ti element. As stated, the overall size is 184 bits (23 bytes). The residual of the segments (if any) is positioned as an ordinary payload. If the payload is smaller than that of the buffering elements (23 bytes), then the residual portion of the buffering elements is assigned zeros, and no payload is positioned. This operation creates a new minor packet called M-Pkt. Next, as discussed below, the value of the internet header length (HL) element in the IP is altered. Lastly, the new M-Pkt is dispatched to the VoIP appliance of the recipient.

3.4. HL element and middle layer 3 appliances

Usually, a vast quantity of packets of diverse kinds and sources travel through layer 3 appliances (e.g., routers). Using the components in the common IP protocol header, the routers are unable to identify the type of packets. The intermediary appliances (between the call ends) for the newly developed VSC technology should be able to distinguish between the P-P call packets produced by the D-VSC unit and other types of packets. This prevents misinterpretation of the modified header element values of the D-VSC unit. Specifically, the routers should not process the values of the IP protocol header's TLen, ID, Prt, and SIPA elements for any and all packets produced by the D-VSC unit.

The VSC technique employs the HL element to classify the P-P call packet created by the D-VSC unit from all the other packet kinds. The purpose of the HL element is to maintain the length of the IP protocol header. The value of this element is always five when used with P-P calls [9], [21]. Therefore, any value less than five may denote a packet with updated header information; however, the D-VSC unit uses the value three. Therefore, it is possible to use any value lower than five to denote a packet with altered header values; however, the D-VSC unit uses the three values. Thus, if the router has a packet with an HL element of three, the router considers that the IP header length is 20 bytes, and the router does not process TLen, ID, Prt, or SIPA values. Like what happened with the proposed and deployed CRTP and robust header compression (ROHC) header standards [29], [30], routers' internal operations should be updated to understand the new P-P calls packet values created by the D-VSC unit.

3.5. VSC method: R-VSC unit

The R-VSC unit executes many operations at the recipient WAN appliance to rebuild the initial P-P call packets. At first, the HL element is inspected to classify the P-P call packet created by the D-VSC unit from all the other packet kinds. After that, the payload segment (if any) of the packets produced by the D-VSC unit is disconnected from the packet header. Then, the voice segment is obtained from the buffering elements of the header. The packet payload and speech segment from the preceding phases will be combined to create the initial P-P call packet. The TLen element's content comes first in the combination operation, then the ID element's content, the Prt element's content, the SIPA element's content, the SrcP element's content, the Len element's content, the Chk element's content, the SSRC element's content, and the Ti element's content. The packet payload (if any) is positioned last. After that, the TLen, Prt, Len, and Ti elements are changed to their initial values. Then, all the leftover elements utilized by the D-VSC unit at the dispatcher appliance are given zeros to prevent misprocessing at the recipient P-P call client. Next, the resulting initial payload is added to the header, which forms the initial packet. Finally, the packet is dispatched to its recipient client.

4. RESULTS AND DISCUSSION

The VSC method was compared with the common technique (the standard 40-byte RTP/UDP/IP header) of transmitting the P-P call packets. The common technique is abbreviated as Crtp. The call capacity (CC) and conserved bandwidth (CBW) were tested for VSC and Crtp methods. Figures 2, 3, and 4 present that the VSC technique's CC matched that of the Crtp technique with LPC, G.726, and G.723.1, respectively. As we can see, when utilizing the VSC technique, the CC is superior (more calls) than that of the Crtp method with the three codecs. Also, the CBW ratio was calculated based on the CC. As presented in Figure 5, the VSC method has generous CBW conservation compared with the Crtp technique with the three codecs. The CBW conserving with LPC, G.726, and G.723.1 is 25.93%, 32.86%, and 38.33%, respectively. These results of CC and CBW are due to buffering the whole or portion of the P-P call packet payload in the buffering elements of the RTP/UDP/IP header. Furthermore, the difference between the VSC and the Crtp methods of both CC and CBW varies across codecs. This is due to the fact that the ratio of payload stored in buffering elements differs between codecs.

According to the above results, the VSC method has improved network capacity utilization and conserved bandwidth. Therefore, more P-P calls can be run in the same network bandwidth as the traditional method. The findings presented highlight the potential of the VSC method for enhancing capacity utilization in point-to-point VoIP calls. The reported increase in network capacity is undoubtedly a positive outcome, which can have practical implications for improving the efficiency and performance of VoIP systems.

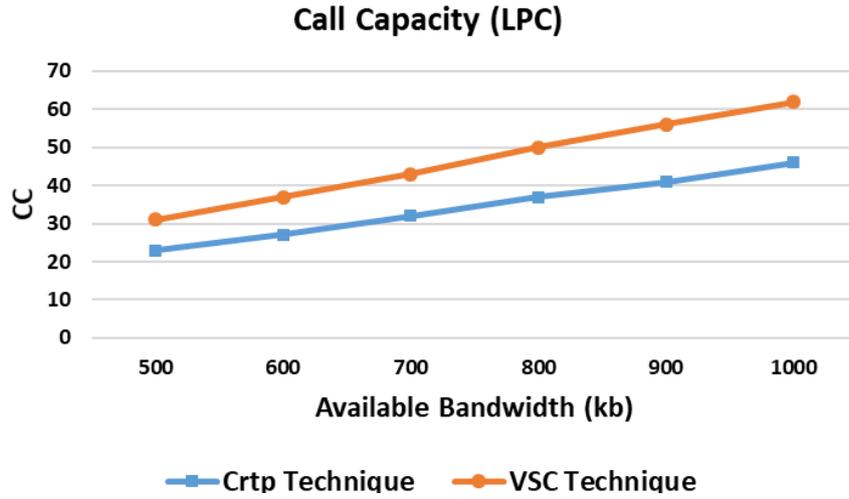


Figure 2. Call capacity using LPC

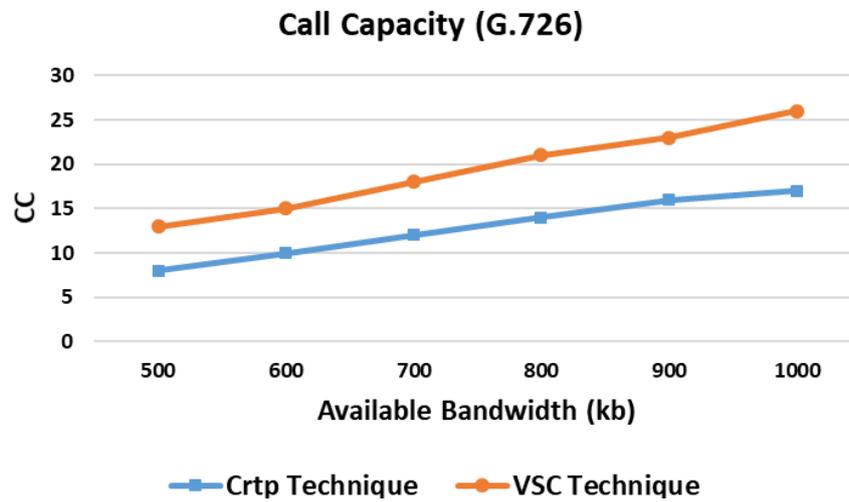


Figure 3. Call capacity using G.726

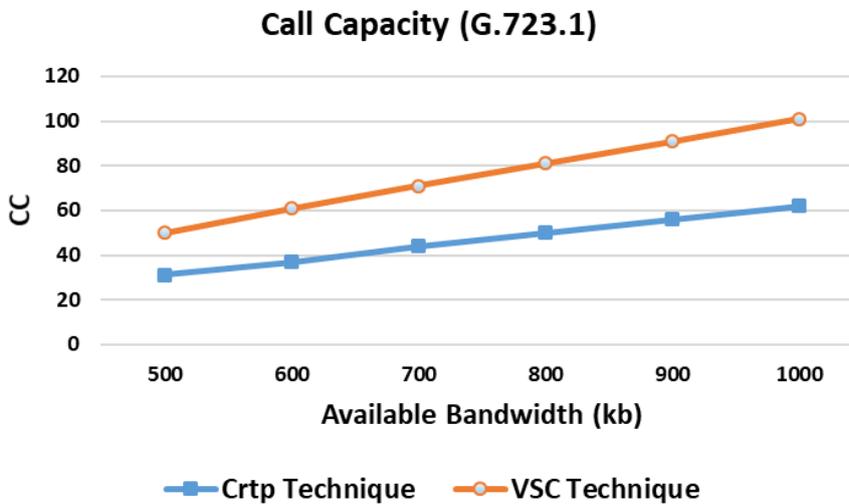


Figure 4. Call capacity using G.723.1

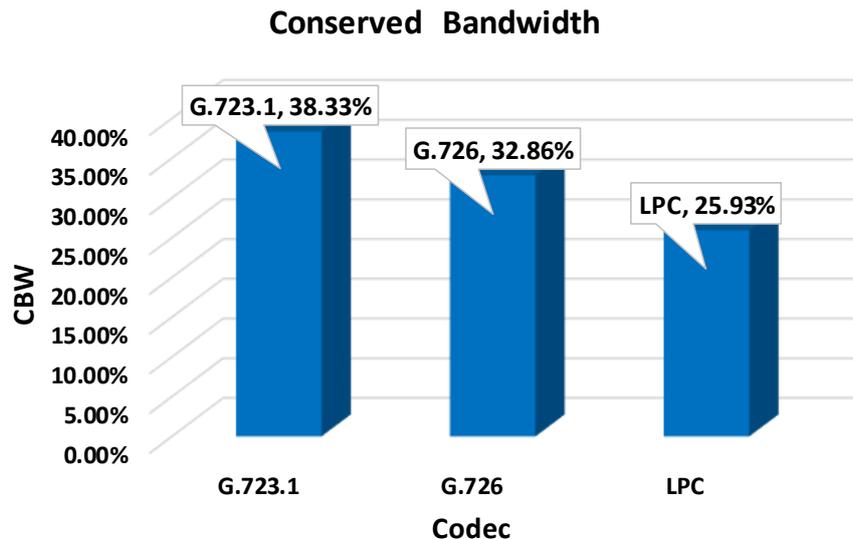


Figure 5. Conserved bandwidth

5. CONCLUSION

VoIP P-P calls are extensively used and have largely replaced traditional telephone calls. There are, however, some liabilities with VoIP P-P calls that prevent its pervasive use. One of the main liabilities is the giant packet header, especially compared to the runt packet payload. Considering this liability, the VSC method was introduced in this research article. The VSC method utilized the header elements in the RTP, UDP, and IP protocols, which are unneeded for VoIP P-P calls, to carry the voice segment of the P-P call packets. As a result, the voice segment is diminished or occasionally eliminated. Consequently, network capacity utilization will increase. The preliminary results clarified the importance of the introduced VSC method. The network capacity has increased by up to 25.93%, 32.86%, and 38.33% with the LPC, G.729, and G.723.1 codecs, respectively. In the future, the proposed VSC method will be integrated with other techniques that handle the bandwidth utilization problem, such as the packet aggregation technique. Also, the VSC method will be evaluated with different network configurations and parameters to validate the results.

REFERENCES

- [1] N. N. Alfaisaly, S. Q. Naeem, and A. H. Neama, "Enhancement of WiMAX networks using OPNET modeler platform," *Indonesian Journal of Electrical Engineering and Computer Science (IJECS)*, vol. 23, no. 3, pp. 1510–1519, Sep. 2021, doi: 10.11591/ijeecs.v23.i3.pp1510-1519.
- [2] K. Hassine and M. Frikha, "A VoIP focused frame aggregation in wireless local area networks: features and performance characteristics," in *2017 13th International Wireless Communications and Mobile Computing Conference (IWCMC)*, Jun. 2017, pp. 1375–1382, doi: 10.1109/IWCMC.2017.7986485.
- [3] R. Huchappa and K. K. Patil, "Evolutionary model to guarantee quality of service for tactical worldwide interoperability for microwave access networks," *IAES International Journal of Artificial Intelligence (IJ-AI)*, vol. 11, no. 2, pp. 687–698, Jun. 2022, doi: 10.11591/ijai.v11.i2.pp687-698.
- [4] N. Gupta, N. Kumar, and H. Kumar, "Comparative analysis of voice codecs over different environment scenarios in VoIP," in *2018 Second International Conference on Intelligent Computing and Control Systems (ICICCS)*, Jun. 2018, pp. 540–544, doi: 10.1109/ICCONS.2018.8663241.
- [5] P. Kadam, M. Kulkarni, and V. Gaikwad, "Bandwidth management for VoIP calling through asterisk," in *2021 2nd Global Conference for Advancement in Technology (GCAT)*, Oct. 2021, pp. 1–6, doi: 10.1109/GCAT52182.2021.9587544.
- [6] A. H. Hussein, M. Al-Zyoud, K. Nairoukh, and S. N., "Enhancing VoIP BW utilization over ITTP protocol," *International Journal of Advanced Computer Science and Applications*, vol. 11, no. 10, pp. 505–510, 2020, doi: 10.14569/IJACSA.2020.0111063.
- [7] R. A. Lara-Cueva and M. S. Simba-Herrera, "Performance evaluation for outdoor wireless scenarios based on IEEE 802.11b/g in the presence of co-channel interference," *Bulletin of Electrical Engineering and Informatics (BEEI)*, vol. 9, no. 1, pp. 187–197, Feb. 2020, doi: 10.11591/eei.v9i1.1649.
- [8] J. Gao, Y. Li, H. Jiang, L. Liu, and X. Zhang, "An RTP extension for reliable user-data transmission over VoIP traffic," in *Communications in Computer and Information Science*, vol. 1095, Springer Singapore, 2019, pp. 74–86.
- [9] G. Tomsho, *Guide to networking essentials*. Cengage Learning, 2019.
- [10] Q. Shambour, S. N. Alkhatib, M. M. Abualhaj, and Y. Alraba'nah, "Effective voice frame shrinking method to enhance VoIP bandwidth exploitation," *International Journal of Advanced Computer Science and Applications*, vol. 11, no. 7, pp. 313–319, 2020, doi: 10.14569/IJACSA.2020.0110741.
- [11] P. Teymoori, A. Dadlani, K. Sohraby, and K. Kim, "An optimal packet aggregation scheme in delay-constrained IEEE 802.11n WLANs," in *2012 8th International Conference on Wireless Communications, Networking and Mobile Computing*, Sep. 2012, pp. 1–4, doi: 10.1109/WiCOM.2012.6478424.

- [12] D. Camps-Mur, M. D. Gomony, X. Pérez-Costa, and S. Sallent-Ribes, "Leveraging 802.11n frame aggregation to enhance QoS and power consumption in Wi-Fi networks," *Computer Networks*, vol. 56, no. 12, pp. 2896–2911, Aug. 2012, doi: 10.1016/j.comnet.2012.05.004.
- [13] E. Charfi, C. Gueguen, L. Chaari, B. Cousin, and L. Kamoun, "Dynamic frame aggregation scheduler for multimedia applications in IEEE 802.11n networks," *Transactions on Emerging Telecommunications Technologies*, vol. 28, no. 2, Apr. 2017, doi: 10.1002/ett.2942.
- [14] M. M. Abualhaj, A. A. Abu-Shareha, and S. N. Al-Khatib, "Utilizing VoIP packet header's fields to save the bandwidth," *Transport and Telecommunication Journal*, vol. 24, no. 1, pp. 33–42, Feb. 2023, doi: 10.2478/ttj-2023-0004.
- [15] W. A. Flanagan, "Header compression across entire network without internet protocol saves bandwidth and latency," in *2020 IEEE 3rd 5G World Forum (5GWF)*, Sep. 2020, pp. 281–285, doi: 10.1109/5GWF49715.2020.9221185.
- [16] Y. Niu, C. Wu, L. Wei, B. Liu, and J. Cai, "Backfill: an efficient header compression scheme for OpenFlow network with satellite links," in *2016 International Conference on Networking and Network Applications (NaNA)*, Jul. 2016, pp. 202–205, doi: 10.1109/NaNA.2016.36.
- [17] R. Garg and S. Sharma, "Modified and improved IPv6 header compression (MIHC) scheme for 6LoWPAN," *Wireless Personal Communications*, vol. 103, no. 3, pp. 2019–2033, Dec. 2018, doi: 10.1007/s11277-018-5894-z.
- [18] M. M. Abualhaj, A. H. Hussein, Q. M. Kharm, Q. Y. Shambour, and S. N. Al-Khatib, "Saving the bandwidth of IPv6 networks using the fields of the packet header," *Intelligent Automation & Soft Computing*, vol. 31, no. 3, pp. 1971–1980, 2022, doi: 10.32604/iasc.2022.021458.
- [19] M. M. Abualhaj, M. M. Al-Tahrawi, and M. Al-Zyoud, "Contracting VoIP packet payload down to zero," *Cybernetics and Information Technologies*, vol. 21, no. 1, pp. 137–150, Mar. 2021, doi: 10.2478/cait-2021-0010.
- [20] M. M. Abualhaj, Q. Y. Shambour, A. H. Hussein, and Q. M. Kharm, "Down to zero size of VoIP packet payload," *Computers, Materials and Continua*, vol. 68, no. 1, pp. 1271–1283, 2021, doi: 10.32604/cmc.2021.014928.
- [21] J. F. Kurose and K. W. Ross, *Computer networking: a top-down approach*. Pearson India Education Services, 2022.
- [22] S. Konshin, M. Z. Yakubova, T. N. Nishanbayev, and O. A. Manankova, "Research and development of an IP network model based on PBX asterisk on the opnet modeler simulation package," in *2020 International Conference on Information Science and Communications Technologies (ICISCT)*, Nov. 2020, pp. 1–5, doi: 10.1109/ICISCT50599.2020.9351405.
- [23] E. Sadot and E. Zilbershtein, "Packet fragmentation prevention." U.S. Patent. No. 9,088,494.
- [24] L. Eggert, G. Fairhurst, and G. Shepherd, "UDP usage guidelines," RFC Editor, Mar. 2017, doi: 10.17487/RFC8085.
- [25] S. A. Khan, M. Moosa, F. Naeem, M. H. Alizai, and J.-M. Kim, "Protocols and mechanisms to recover failed packets in wireless networks: history and evolution," *IEEE Access*, vol. 4, pp. 4207–4224, 2016, doi: 10.1109/ACCESS.2016.2593605.
- [26] M. M. Abualhaj, Q. Y. Shambour, and A. R. H. Hussein, "Effective packet multiplexing method to improve bandwidth utilisation," *International Journal of Computer Applications in Technology*, vol. 63, no. 4, pp. 327–336, 2020, doi: 10.1504/IJCAT.2020.110410.
- [27] A. R. H. Hussein, M. Abu-Alhaija, and K. Nairoukh, "New RTP packet payload shrinking method to enhance bandwidth exploitation over RTP protocol," *International Journal of Advanced Computer Science and Applications*, vol. 11, no. 8, pp. 139–143, 2020, doi: 10.14569/IJACSA.2020.0110818.
- [28] D. Strzeciwillk, "Performance analysis of VoIP data over IP networks," *International Journal of Electronics and Telecommunications*, vol. 67, no. 4, pp. 743–750, Jul. 2021, doi: 10.24425/ijet.2021.139801.
- [29] W. Wu and Z. Ding, "A Markovian design of bi-directional robust header compression for efficient packet delivery in wireless networks," *IEEE Transactions on Wireless Communications*, vol. 18, no. 1, pp. 20–33, Jan. 2019, doi: 10.1109/TWC.2018.2875448.
- [30] C. Feres, W. Wu, and Z. Ding, "A Markovian ROHC control mechanism based on transport block link model in LTE networks," in *2018 IEEE International Conference on Communications (ICC)*, May 2018, pp. 1–7, doi: 10.1109/ICC.2018.8422165.

BIOGRAPHIES OF AUTHORS



Mosleh M. Abualhaj    is a senior lecturer in Al-Ahliyya Amman University. He received his first degree in computer science from Philadelphia University, Jordan, in 2004, master degree in computer information system from the Arab Academy for Banking and Financial Sciences, Jordan in 2007, and Ph.D. in multimedia networks protocols from Universiti Sains Malaysia in 2011. His research area of interest includes VoIP, multimedia networking, and congestion control. He can be contacted at email: m.abualhaj@ammanu.edu.jo.



Ahmad Adel Abu-Shareha    received his first degree in computer science from Al Al-Bayt University, Jordan, 2004, Master degree from Universiti Sains Malaysia (USM), Malaysia, 2006, and Ph.D. degree from USM, Malaysia, 2012. His research focuses on data mining, artificial intelligent and multimedia security. He investigated many machine learning algorithms and employed artificial intelligent in variety of fields, such as network, medical information process, knowledge construction and extraction. He can be contacted at email: a.abushareha@ammanu.edu.jo.



Sumaya Nabil Al-Khatib    is a senior lecturer in Al-Ahliyya Amman University. She received his first degree in computer science from Baghdad University, Iraq, in June 1994 and master degree in computer information system from the Arab Academy for Banking and Financial Sciences, Jordan in February. Her research area of interest includes VoIP, multimedia networking, and congestion control. He can be contacted at email: sumayakh@ammanu.edu.jo.



Mohammad O. Hiari    is a lecturer in Al-Ahliyya Amman University. He received his first degree in software engineering from Philadelphia University, Jordan, in August 2004 and master degree in computer science from Al Balqa Applied University, Jordan in February 2016. His research area of interest includes VoIP, multimedia networking, and congestion control. He can be contacted at email: m.hyari@ammanu.edu.jo.



Layth Al-Mahadeen    is a full-time lecturer at Al-Balqa Applied University, Jordan. He received his BSc in computer information systems from Mutah University, Jordan in 2002, and his MSc in information technology management from Sunderland University, UK in 2009. He holds a Ph.D. degree in management information systems from Girne American University, Cyprus, 2017 with specialization in business intelligence. His research interests are machine learning, data mining, NLP, and recommender systems. He can be contacted at email: layth.mahadeen@bau.edu.jo.