

# Enhancing Bandwidth Utilization of IP Telephony Over IPv6 Networks

Hani Al-Mimi<sup>1,\*</sup>, Yousef Alrabanah<sup>2</sup>, Mosleh M. Abualhaj<sup>3</sup> and Sumaya N. Al-khatib<sup>3</sup>

<sup>1</sup>Al-Zaytoonah University of Jordan, Faculty of Science and Information Technology, Department of Cybersecurity, Amman, 11733, Jordan

<sup>2</sup>Al-Ahliyya Amman University, Faculty of Information Technology, Department of Software Engineering, Amman, 19328, Jordan

<sup>3</sup>Al-Ahliyya Amman University, Faculty of Information Technology, Department of Networks and Information Security, Amman, 19328, Jordan

\*Corresponding Author: Hani Al-Mimi. Email: hani.mimi@zuj.edu.jo

Received: 13 October 2021; Accepted: 13 January 2022

**Abstract:** The demand for the telecommunication services, such as IP telephony, has increased dramatically during the COVID-19 pandemic lockdown. IP telephony should be enhanced to provide the expected quality. One of the issues that should be investigated in IP telephony is bandwidth utilization. IP telephony produces very small speech samples attached to a large packet header. The header of the IP telephony consumes a considerable share of the bandwidth allotted to the IP telephony. This wastes the network's bandwidth and influences the IP telephony quality. This paper proposes a mechanism (called Smallerize) that reduces the bandwidth consumed by both the speech sample and the header. This is achieved by assembling numerous IP telephony packets in one header and use the header's fields to carry the speech sample. Several metrics have been used to measure the achievement Smallerize mechanism. The number of calls has been increased by 245.1% compared to the typical mechanism. The bandwidth saving has also reached 68% with the G.28 codec. Therefore, Smallerize is a possible mechanism to enhance bandwidth utilization of the IP telephony.

**Keywords:** –IP telephony; codec; bandwidth utilization; IPv6

## 1 Introduction

IPv6 protocol is the future of IP-based networks, including the Internet. The main feature of IPv6 over IPv4 is the ample IP address space, whereas it provides 128-bit (16-byte) address space compared to 32-bit (4-byte) in IPv4. This large 16-byte IP address is the reason for the significant 40-byte IPv6 protocol compared 20-byte IPv4 protocol [1]. Meanwhile, many small-packet flows run over the networks, including online games and IP telephony (i.e., VoIP). The payload size of the small packets is mostly within 100-byte [2]. Therefore, attaching the IPv6 header to these small payloads is a severe problem that wastes the network's bandwidth.

The packet payload of IP telephony is composed of speech samples produced by a codec. The codec converts the analog waveform to digital speech samples in short periods to avoid unacceptable delay. The digital speech samples are typically between 10-byte to 30-byte, as shown in Tab. 1 [3,4]. In certain



This work is licensed under a Creative Commons Attribution 4.0 International License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

cases, the VoIP packet payload consists of more than one voice frame. The speech sample is first encapsulated in the 12-byte of Real-time Transport Protocol (RTP) protocol at the application layer. The resulted application layer protocol data unit (PDU) is then encapsulated by 8-byte UDP and 20/40-byte IPv4/IPv6 protocols in order. In IPv6, these three protocols are imposed a 60-byte header to each of the IP telephony packets [5,6]. Therefore, the ratio of the consumed bandwidth by the header is between 66.7% and 85.71%, based on the length of the digital speech sample. The bandwidth consumed by the large header can be lightened by assembling several PDUs of the transport layer (Speech sample + RTP + UDP) in one IPv6 header, known as packet multiplexing [5,7]. The resulted packet is called multiplexed packet (P-Mux). The ratio of the rescued bandwidth from the large IPv6 header is based on the number of assembled transport layer PDUs in the IPv6 header.

**Table 1:** Common speech codecs

Codec	G.729	G.728	G.723.1	G.726
Sample length (byte)	10	10	24	20
Mean opinion score	4.1	3.61	3.9	3.85
Bitrate (Kbps)	<b>8</b>	<b>16</b>	<b>5.3</b>	<b>24</b>

Many of the IP telephony calls are point-to-point (P2P), between only two ends. For such calls, much of the information in the RTP and UDP headers are not needed. The information in the RTP header is used to transfer all the real-time multimedia data such as video conferencing over the Internet and Internet audio/video streaming. On the other hand, the information in the UDP protocol is used to transfer all types of unreliable data over IP networks [8–12]. Therefore, this paper uses the fields in the RTP and UDP headers that are unneeded for P2P calls to carry the speech samples of the IP telephony packets besides packet multiplexing. Thus, more of the consumed bandwidth by the header can be rescued, particularly with P2P calls.

The organization of this paper is as follows: Section 2 presents a brief review of the current work of packet multiplexing mechanism. Section 3 gives a profound explanation of the proposed mechanism. Section 4 presents the performance analysis of the proposed mechanism. Section 5 is the conclusion.

## 2 Related Works

Several packet multiplexing mechanisms have been created to reduce the severity of the large header for small packets flows. One of the first IP telephony multiplexing mechanisms was created by Hoshi et al. [13]. Hoshi proposed to assemble several PDUs of the application layer (Speech sample + RTP) in one UDP/IP header. The packets are assembled in one P-Mux at the sender IP telephony gateways (IP-GW) and disassembled at the receiver IP-GW. Hoshi's mechanism relies on the Synchronization Source (SSRC) identifier field in the RTP header to identify the speech stream. The proposed mechanism is implemented and evaluated against the typical mechanism (no multiplexing) using G.723.1. The result showed that the bandwidth is enhanced by 40% and the number of the packets is reduced to by 87.5% in the tested cases.

Sze et al. proposed another multiplexing mechanism to promote bandwidth utilization of IP telephony networks [14]. Similar to Hoshi's mechanism, the multiplexing occurs at the transport layer of the OSI stack, in which multiple application layer PDUs are piggybacked into one UDP/IP header. Besides the multiplexing, the proposed mechanism adopted a simplified header compression mechanism [15]. The basic idea of this simplified version of header compression is to remove the UDP/IP header fields that are unnecessary to transport the IP telephony packet to the intended destination. The proposed mechanism is

designed to be compatible with H.323 signaling environment. In addition, the P-Mux packet size is limited by the size of the network maximum transmission unit (MTU), or a period equals a speech sample construction duration. Combining multiplexing and a simplified version of the header compression promoted the bandwidth utilization of IP telephony networks by 300%.

More recently, one of the top multiplexing mechanisms was created and patented by Roay and sponsored by Cisco establishment [16]. Several transport layer PDUs (RTP, UDP header, and speech sample) are assembled in P-Mux. The first PDU within the P-Mux packet contains information to tag the P-Mux packet. This tag is utilized by the receiver IP-GW to distinguish the P-Mux packet from the normal packet. The capability of the receiving IP-GW to handle the P-Mux packet is exchanged during the P2P call formation using SIP/H.323 protocols. If not, the transmitted packet ought to be a normal packet (Not P-Mux packets). To keep away from any extra latency, the proposed mechanism assembles the packets available in the buffer and does not wait for any packets to come. In addition, the P-Mux is transmitted as soon as the size reaches the MTU. Clearly, Roy's mechanism rescues the bandwidth by removing a 20-byte IPv4 header from each PDU within the P-Mux or 40-byte in the case of IPv6.

The mechanisms above are using the concept of packet multiplexing to save the bandwidth. Besides packet multiplexing, this paper uses the packet header's fields that are unnecessary for P2P calls to carry the speech sample. This smallerizes the packet payload and reduces the consumed bandwidth by IP telephony systems. The proposed mechanism, called Smallerize, is designed for P2P IP telephony calls over IPv6 networks.

### 3 Smallerize Mechanism

The Smallerize mechanism is created to lighten the bandwidth needed by IPv6 telephony, based on two main algorithms. First, the Smallerize mechanism multiplexes the packets to the same IP-GW in one P-Mux. The P-Mux packet contains several application layer PDUs in one UDP/IPv6 header. Henceforth, the term PDU refers to the application layer PDU, which contains the RTP header and a speech sample. The packet multiplexing process is performed at the sender IP-GW (IP-GWs). The P-Mux packet is dissembled at the receiver IP-GW (IP-GWr) to build the standard IP telephony packet (S-Pkt). Fig. 1 clarifies the packet multiplexing concept along with the P-Mux packet and S-Pkt packet. Second, part of the voice sample of the packet will be carried in the RTP header of each PDU within the P-Mux, as discussed below.

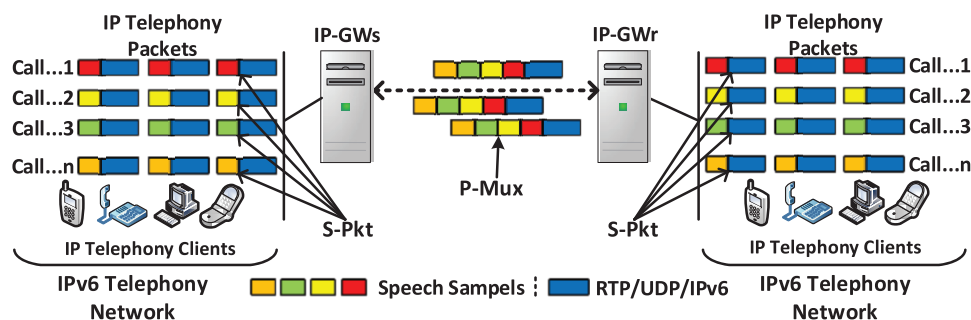


Figure 1: Packets multiplexing elements

#### 3.1 RTP Header of the PDU

The Smallerize mechanism assembles several PDUs in one P-Mux packet. As stated, these PDUs consist of RTP header and speech sample. Some of the fields in the RTP header are unnecessary to transfer the speech data of the P2P calls. The 4-byte SSRC field is usually utilized to fix the conflict if the sequence number is the

same for two sources, identify the call source in multicast IP telephony sessions, or use a translator or mixer with IP telephony. However, none of these cases are in the P2P calls of the IP telephony [10,12]. Therefore, the Smallerize mechanism can utilize the SSRC field to carry part of the speech sample data.

The 4-byte Timestamp field of the first packet produced is chosen randomly. Then, it is increased by the same value for each of the following packets. The 2-byte Sequence Number field is also chosen randomly and increased by the same value for each of the following packets. Therefore, the Timestamp can be derived from the Sequence Number based on the harmonic increment between them [10,12]. For illustration, assume the Timestamp of the first packet is 100 and the increment value is 50. Then, the Timestamp of the first five packets is 100, 150, 200, 250, and 300, respectively. In addition, assume the Sequence Number of the first packet is 50 and the increment value is 10. Then, the Sequence Number of the first five packets is 50, 60, 70, 80, and 90, respectively. Clearly, the Timestamp can be derived from the Sequence Number using basic math. The delta (D) between the Sequence Number and Timestamp is calculated for each call. The D value is stored in the Socket Table (Tab. 2), which is discussed in Subsection 3.4. The IP-GWr uses the D value to calculate the original value of the Timestamp by adding the D value to the Sequence Number in the PDU header. Therefore, the Smallerize mechanism can utilize the Timestamp field to carry part of the speech sample data. Accordingly, carrying an 8-byte speech sample in the SSRC and Timestamp shortens each of the PDU in the P-Mux and, thus, save the bandwidth.

**Table 2:** Socket Table

IP-GWs			IP-GWr			
New PT value	Destination socket	Actual PT value	New PT value	Destination socket	Actual PT value	D value
1	2001:4321::2b22:5050	7	1	2001:4321::2b22:5050	7	5045273232
2	2001:4321::2b27:5050	15	2	2001:4321::2b27: 5050	15	6055327216
5	2001:4321::2b29:5051	15	5	2001:4321::2b29: 5051	15	5145253455
7	2001:4321::2b26:5050	7	7	2001:4321::2b26: 5050	7	6145223451
13	2001:4321::2b25: 5051	15	13	2001:4321::2b25: 5051	15	6145243459

### 3.2 Smallerize Mechanism–Sender Side

The Smallerize mechanism implements several steps to shorten the speech sample and assemble the S-Pkt packets in one P-Mux. First, the PDU is extracted from the S-Pkt. Then, 8-byte of the speech sample of every PDU is placed in the SSRC and the Timestamp fields of the PDU header (RTP). This produces PDU with a smaller speech sample. After that, the produced PDUs destined to the same IP-GW are assembled in one UDP/IPv6 header. Fig. 2 shows the format of the P-Mux packet. The destination socket of each call is kept in a specific table (called Socket table) on IP-GWs and IP-GWr. The purpose of the Socket table (Tab. 2) is discussed in Subsection 3.4. Fig. 3 demonstrates the flowchart of the Smallerize mechanism at the sender side.

### 3.3 Smallerize Mechanism–Receiver Side

The Smallerize mechanism implements several steps to disassemble the P-Mux and retrieve the S-Pkt. First, take off the UDP/IPv6 header from the P-Mux packet. Then, disassemble the PDUs within the P-Mux payload. Next, pull the speech sample inside the SSRC and the Timestamp fields and place it at the end of the speech sample of the PDU. This produces the PDU with the original full-length speech sample. Both SSRC and the Source Port fields are disabled to avoid misinterpreting them by the callee client. Following, derive

the value of the Timestamp as explained in Subsections 3.1 and 3.4. Finally, add the UDP/IPv6 header to the PDU to build the standard S-Pkt packet. The destination socket of each S-Pkt packet is retrieved from the Socket table (Tab. 2), as explained in Subsection 3.4. The resulted original S-Pkt packet is sent to the final destination. Fig. 4 demonstrates the Smallerize mechanism at the receiver side.

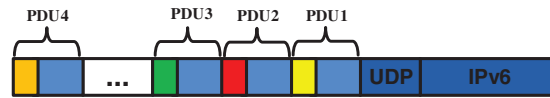


Figure 2: P-Mux packet format

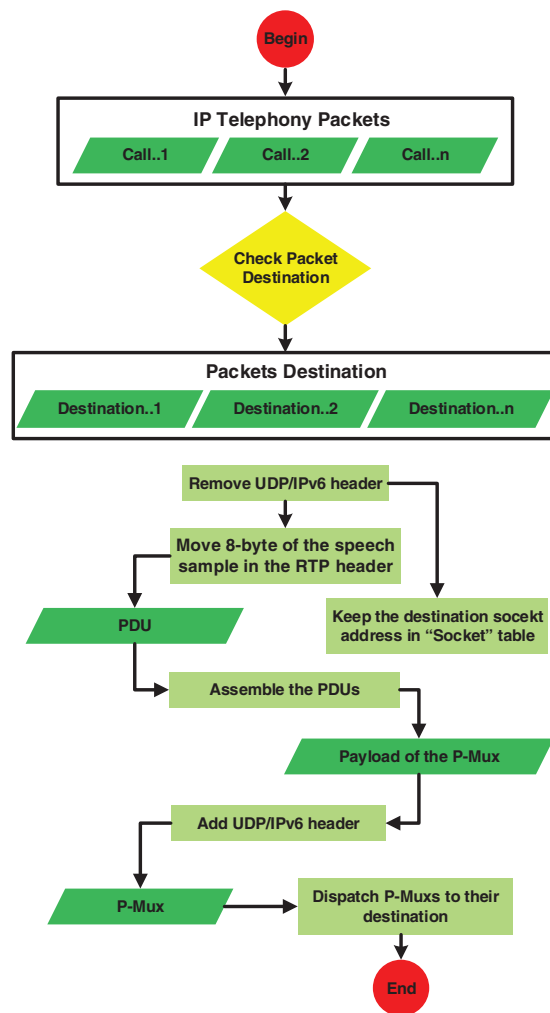
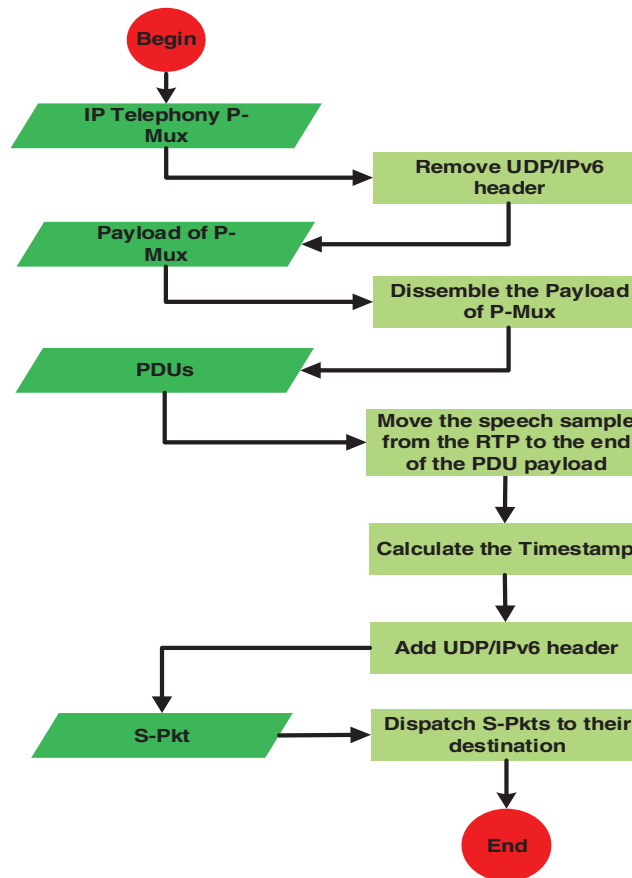


Figure 3: The smallerize mechanism-sender side

### 3.4 Socket Table

The Smallerize mechanism adds the UDP/IPv6 header to the PDU, including the destination socket address, to build the S-Pkt packet. The socket information is stored in the Socket table at the IP-GWs and IP-GWr. A value is needed to identify the destination socket address of the packets belonging to the same call. The Smallerize mechanism utilizes the Payload Type (PT) field in the RTP header for this purpose.

The 7-bit PT field contains the type of the used codec [10]. However, the codec type is exchanged by the signaling protocol at the call establishment phase. Therefore, the value of the PT can be saved in the Socket table, and no need to send it in the RTP header of each PDU. Therefore, the Smallerize mechanism uses PT to distinguish the destination socket of the packets inside the Socket table. Besides, as stated, the D value is stored in the Socket table to retrieve the Timestamp value. The D value is calculated from the first packet of each call and sent to the IP-GWr. Tab. 2 exhibits the Socket table at IP-GWs and IP-GWr gateways.



**Figure 4:** The smallerize mechanism-receiver side

#### 4 Smallerize Mechanism Analysis

The Smallerize mechanism was analyzed and compared to the traditional IPv6 mechanism (without multiplexing or payload shortening) and Roay mechanism. These mechanisms were compared based on three metrics: the number of calls (No. of calls), bandwidth saving, and speech sample reduction. The Smallerize mechanism was implemented at the IP-GWs and IP-GWr gateways. The processes occur at IP-GWs and IP-GWr gateways have been explained in Section 3. The scheme of simulated network is similar to the one in Fig. 1. In which, each gateway was simulated as a node that is connected to several IP Telephony clients. The channel between the two gateways is represented as a queue of 15 packets. The number of calls has been calculated just before the channel is saturated and the packet loss started. The G.728 and G.726 codecs were used in the experiments for more realistic analysis.

### 4.1 No. of Calls

Figs. 5 and 6 show the Smallerize mechanism's number of calls vs. the IPv6 mechanism and Roay mechanism, using G.26 and G.28 codecs, respectively. The Smallerize mechanism revealed more calls with the two codecs versus the IPv6 mechanism and Roay mechanism. Furthermore, the number of calls with G.28 codec has been increased over G.26 codec. For example, the number of calls with the G.28 codec at 1000 kb bandwidth is 111, 81, and 35 when running the Smallerize mechanism, Roay mechanism, and IPv6 mechanism, respectively. Therefore, the calls are increased by up to 317% and 231% over the conventional IPv6 mechanism when using the Smallerize mechanism and Roay mechanism, respectively. On the other hand, the number of calls with the G.26 codec at 1000 kb bandwidth is 76, 61, and 31 when running the mechanism, Roay mechanism, and IPv6 mechanism, respectively. Therefore, the number of calls is increased by up to 245.1% and 197% over the typical IPv6 mechanism when using the Smallerize mechanism and Roay mechanism, respectively. This result is attributed to assembling multiple S-Pkt packets in a single P-Mux packet with a single UDP/IPv6 header instead of a single UDP/IPv6 header to each S-Pkt. However, the G.28 codec outperforms G.26 codec because the speech sample of the G.26 codec is 20-bytes while the speech sample of the G.28 codec is 10-bytes. Therefore, placing an 8-byte of speech payload in the SSRC and Timestamp fields influences the 10-byte G.28 codec over the 20-byte G.26 codec. Thus, the shorter the speech sample of a codec, the more the number of calls is increased when using the Smallerize mechanism.

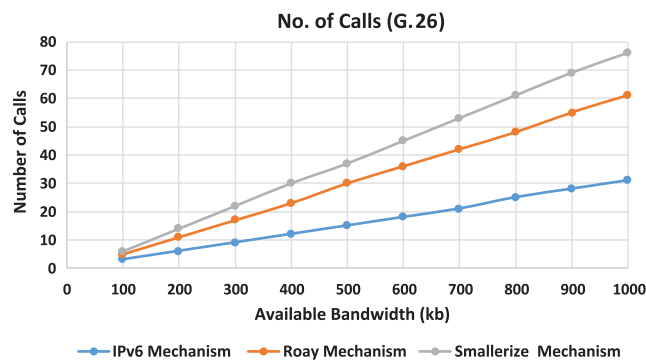


Figure 5: No. of calls (G.26 codec)

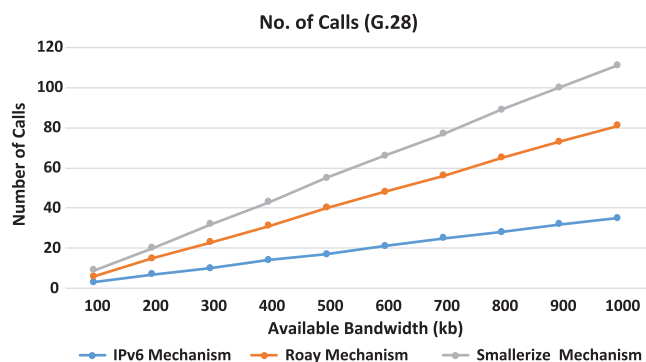
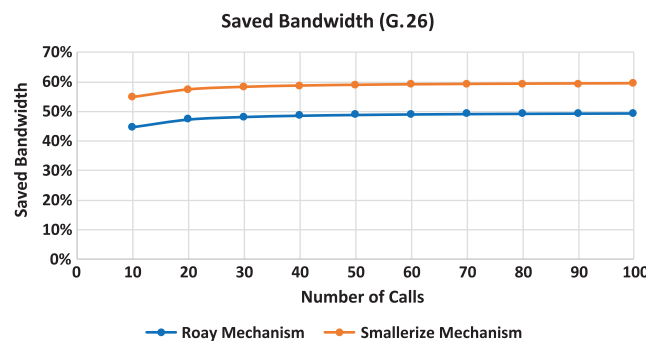


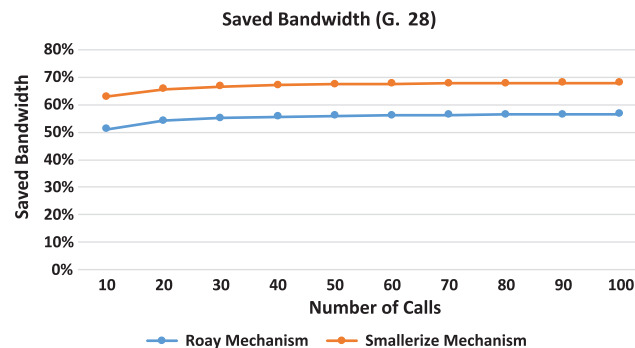
Figure 6: No. of calls (G.28 codec)

## 4.2 Bandwidth Saving

Figs. 7 and 8 present the bandwidth saving of the Smallerize mechanism and Roay Smallerize mechanism matched to the conventional IPv6 mechanism, using G.26 and G.28 codecs, respectively. Again, with the two codecs, the Smallerize mechanism presented more bandwidth saving than Roay mechanism. Furthermore, the bandwidth saving with G.28 codec has increased over G.26 codec. For example, the bandwidth saving with the G.28 codec at 100 calls is 68% and 57% when running the Smallerize mechanism and Roay mechanism matched to the IPv6 mechanism, respectively. Thus, the Smallerize mechanism presented more bandwidth saving by up to 10% than the Roay mechanism. On the other hand, the Smallerize mechanism gave bandwidth saving with the G.26 codec equals 60% at 100 calls which is less than the G.28 codec. This result is attributed to the exact reasons for increasing the capacity of the call.



**Figure 7:** Bandwidth saving (G.26 codec)

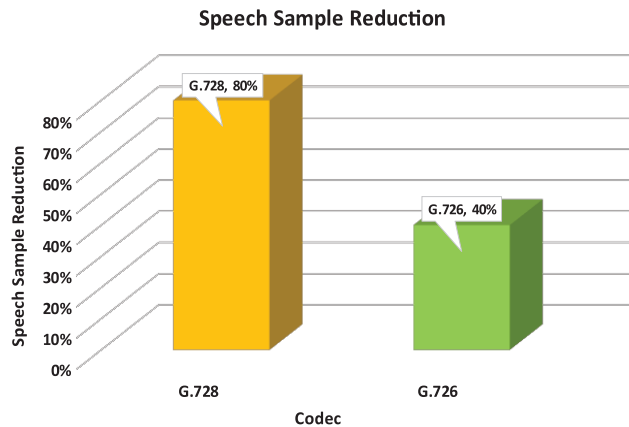


**Figure 8:** Bandwidth saving (G.28 codec)

## 4.3 Speech Sample Reduction

Fig. 9 displays the speech sample reduction ratio when utilizing the Smallerize mechanism. The reduction ratio is 40% and 80% when using the G.26 codec and G.28 codec, respectively. The resulting reduction is due to moving a portion of the speech sample in the 8-byte SSRC and Timestamp fields. Furthermore, the reduction ratio is higher when using the G.28 codec versus the G.26 codec. This is because the speech sample of the G.28 codec is shorter than the speech sample of the G.26 codec. Thus, the shorter the speech sample of a codec, the more the reduction ratio.





**Figure 9:** Speech sample reduction ratio

**4.4 Impact of Smallerize on Network Performance**

The No. of calls, bandwidth saving, and speech sample reduction metrics reflect the bandwidth utilization. The Smallerize mechanism outperforms the similar mechanisms with the three metrics. Thus, it accomplishes the main objective of increasing the bandwidth utilization when running IP telephony over IPv6 networks. Besides, the Smallerize mechanism minimized the number of IPv6 packets injected into the network due to assembling multiple S-Pkt packets in a single P-Mux packet. This will highly influence the IPv6 network performance elements, as shown in [Tab. 3 \[16–25\]](#).

**Table 3:** Smallerize network performance

Element	Effect
Header size	The large UDP/IPv6 header's wasted bandwidth has become negligible because multiple S-Pkt packets are assembled in one P-Mux packet with a single UDP/IPv6 header, rather than a dedicated UDP/IPv6 header to each S-Pkt.
Speech frame	The speech sample becomes shorter due to carry 8-bytes in the SSRC and timestamp fields.
Bandwidth saving	The wasted bandwidth is saved by: i) assembling multiple S-Pkt packets in one P-Mux packet with a single UDP/IPv6 header and ii) store part of the speech sample in the SSRC and Timestamp fields.
Number of calls	The IPv6 network number of calls has increased for the same reasons in the bandwidth saving.
Number of packets	The number of IPv6 packets injected into the network has been minimized due to assembling multiple S-Pkt packets in a single P-Mux packet.
Routers performance	The layer 3 routers are processing one P-Mux packet rather than many S-Pkt packets. This speeds up the routing process and saves the router resources.
Packet loss	The router buffer can accommodate more packets because the header size is reduced and the speech sample is reduced. Thus, this will impact (reduce) the ratio of packet loss. In addition, speeding up the routing process reduces the possibility of overwhelming the router buffer, reducing packet loss.
Latency	The latency has decreased due to speeding up the routing process. However, the processing overhead compensates for this latency. Therefore, the latency will almost neither have impacted positively nor negatively.

## 5 Conclusion

IP telephony systems waste a significant share of the bandwidth in IPv6 networks. This paper created the Smallerize mechanism to minimize this wasted bandwidth, particularly for P2P IP telephony calls. The Smallerize mechanism saved the bandwidth by i) assembling numerous IP telephony packets in one P-Mux with a single UDP/IPv6 header and ii) use the SSRC and Timestamp fields in each PDU header to carry the speech sample of the packet and, thus shorten the payload. The Smallerize mechanism performs the assembling and shortening operations at the IP-GWs at the sender side. It reverses the operations at the IP-GWr at the receiver side to restore the S-Pkt. The Smallerize mechanism was analyzed and compared to the traditional IPv6 mechanism and Roay mechanism. The number of calls has been increased by 245.1% compared to the typical IPv6 mechanism. In addition, the bandwidth saving has reached 68% with the G.28 codec. Finally, the speech sample reduction has reached 40% and 80% when utilizing the G.26 codec and G.28 codec, respectively. Thses metrics are reflecting the the bandwidth utilization. The Smallerize mechanism outperforms the similar mechanisms with the three metrics. Thus, it accomplishes the main objective of increasing the bandwidth utilization when running IP telephony over IPv6 networks. Besides, the Smallerize mechanism minimized the number of IPv6 packets injected into the network due to assembling multiple S-Pkt packets in a single P-Mux packet. This will highly influence the IPv6 network performance elements. Therefore, Smallerize mechanism achieved the goal of enhancing the performance of IP telephony systems. As future works, the impact of changing the original value of the SSRC and the Timestamp fields on the security issues and SDN-based networks will be discussed. In addition, a mathematical model for the proposed mechanism will be considered. Finally, a real implementation of the proposed method will be considered.

**Funding Statement:** The authors received no specific funding for this study.

**Conflicts of Interest:** The authors declare that they have no conflicts of interest to report regarding the present study.

## References

- [1] T. Lammler, “*CCNA Routing and Switching Complete Deluxe Study Guide: Exam 100–105, Exam 200–105, Exam 200–125*,” 2nd edition, Hoboken, NJ, USA: John Wiley & Sons, 2016.
- [2] S. Jose, H. David, N. Julian, R. José and P. Fernando *et al.*, “Small-packet flows in software defined networks: Traffic profile optimization,” *Networks*, vol. 10, no. 4, pp. 176–187, 2015.
- [3] A. Hussein, M. Al-Zyoued, K. Nairoukh and S. N. Al-Khatib, “Enhancing VoIP BW utilization over ITTP protocol,” *International Journal of Advanced Computer Science and Applications*, vol. 11, no. 10, pp. 505–510, 2020.
- [4] N. Gupta, N. Kumar and H. Kumar, “Comparative analysis of voice codecs over different environment scenarios in VoIP,” in *Proc. of the 2018 Second Int. Conf. on Intelligent Computing and Control Systems (ICICCS)*, Madurai, India, pp. 540–544, 2018.
- [5] M. M. Abualhaj, Q. Y. Shambour and A. H. Hussein, “Effective packet multiplexing method to improve bandwidth utilization,” *International Journal of Computer Applications in Technology*, vol. 63, no. 4, pp. 327–336, 2020.
- [6] S. Hagen, “*IPv6 Essentials: Integrating IPv6 into Your IPv4 Network*,” 3rd edition, Sebastopol, California, USA: O’Reilly Media, 2014.
- [7] K. Hassine and M. Frikha, “A VoIP focused frame aggregation in wireless local area networks: Features and performance characteristics,” in *Proc. of the 2017 13th Int. Wireless Communications and Mobile Computing Conf. (IWCMC)*, Valencia, Spain, pp. 1375–1382, 2017.
- [8] G. Tomsho, “*Guide to Networking Essentials*,” 8th edition, Boston, MA, USA: Cengage Learning, 2019.
- [9] M. Kolhar, “Zeroize: A new method to improve the utilization of 5G networks when running VoIP over IPv6,” *Applied System Innovation*, vol. 4, no. 4, no. 72, pp. 1–11, 2021.

- [10] H. W. Barz and G. A. Bassett, “*Multimedia Networks: Protocols, Design and Applications*,” 1st edition, Hoboken, NJ, USA: Wiley, 2016.
- [11] A. A. Alkhatib, A. A. Hnaif and T. G. Kanan, “Proposed simple system for Road Traffic Counting,” *International Journal of Sensors, Wireless Communications and Control*, vol. 8, no. 3, pp. 334–345, 2018.
- [12] J. Gao, Y. Li, H. Jiang, L. Liu and X. Zhang, “An RTP extension for reliable user-data transmission over VoIP traffic,” in *Proc. of the Int. Symp. on Security and Privacy in Social Networks and Big Data*, Copenhagen, Denmark, pp. 74–86, 2019.
- [13] T. Hoshi, K. Tanigawa and K. Tsukada, “Proposal of a method of voice stream multiplexing for IP telephony systems,” in *Proc. of the 1999 Internet Workshop. IWS99*. (Cat. No. 99EX385), Osaka, Japan, pp. 182–188, 1999.
- [14] 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, no. 7, pp. 1360–1368, 2002.
- [15] S. Casner and V. Jacobson, “Compressing IP/UDP/RTP headers for lowspeed serial links,” USA: RFC Editor, *IETF RFC 2508*, 1999.
- [16] B. Roay, “Generic UDP multiplexing for voice over internet protocol (VOIP),” U.S. Patent No. 8,553,692. 8, Oct. 2013.
- [17] C. Vulkan, A. Rakos, Z. Vincze and A. Drozdy, “Reducing overhead on voice traffic,” U.S. Patent No. 8,553,692, 8 Oct. 2013.
- [18] F. De Rango, P. Fazio, F. Scarcello and F. Conte, “A new distributed application and network layer protocol for VoIP in mobile ad hoc networks,” *IEEE Transactions on Mobile Computing*, vol. 13, no. 10, pp. 2185–2198, 2014.
- [19] Z. Shah, A. Suleman, I. Ullah and A. Baig, “Effect of transmission opportunity and frame aggregation on VoIP capacity over IEEE 802.11n WLANs,” in *Proc. of 8th Int. Conf. on Signal Processing and Communication Systems (ICSPCS)*, Gold Coast, QLD, Australia, pp. 1–7, 2014.
- [20] S. Seytnazarov and Y. Kim, “QoS-aware adaptive A-MPDU aggregation scheduler for enhanced VoIP capacity over aggregation-enabled WLANs,” in *Proc. of NOMS 2018–2018 IEEE/IFIP Network Operations and Management Symposium*, Taipei, Taiwan, pp. 1–7, 2018.
- [21] W. Cui, X. Meng, H. Yang, Q. Kang and Z. Zhao, “QoS-aware approach for maximizing rerouting traffic in IP networks,” *KSI Transactions on Internet and Information Systems*, vol. 10, no. 9, pp. 4287–4306, 2016.
- [22] M. M. Abualhaj, S. N. Al-Khatib and Q. Y. Shambour, “PS-PC: An effective method to improve VoIP technology bandwidth utilization over ITTP protocol,” *Cybernetics and Information Technologies*, vol. 20, no. 3, pp. 147–158, 2020.
- [23] I. Nedyalkov, “An original and simple method for studying the performance of a VoIP network,” in *Proc. of National Conf. with Int. Participation (ELECTRONICA)*, Sofia, Bulgaria, pp. 1–4, 2020.
- [24] Q. Shambour, S. Alkhatib, M. Abualhaj and Y. Alrab'nah, “Effective voice frame shrinking method to enhance VoIP bandwidth exploitation,” *International Journal of Advanced Computer Science and Applications (IJACSA)*, vol. 11, no. 7, pp. 313–319, 2020.
- [25] Q. Kharma, A. Hussein, F. Taweel, M. Abualhaj and Q. Shambour, “Investigation of techniques for VoIP frame aggregation over A-MPDU 802.11 n,” *Intelligent Automation and Soft Computing*, vol. 31, no. 2, pp. 869–883, 2022.