Zeroize: A New Method to Improve the Utilization of 5G Networks When Running VoIP over IPv6

Manjur Kolhar

Department of Computer Science, College of Arts and Science, Prince Sattam Bin Abdulaziz University, Wadi Ad Dawaser 11990, Saudi Arabia; manjur.kolhar@gmail.com

Abstract: 5G technology is spreading extremely quickly. Many services, including Voice Over Internet Protocol (VoIP), have utilized the features of 5G technology to improve their performance. VoIP service is gradually ruling the telecommunication sector due to its various advantages (e.g., free calls). However, VoIP service wastes a substantial share of the VoIP 5G network’s bandwidth due to its lengthy packet header. For instance, the share of the packet header from bandwidth and channel time reaches 85.7% of VoIP 5G networks when using the IPv6 protocol. VoIP designers are exerting considerable efforts to solve this issue. This paper contributes to these efforts by designing a new technique named Zeroize (zero sizes). The core of the Zeroize technique is based on utilizing the unnecessary fields of the IPv6 protocol header to keep the packet payload (voice data), thereby reducing or “zeroizing” the payload of the VoIP packet. The Zeroize technique substantially reduces the expanded bandwidth of VoIP 5G networks, which is reflected in the wasted channel time. The results show that the Zeroize technique reduces the wasted bandwidth by 20% with the G.723.1 codec. Therefore, this technique successfully reduces the bandwidth and channel time of VoIP 5G networks when using the IPv6 protocol.

Keywords: 5G network; IPv6; VoIP; bandwidth utilization; header overhead

1. Introduction

The technology world is continually changing and developing to advance the current technologies with new services and features that make people’s lives easier, faster, and better. As an advancement of 4G, 5G technology opens up the horizon to improve many other services and technologies, owing to its superior bandwidth and download speeds [1]. The number of subscribers of 5G technology is predicted to increase 10 times by 2025 and reach 1.7 billion [2]. Many technologies will be improved by utilizing the features of 5G, such as Voice Over Internet Protocol (VoIP).

VoIP is promising technology that brings substantial benefits for subscribers. VoIP allows subscribers to communicate to anyone around the world freely by using smartphones or any other smart devices. This condition is because the VoIP data move in IP packets over the data connection. The number of mobile VoIP users is expected to be more than 3 billion. The Internet transported more than 156 petabytes of VoIP traffic every month in 2015 [3,4]. Mobile VoIP technology necessitates minimal latency in data transmission. Accordingly, one of the services that will remarkably improve when working with 5G technology is VoIP. VoIP systems are anticipated to wholeheartedly embrace 5G technology. VoIP will benefit from the high speed of 5G to enhance the call quality up to user satisfaction. VoIP will also benefit from the high bandwidth of 5G to tremendously maximize the call capacity [5,6].

Despite these possible benefits, VoIP is wasting a considerable share of the 5G technology bandwidth with no valuable use. This condition is due to the lengthy header of the VoIP packet compared with its payload [7,8]. For instance, the G.728 codec produces 10-byte digital voice data (packet payload). In the case of IPv4, the header of the VoIP packet, which consists of 40-byte RTP/UDP/IPv4 (12-byte RTP, 8-byte UDP, and 20-byte
IPv4), wastes up to 80% of the bandwidth specified for a VoIP call. The problem even worsens when it comes to IPv6. Specifically, the header of the VoIP packet, which consists of 60-byte RTP/UDP/IPv6 (40-byte IPv6), wasted up to around 85.7% of the bandwidth specified for a VoIP call. This consumption is a huge waste of bandwidth resources. Table 1 shows the apportioned bandwidth of different voice codes [9–11]. A huge amount of wasted bandwidth is evident with all the codecs. Theoretically, this process also consumes the same percentage of transmission time of the 5G networks.

Table 1. Apportioned bandwidth of different voice codes.

| VoIP Packet          | Packet Length | Voice Data Length | Apportioned Bandwidth |
|----------------------|---------------|-------------------|-----------------------|
| IPv4/G.728           | 50            | 10                | 80%                   |
| IPv6/G.728           | 70            | 10                | 85.7%                 |
| IPv4/LPC             | 54            | 14                | 74%                   |
| IPv6/LPC             | 74            | 14                | 81%                   |
| IPv4/G.723.1         | 60            | 20                | 66.7%                 |
| IPv6/G.723.1         | 80            | 20                | 75%                   |
| IPv4/G.726           | 70            | 30                | 57.1%                 |
| IPv6/G.726           | 90            | 30                | 66.7%                 |

The propagation of the IPv6 protocol has increased year after year. IPv6 has emerged from the “Innovators” and “Early Adoption” stages of deployment and is now in the “Early Majority” phase [12]. According to the authors of [12], the rate of global adoption of IPv6 increased from 2.8% in 2014 to 23.48% as of 2018. The worldwide adoption of IPv6 has been growing intensely at an average of more than 51% each year [13]. The integration of VoIP and IPv6 over 5G networks is inevitable. Therefore, the above problem of wasting the resources of 5G networks should be considered and carefully investigated. VoIP developers have developed many solutions to the lengthy header of VoIP. These solutions include combining the VoIP packet in one header, compacting the packet’s header, replacing the VoIP header protocols with new VoIP-specific protocols, and utilizing the header’s superfluous fields to carry the voice data of the packet [11,14,15].

The main advantage of the last solution of utilizing the superfluous header’s fields is that many of the proposed methods under it are compatible with the existing standards and equipment. Thus, they are workable in the current environment without major changes. However, none of these methods have been developed for VoIP when using IPv6, specifically over 5G networks. In this article, a new method is proposed to enhance the bandwidth utilization of 5G networks when running VoIP on IPv6. The proposed method uses the superfluous fields of the IPv6 protocol to carry the digital voice data of the packet.

Accordingly, the proposed method reduces the payload size or makes it zero. This condition enhances the bandwidth utilization of the used networks, such as 5G networks. The occupied airtime to transmit the VoIP packet over 5G networks is reduced. This condition indirectly affects the voice connection quality due to the decrease in the packet loss and delay.

The rest of this paper is organized as follows. Section 2 introduces the main solutions that handle the bandwidth problem of VoIP. Section 3 presents the proposed method, including its core idea of handling the bandwidth issue using the IPv6 protocol fields to carry the packet payload. Section 4 discusses the performance of the proposed method versus other comparable methods. Section 5 summarizes the findings of this study.

2. Related Work

The bandwidth obstacle has slowed the propagation of VoIP for some time. Many solutions have been proposed to overcome this obstacle and accelerate the adoption of VoIP. One of the first solutions suggested is collecting the packets sharing the same path and encapsulating them in one header, known as the packet or frame aggregation standard. Some of these methods have been designed to aggregate the frames at layer two in one
header, whereas others have been designed to aggregate the packets at layer three in one header. Performing the aggregation at layer two accomplishes better bandwidth utilization. Maximizing the number of aggregated packets, which is based on the parameters of the environment, saves more bandwidth [3,16]. Seytnazarov et al. presented one of the best aggregation methods in 2018. Seytnazarov’s method works in the IEEE 802.11n wireless networks. At the transmitter, several layer two frames (i.e., multiplexing MAC protocol data units (MPDUs)) are grouped in one aggregation (A-MPDU). If one of the grouped MPDUs is spoiled, then it causes the resending of only that MPDU. The frames in the suggested method are split into three groups: voice, video, and streaming groups with priority queuing (PQ), which assumes scheduling. The A-MPDU’s allowable length fluctuations adapt in accordance with the urgency delay (UD), which is the remaining period to dequeue the frame. The frame with the minimal UD is the first frame added to the A-MPDU. The allowable aggregation time of the A-MPDU is set on the basis of the remaining UD of this frame. In addition to frame aggregation, a new scheduling mechanism is suggested to handle the drawbacks of the PQ scheduler under saturated conditions. The analysis of the suggested technique showed a more advanced throughput rate with minor delay and packet loss than the other comparable methods [16].

Another established solution for the bandwidth snag is compacting the header of the packet, known as header compression, by eliminating some of the header’s fields. The other end of the call can obtain their values by using simple derivations because they are following certain patterns. The header is compacted up to two bytes when using IPv4, which greatly conserves the bandwidth. William A. Flanagan designed one of the recent compression methods in 2020, which was named “Session Bridging”. This method is a new customization of header compression that reduces bandwidth poverty. It splits the links into end chunks marked by distinct protocol instances using standard headers. A chunk segment between them conveys voice data with compressed headers. This method takes the benefit of multiprotocol label switching and virtual Ethernet connections in current networks. The analysis of certain networks proves that session bridging is viable for latency-sensitive applications (e.g., VoIP and gaming) [17].

Some VoIP developers have suggested creating other VoIP-specific protocols, such as the Inter-Asterisk Exchange (IAX) protocol. Although IAX can carry all types of streaming media, it is mainly developed for voice calls over IP commuter networks. IAX uses its 4-byte mini-header to carry the digital voice data and replace the 12-byte RTP protocol, which saves good bandwidth. The IAX protocol supports trunking, in which voice data from several sessions are combined in one stream of packets. This condition saves a huge share of bandwidth consumed by the IP header without extra delay [18].

One of the recently considered solutions is to utilize the header’s superfluous fields to carry the voice data of the packet. This solution is based on the assumption that the VoIP packet header protocol transfers various types of application data. Therefore, some of the fields in these protocols are necessary with some applications, but they are not with some other applications. The latter are burdening the bandwidth without valuable use. These fields are called the superfluous fields [14,15]. This last solution to the bandwidth problem uses the superfluous fields to carry the voice data of the packet, thereby reducing or eliminating the payload and saving bandwidth. One of the best methods under this solution was proposed by Abualhaj et al., and it is named short voice frame (SVF). The SVF method successfully utilizes the seven superfluous fields of the packet header on the basis of certain assumptions. These fields manage to save 17 bytes of each VoIP packet, which is typically between 50 and 70 bytes. The analysis results proved that the saved bandwidth can reach around 29% in some instances [14]. Another method called zero size payload (ZSP) is developed to utilize superfluous fields. The ZSP method uses different assumptions compared with the SVF method. Therefore, it manages to utilize the 10 superfluous fields with 19 bytes. The analysis results confirmed that the saved bandwidth can reach around 32% in some instances [15].
Developers have exerted great efforts to save bandwidth when running VoIP. The bandwidth issue becomes difficult when integrating the VoIP and IPv6 protocol, owing to the large size of the IPv6 header. However, the bandwidth problem of the VoIP and IPv6 protocol is not addressed, especially with 5G wireless networks. In this study, a new method is designed to solve this problem using the superfluous fields in the VoIP packet header. The proposed method uses the source IPv6 address to carry the digital voice data of the VoIP packet. This process minimizes the VoIP packet payload size or makes it zero; thus, the designed method is called the Zeroize (zero sizes) method. The Zeroize method enhances bandwidth utilization and improves the quality of the calls.

3. Zeroize Method Design

This section discusses the working mechanism of the designed Zeroize method. The main aim of the Zeroize method is to increase the capacity of the 5G network channel when running the VoIP over IPv6 protocol. The Zeroize method is applied to a caller client (e.g., Xlite) and intermediary wireless equipment (e.g., wireless router). Implementing the Zeroize method on either of them can enhance the channel capacity with the same ratio. This study assumes that the Zeroize method can be applied to the caller client. Figure 1 shows a topology in which the Zeroize method can be implemented. Though the assumed environment is 5G, the Zeroize method can be implemented with all technologies. In addition, the Zeroize method can be implemented in a LAN and private WAN, aside from the Internet (Public WAN).

![Zeroize method topology](image)

**Figure 1.** Zeroize method topology.

3.1. Core Idea

The Zeroize method uses the superfluous fields of the VoIP packet header to increase the call capacity. The superfluous fields are used to keep the digital voice data of the packet. The studies in [14,15] considered several fields of the VoIP packet as superfluous for VoIP applications on the basis of certain rules. Several fields throughout the RTP, UDP, and IPv4 meet the rules and are deemed superfluous, including the source IPv4 address. However, the Zeroize method is intended for the VoIP over IPv6 protocol. The size of the source IPv6 address is 128-bit (16 bytes), which is greater than any common codec frame, as shown in Table 1. Therefore, only the source IPv6 address is deemed a superfluous field and used to keep the digital voice data of the packet.

3.2. Source IPv6 Address

In packet-based networks, the purpose of the source IPv6 address is to identify the sender of the data. The receiver uses this source IPv6 address to send back a response to the sender [19]. In VoIP applications, the call is started over two stages, namely signaling and media transfer. In the signaling stage, a specific protocol (e.g., SIP protocol) is utilized to start the call and agree on the call parameters, including the IPv6 address of each call.
end. In other words, each of the calls knows the IPv6 address of the other end at this stage. In the media transfer stage, a different specific protocol is utilized (e.g., RTP protocol) to carry the voice data \[20,21\]. The RTP, with the help of other protocols (i.e., UDP/IPv6), uses the call parameters from the first stage, including the IPv6 address, to send the voice data to the intended destination. The point is that the source IPv6 address is included in every voice data packet (media packets). However, the VoIP media transfer session is not request–response, and the callee client does not send back a response to the sender by itself. In case the callee replies to the caller, the callee client uses the source IPv6 address, which is known from the first stage \[14,15,19\]. In addition, the other network devices (e.g., switches, routers, or firewalls) are not affected. Typically, the switch is a layer two device while the IPv6 protocol is layer three, where the router is using the destination IPv6 address to perform its main function of routing and the firewall denies a VoIP call during the call setup by SIP or H323 VoIP signaling protocols, but not while transferring the voice media data using RTP/UDP/IP protocols. That aside, none of the other protocols will be affected by this design. Accordingly, the source IPv6 address in the voice data packets is superfluous and can be assigned a different purpose, such as carrying the voice data itself. This condition reduces the consumed bandwidth by each call and increases the channel call capacity.

3.3. Carrying the Voice Data

The Zeroize method uses the source IPv6 address of the packet to carry the voice data. The sender client extracts the voice data from the packet. The voice data are then placed in the source IPv6 address field. The remaining bits of the IPv6 address are padded because the voice data are small. This process changes the packet payload length and affects the payload length field in the IPv6 header and the length field in the UDP header \[19\]. Therefore, the value of these fields should be modified appropriately on the basis of the new size. The pseudocode in Figure 2 clarifies the algorithm of the VoIP client at the sender side. Lines 1–6 are used to identify the variables of the pseudocode. Lines 8–12 are used to place the voice data in the source IPv6 address field and set the remaining bits (of the IPv6 address field) to zeros. Lines 13–17 are used to modify the value of the Payload Length field in the IPv6 header and the value of the Length field in the UDP header. The VoIP client extracts the voice data from the source IPv6 address field and places it in the playout buffer at the receiver. Only the voice data must be extracted without the padding. For this process, the VoIP client should check the used codec to specify the length of the voice data. The used codec is specified in the signaling stage of the VoIP call. Figures 3 and 4 clarify the process performed at the sender and receiver sides by the VoIP client, respectively.

| Zeroize Method | Sender VoIP Client |
|----------------|--------------------|
| 1 //L is length of voice data in bits //Integer |
| 2 //L-Padding is length of padding //Integer |
| 3 //SrcIPv6 is the source IPv6 address // String |
| 4 //Padding is the zeros to be append to SrcIPv6 // String |
| 5 //L-Pld is the value of Payload Length field in the IPv6 header //Integer |
| 6 //Len is the value of Length field in the UDP header //Integer |

7 For each VoIP packet {
8     // Extract the voice data;
9     L = Length (voice data);
10    L-Padding = 128 – L;
11    Padding = L-Padding number of zeros;
12    SrcIPv6 = voice data + Padding;
13    L-Pld = 20;
14    Len = 12;
15    If L>16 Then
16        L-Pld = 20 + (L / 8 - 16)
17        Len = 12 + (L / 8 - 16)
18   }

Figure 2. Zeroize method algorithm: sender side.
Zeroize Method

1. \( L \) is length of voice data //Integer
2. \( L\)-Padding is length of padding //Integer
3. \( \text{SrcIPv6} \) is the source IPv6 address // String
4. \( \text{Padding} \) is the zeros to be append to \( \text{SrcIPv6} \) // String
5. \( L\)-Pld is the value of Payload Length field in the IPv6 header //Integer
6. \( \text{Len} \) is the value of Length field in the UDP header //Integer
7. For each VoIP packet {
   8. Extract the voice data;
   9. \( L = \) Length (voice data);
   10. \( L\)-Padding = 128 – \( L \);
   11. \( \text{Padding} = \) \( L\)-Padding number of zeros;
   12. \( \text{SrcIPv6} = \) voice data + \( \text{Padding} \);
   13. \( L\)-Pld = 20;
   14. \( \text{Len} = 12 \);
   15. If \( L > 16 \) Then {
      16. \( L\)-Pld = 20 + (\( L \) / 8 - 16)
      17. \( \text{Len} = 12 + (\( L \) / 8 - 16) \)
   18. }

Figure 2. Zeroize method algorithm: sender side.

Separate the voice data from the packet header.

Step 1:
Place the voice data in the packet header: 16-byte in “Source IPv6 Address”

Step 2:
Update the “Payload Length” and the “Length” fields.

Step 3:
The remaining voice data (if any) is attached to the packet header.

Step 4:
Transmit the packet.

Figure 3. Zeroize method process: sender side.

4. Zeroize Method Performance Analysis

This section analyzes the proposed Zeroize method. Although intended for 5G networks, the Zeroize method applies to many other wireless technologies (e.g., IEEE 802.11). The proposed method is analyzed on the basis of the layer three header and above because the layer two header changes depending on the used technology. The Zeroize method is tested and analyzed versus the standard RTP (S-RTP method) method of transferring the VoIP packets, the SVF method proposed in [14], and the ZSP method proposed in [15]. The SVF and ZSP methods were selected because they are similar to the proposed Zeroize method. The SVF and ZSP methods share the same primary objective as the Zeroize method, conserving the network bandwidth when using VoIP applications. The SVF and
ZSP methods use the same main concept to save bandwidth: carrying the voice data in the superfluous fields of the packet header. However, the SVF and ZSP methods work over IPv4 networks and are intended only for point-to-point calls. The proposed Zeroize method was tested versus the other methods in terms of channel capacity and apportioned bandwidth reduction. The LPC, G.726, and G.723.1 codecs were assumed for the comparison.

Network Simulation 3 (NS3) was used to simulate the proposed Zeroize method. After simulating the Zeroize method, a network topology using NS3 was designed to evaluate the Zeroize method and compared with the SVF and ZSP methods. The simulation topology included two components of the Zeroize method, which were situated at the sender and receiver sides, as discussed in Section 3. Each component utilized a buffer with a maximum size of five. The connection between the two components was mimicked as a first-in-first-out buffer. A constant bit rate traffic generator was attached to each end.

4.1. Channel Capacity (Ch-Capacity)

The Ch-Capacity of the proposed Zeroize method is presented in this section. The Ch-Capacity was analyzed for each channel bandwidth between 100 and 1000 kbps. Losing the packets designates that the channel is saturated. Hence, the Ch-Capacity of a channel equals the number of calls before losing the packets and is measured accordingly. Figures 4–6 explore the Ch-Capacity for the Zeroize method in comparison to the S-RTP, SVF, and ZSP methods with G.723.1, G.726, and LPC codecs, respectively. The Ch-Capacity when using the Zeroize method outperformed (i.e., more calls) the S-RTP method. The difference in the Ch-Capacity increased with the existing bandwidth. This condition was due to eliminating the payload by utilizing the superfluous field (source IPv6 address) of the header to carry the payload. However, the Ch-Capacity when using the Zeroize method underperformed (i.e., fewer calls) the SVF and ZSP methods. This condition was because of the header of the IPv6 protocol header being twice the IPv4 protocol header and consuming more bandwidth for each call. The size of the superfluous fields of the SVF and ZSP methods was more than that of the Zeroize method. The Ch-Capacity of the SVF and ZSP methods was the same as the LPC codec. This condition was because the size of the LPC voice data was 14 bytes, which was smaller than the superfluous fields of the SVF method (17 bytes) and the ZSP method (19 bytes). Therefore, the superfluous fields of the two methods could carry the entire 14-byte voice data of the LPC codec. The extra two-byte superfluous fields of the ZSP method were not utilized when using a codec with voice data (e.g., LPC codec) smaller than the superfluous fields.

\[
\text{Available Bandwidth (kb)}
\]

\[
\begin{align*}
\text{No. of Calls} & \quad \text{RTP Method} \quad \text{Zeroize Method} \quad \text{SVF Method} \quad \text{ZSP Method} \\
100 & \quad \bullet \quad \circ \quad \triangle \quad \star \\
200 & \quad \bullet \quad \circ \quad \triangle \quad \star \\
300 & \quad \bullet \quad \circ \quad \triangle \quad \star \\
400 & \quad \bullet \quad \circ \quad \triangle \quad \star \\
500 & \quad \bullet \quad \circ \quad \triangle \quad \star \\
600 & \quad \bullet \quad \circ \quad \triangle \quad \star \\
700 & \quad \bullet \quad \circ \quad \triangle \quad \star \\
800 & \quad \bullet \quad \circ \quad \triangle \quad \star \\
900 & \quad \bullet \quad \circ \quad \triangle \quad \star \\
1000 & \quad \bullet \quad \circ \quad \triangle \quad \star \\
\end{align*}
\]

Figure 5. Ch-Capacity (G.723.1 codec).
4.1. Channel Capacity (Ch-Capacity)

The Ch-Capacity of the proposed Zeroize method is presented in this section. The Ch-Capacity was analyzed for each channel bandwidth between 100 and 1000 kbps. Losing the packets designates that the channel is saturated. Hence, the Ch-Capacity of a channel equals the number of calls before losing the packets and is measured accordingly. Figures 4–6 explore the Ch-Capacity for the Zeroize method in comparison to the S-RTP, SVF, and ZSP methods with G.723.1, G.726, and LPC codecs, respectively. The Ch-Capacity when using the Zeroize method outperformed (i.e., more calls) the S-RTP method. The difference in the Ch-Capacity increased with the existing bandwidth. This condition was due to eliminating the payload by utilizing the superfluous field (source IPv6 address) of the header to carry the payload. However, the Ch-Capacity when using the Zeroize method underperformed (i.e., fewer calls) the SVF and ZSP methods. This condition was because of the header of the IPv6 protocol header being twice the IPv4 protocol header and consuming more bandwidth for each call. The size of the superfluous fields of the SVF and ZSP methods was more than that of the Zeroize method. The Ch-Capacity of the SVF and ZSP methods was the same as the LPC codec. This condition was because the size of the LPC voice data was 14 bytes, which was smaller than the superfluous fields of the SVF method (17 bytes) and the ZSP method (19 bytes). Therefore, the superfluous fields of the two methods could carry the entire 14-byte voice data of the LPC codec. The extra two-byte superfluous fields of the ZSP method were not utilized when using a codec with voice data (e.g., LPC codec) smaller than the superfluous fields.

The Ch-Capacity was also calculated using mathematical equations to validate the results. The bandwidth needed by each call depends on the used codec and the packet header length. Equation (1) can be used to find the bandwidth needed by each call:

\[ C_{bw} = \frac{(L_{pld} + L_{h}) \cdot \left( \frac{C_{br}}{8} \right)}{1000} \times 8 \]  

where \( C_{bw} \) is the bandwidth needed by each call, \( L_{pld} \) is the packet payload length, \( L_{h} \) is the packet header’s length, and \( C_{br} \) is the bit rate of the used codec. For instance, assume the G.728 codec is used with a bit rate of 16 kbps, and the IPv6 protocol is in the packet header. Then, the bandwidth needed by each call when using the Zeroize method is 96 kbps. On the other hand, the bandwidth needed by each call when using the S-RTP method is 112 kbps. Therefore, the saving in the Ch-Capacity is around 14.3% when using the G.728 codec. The saving in the Ch-Capacity varies from one codec to another, based on the voice data length of each codec.

4.2. Apportioned Bandwidth Reduction

The apportioned bandwidth ratio of the proposed Zeroize method is discussed in this section. The apportioned bandwidth ratio was analyzed with 10–100 calls. The Zeroize, SVF, and ZSP methods were analyzed in comparison to the S-RTP method. Figures 7 and 8 show the bandwidth reduction of the Zeroize, SVF, and ZSP methods in comparison to the S-RTP method. The apportioned bandwidth saving when using the Zeroize method was around 20% compared with that of the S-RTP method with the G.723.1 codec. However, the bandwidth saving when using the Zeroize method underperformed the SVF and ZSP methods. This condition was because the header of the IPv6 protocol header was twice the IPv4 protocol header and consumed more bandwidth for each call. The size of the superfluous fields of the SVF and ZSP methods was more than that of the Zeroize method. The SVF and ZSP methods outperformed the Zeroize method in terms of Ch-Capacity and apportioned bandwidth reduction. However, the Zeroize method works over IPv6 networks, unlike the SVF and ZSP methods, which work over IPv4 networks. The SVF and ZSP methods utilize many fields as superfluous fields to carry the voice data. This condition makes them workable only in specific scenarios, which are the scenarios that they are designed for. However, the Zeroize method utilizes only the source IPv6 address field as a superfluous field to carry the voice data. This condition makes the Zeroize method a more general method that works in all scenarios. The SVF and ZSP methods burden
the network compared with the Zeroize method for several reasons. The SVF and ZSP methods are designed to work at the gateway routers. Therefore, the resources of these routers are consumed, especially with the high number of calls. The Zeroize method is designed to work at the client-side without burdening the routers. The Zeroize method uses only one field (source IPv6 address) to carry the voice data without the need to set back the original value of this field. However, the SVF and ZSP methods use several fields to carry the voice data, thereby imposing more operations and calculations, such as setting back the original value of the superfluous fields at the receiver gateway, calculating the new length field value of the UDP protocol, and assigning a new value to the Internet header length field of the IP protocol. Performing these operations for every call packet consumes the routers’ resources, especially with a high number of calls. The SVF and ZSP methods burden the network compared with the Zeroize method.

5. Conclusions

VoIP services provide many features that make them extremely wide-reaching in the telecommunication sector. The integration of VoIP and IPv6 over 5G networks is unavoidable. This integration leads to wasting up to 85.7% of the bandwidth and airtime...
of a 5G network. This study contributes to the efforts to solve this issue by proposing the Zeroize method. The proposed method succeeds in using the source IPv6 address field in the IPv6 protocol header to carry the packet’s digital voice data and reduce or eliminate the VoIP packet payload. The sender client extracts the voice data from the packet and places it in the source IPv6 address field. At the receiver, the VoIP client extracts the voice data from the source IPv6 address field and places it at the end of the packet as a normal payload. The Zeroize method was compared with the S-RTP, SVF, and ZSP methods by using three different codecs (LPC, G.726, and G.723.1). The Zeroize method reduced the wasted bandwidth by 20% compared with the S-RTP method with the G.723.1 codec. The proposed method is a promising solution to reduce the wasted bandwidth and airtime of 5G networks when running VoIP over the IPv6 protocol. The effect of altering the original value of the source IPv6 address field and software-defined networking-based networks will be investigated in future works.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The author declares no conflict of interest.

References
1. Kalogiros, C.; Zois, G.; Darzanous, G.; Hallingby, H.K.; Lønsethagen, H.; Weiss, M.B.; Gavras, A. The potential of 5G experimentation-as-a-service paradigm for operators and vertical industries: The case of 5G-VINNI facility. In Proceedings of the 2019 IEEE 2nd 5G World Forum (5GWF), Dresden, Germany, 30 September–2 October 2019; pp. 347–352. [CrossRef]
2. Positive 5G Outlook Post COVID-19: What Does It Mean for Avid Gamers? Forest Interactive. Available online: https://www.forest-interactive.com/newsroom/positive-5g-outlook-post-covid-19-what-does-it-mean-for-avid-gamers/ (accessed on 13 November 2020).
3. Abualhaj, M.M.; Hussein, A.H.; Kolhar, M.; AlHija, M.A. Survey and analysis of VoIP frame aggregation methods over A-MSDU IEEE 802.11 n wireless networks. Comput. Mater. Contin. 2020, 66, 1283–1300. [CrossRef]
4. 16 Important VoIP Stats to Know in 2021. Available online: https://hostingtribunal.com/blog/voip-stats/#gref (accessed on 3 March 2021).
5. How Will 5G Improve Your VoIP Service? Available online: https://www.toolbox.com/collaboration/team-collaboration/articles/how-will-5g-improve-your-voip-service/ (accessed on 3 March 2021).
6. Volkov, A.; Baskakov, A.; Muratchaev, S.; Volkova, E. Development of a Complex Model for VoIP Technology with the Possibility of Application in 5G Networks. In Proceedings of the 2018 IEEE East-West Design & Test Symposium (EWDTS), Kazan, Russia, 14–17 September 2018; pp. 1–5. [CrossRef]
7. Karjee, J.; Khat, S.; Sarkar, D.; Tammineedi, H.L.C.; Chavva, A.K.R. 5G-NR Cross Layer Rate Adaptation for VoIP and Foreground/Background Applications in UE. In Proceedings of the 2020 IEEE 3rd 5G World Forum (5GWF), Bangalore, India, 10–12 September 2020; pp. 80–85. [CrossRef]
8. Abualhaj, M.M.; Al-Khatib, S.N.; Kolhar, M.; Munther, A.; Alrabab‘nah, Y. Effective Voice Frame Pruning Method to Increase VoIP Call Capacity. TEM J. 2020, 9, 48–54.
9. Narayan, S.; Gordon, M.; Branks, C.; Fan, L. VoIP network performance evaluation of operating systems with IPv4 and IPv6 network implementations. In Proceedings of the 2010 3rd International Conference on Computer Science and Information Technology, Chengdu, China, 9–11 July 2010; pp. 669–673. [CrossRef]
10. Gupta, N.; Kumar, N.; Kumar, H. Comparative Analysis of Voice Codecs over Different Environment Scenarios in VoIP. In Proceedings of the 2018 Second International Conference on Intelligent Computing and Control Systems (ICICCS), Madurai, India, 14–15 June 2018; pp. 540–544. [CrossRef]
11. Abualhaj, M.M.; Shambour, Q.Y.; Hussein, A.H. Effective packet multiplexing method to improve bandwidth utilisation. Int. J. Comput. Appl. Technol. 2020, 63, 327–336. [CrossRef]
12. State of IPv6 Deployment 2018. Available online: https://www.internetsociety.org/resources/2018/state-of-ipv6-deployment-2018/#:~:text=Many%20broadband%20ISPs%20have%20IPv6,measurement%20of%20IPv6%20deployment%2066%25 (accessed on 3 March 2021).
13. PERSPECTIVE: Future of the Internet Depends on Global Adoption of IPv6. Available online: https://www.hstoday.us/subject-matter-areas/cybersecurity/future-of-the-internet-depends-on-global-adoption-of-ipv6/ (accessed on 3 March 2021).
14. Abualhaj, M.M.; Al-Tahrawi, M.M.; Al-Zyoud, M. Contracting VoIP Packet Payload Down to Zero. Cybern. Inf. Technol. 2021, 21, 137–150.
15. Abualhaj, M.M.; Shambour, Q.Y.; Hussein, A.H.; Kharma, Q.M. Down to Zero Size of VoIP Packet Payload. *CMC-Comput. Mater. Contin.* **2021**, *68*, 1271–1283. [CrossRef]

16. Seytnazarov, S.; Kim, Y.T. Qos-aware adaptive a-mpdu aggregation scheduler for enhanced voip capacity over aggregation-enabled wlans. In Proceedings of the IEEE/IFIP Network Operations and Management Symposium, Taipei, Taiwan, 23–27 April 2018; pp. 1–7.

17. Flanagan, W.A. Header Compression Across Entire Network Without Internet Protocol Saves Bandwidth and Latency. In Proceedings of the 2020 IEEE 3rd 5G World Forum (5GWF), Bangalore, India, 10–12 September 2020.

18. Spencer, M.; Capouch, B.; Guy, E.; Miller, F.; Shumard, K. *Iax: Inter-Asterisk Exchange Version 2*. Internet Request for Comments. 2010. Available online: https://en.wikipedia.org/wiki/Request_for_Comments (accessed on 2 August 2021).

19. Hagen, S. *IPv6 Essentials: Integrating IPv6 into Your IPv4 Network*, 3rd ed.; O’Reilly Media: New York, NY, USA, 29 June 2014.

20. Khudher, A.A. Sip aspects of ipv6 transitions: Current issuesand future directions. *J. Eng. Sci. Technol.* **2019**, *14*, 448–463.

21. Abualhaj, M.M.; Al-Khatib, S.N.; Shambour, Q.Y. PS-PC: An Effective Method to Improve VoIP Technology Bandwidth Utilization over ITTP Protocol. *Cybern. Inf. Technol.* **2020**, *20*, 147–158. [CrossRef]