Acta Polytechnica **63**(5):347–355, 2023
https://doi.org/10.14311/AP.2023.63.0347

IMPROVING QOS IN MOBILE MULTIMEDIA STREAMING WITH SCTP-PQ

Alisettar Hüseynli[a,*], Mehmet Şimşek[b], M. Ali Akcayol[a]

[a] *Gazi University, Faculty of Engineering, Department of Computer Engineering, Maltepe 06750, Ankara, Türkiye*

[b] *Düzce University, Faculty of Engineering, Department of Computer Engineering, Konuralp Campus 81620, Düzce, Türkiye*

[*] corresponding author: alisattar.huseynli@gazi.edu.tr

Abstract. The Stream Control Transmission Protocol (SCTP) is often the preferred transport layer protocol in streaming applications. This protocol combines the best aspects of Transmission Control Protocol (TCP) and User Datagram Protocol (UDP), but also offers additional features. SCTP supports multihoming and multi-streaming applications and has a congestion mechanism like TCP. Media streaming consists of different types of frames with different levels of importance. For example, I-frames carry more information than B-frames in Moving Picture Experts Group (MPEG). Usually, MPEG frames are processed using the First-In-First-Out (FIFO) algorithm. In this paper, a four-level priority queue integrated protocol named SCTP Priority Queue (SCTP-PQ) has been developed to reduce jitter and delay in real-time multimedia streaming for mobile devices. As part of the development, priority and retransmitted packets are determined on the sending side and these packets are processed faster by using the priority queue on the receiving side. In this way, the average queue delay of priority packets on the receiving side is reduced by 90 % and the throughput values are increased by an average of 10 times. The developed protocol has been extensively tested and compared with SCTP. The results show that the SCTP-PQ outperforms the standard SCTP in terms of jitter and delay.

Keywords: Priority queue, SCTP, delay, jitter, QoS.

1. Introduction

Stream Control Transmission Protocol (SCTP) was developed by the Internet Engineering Task Force (IETF) as a reliable transport protocol to transport Signalling System 7 (SS7) messages over the Internet Protocol (IP) networks [1]. With its advanced features, which are not provided by Transmission Control Protocol (TCP) or User Datagram Protocol (UDP), SCTP is able to support a wider range of applications than signalling transport [2].

Today, many types of mobile services are provided to customers over the Internet. An important part of those services is multimedia services such as IP Television (IPTV). Typical multimedia services use different layer protocols to ensure the required level of Quality of Service (QoS). When transporting multimedia data over the internet, some problems can occur, such as jitter, packet corruption, connection latency, and zapping latency [3]. Many different approaches have been used to improve QoS in multimedia streaming [4]. In the literature, there are many transport protocols for multimedia streaming [1, 5–12]. SCTP is the widely accepted transport protocol for IPTV [13, 14]. Today, SCTP is used in real-time multimedia standardised streaming applications [15–17].

SCTP is a connection-oriented protocol that aims to combine the fast operation of UDP with the reliability [18, 19], sequencing, and congestion control features of TCP. SCTP is designed to meet the needs of IP applications that require features not offered by TCP or UDP [20]. Like TCP, SCTP is an end-to-end and full duplex protocol [21]. SCTP uses a four-way handshake logic using cookies that remove the SYN to prevent Denial of Service (DoS) attacks on the host [22]. It is a message-oriented protocol, which helps the protocol keep states during operation and react upon events occurring in the network. It provides message packing and fragmentation to allow faster data transmission during initialisations just like TCP. SCTP detects duplicated, corrupted, discarded, or reordered packets [23]. These features are the main requirements in many of the current applications existing in IP networks. SCTP was originally designed to transmit voice packages over the Internet. It offers a number of features that are not offered by TCP and UDP, such as multi-streaming and multihoming [24–27].

TCP and UDP are not sufficient for this type of application, because higher layer protocols, such as H.248, H.323, and Session Initiation Protocol (SIP), require more complex services [13]. In addition, SCTP is fair to other SCTP connections and friendly with TCP [19]. SCTP provides ordered and reliable packet transmission. However, SCTP does not support priority at the receiver side [19]. As it is known, some data packets presented by the upper network layers have different priorities both within the same application (e.g. MPEG format I, B, and P frames) and between different applications (e.g. web browsing traffic vs. Voice over IP traffic). In addition, in real-time applications, retransmitted packets must be delivered to the target early for priority processing. Fast delivery of priority packets to the destination increases QoS and QoE.

In this study, four-level Priority Queue integrated SCTP (SCTP-PQ) was proposed to optimise the QoS of real-time multimedia streaming applications for mobile devices. SCTP-PQ supports four-level priority for different types of frames of real-time multimedia streaming clients. SCTP-PQ rearranges packets at the receiver side buffer according to defined priority levels to process higher priority packets earlier. The proposed SCTP-PQ has four priority levels for different types of frames. The retransmitted packets have the highest priority, level-0, because retransmitted packets should be played immediately. I-frames have level-1 priority because these frames can be processed independently. P-frames and B-frames have level-2 and level-3 priority, respectively (level-0<level-1<level-2<level-3). In this study, we focused on rearranging packets at the receiver side buffer in order to optimise the QoS of SCTP. The main aim of SCTP-PQ is to reduce delay and jitter for higher priority packets.

The contributions of the proposed SCTP-PQ can be summarised as follows:

- Reducing jitter and delay of real-time multimedia streaming.
- End-to-end prioritisation of multimedia data chunks using SCTP headers.
- The proposed SCTP-PQ does not require any changes to the existing infrastructure of hosts.
- It can be generalised and applied to other kinds of applications.

The rest of the paper is organised as follows: in Section 2, we present the related work, Section 3 presents detailed information about the proposed SCTP-PQ, Section 4 contains experimental results, Section 5 concludes the study.

2. Related work

SCTP is defined as a transport layer protocol by the IETF in RFC 2960 [18] as shown in Figure 1.

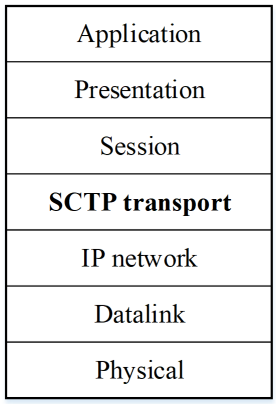


Figure 1. OSI model and SCTP.

SCTP is located between the network and the upper layers. The user data are taken from upper layers. Before being sent to the lower layers, the data are fragmented in the transport layer. On the receiver side, the data received from the lower layers are checked, reassembled, and delivered to the upper layer.

A multihomed SCTP can bind to multiple IP addresses [28]. Currently, SCTP uses multihoming only for redundancy, not for load balancing. The multihoming structure of SCTP is shown in Figure 2.

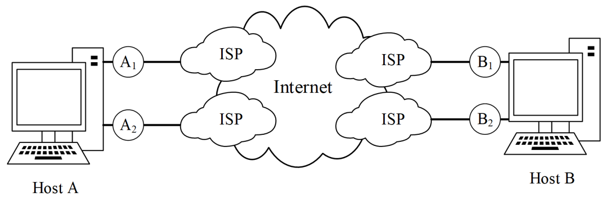


Figure 2. The multihoming structure of SCTP.

SCTP keeps track of the reachability of each of the destination addresses through two mechanisms: Acknowledgements (ACKs) of data chunks and heartbeat chunks that are control chunks, which periodically probe the status of the destination.

Multi-streaming is another novel service of SCTP, which is located at the transport layer. This is the logical separation of data within an association [29]. In SCTP, the stream is established from one SCTP endpoint to another. It is a one-way logical channel. The stream sequence number is used to maintain order and reliability for each data chunk [28]. However, no data rows are preserved between streams. This approach avoids the head-of-line blocking problem of TCP [30]. The multi-streaming structure of SCTP is shown in Figure 3.

SCTP congestion control is based on the proven mechanism used in TCP [21, 31]. However, there are some differences between congestion control of TCP and SCTP. The fast retransmission algorithm based on Selective Acknowledgment (SACK) is similar to TCP SACK [20]. However, instead of the clear fast recovery cycle, SCTP automatically provides fast recovery using SACK. SACK provides a more robust response than TCP in the case of multiple losses. This slows down the slow start phase after multiple segment losses, retains bandwidth, and improves throughput.

SCTP identified the two security objectives: the availability of reliable and timely data transport, and the integrity of the end-to-end data. Using a four-way handshake mechanism with a cookie, SCTP eliminates the risk of DoS attacks. SCTP can be used with IPSec (Internet Protocol Security) or TLS (Transport Layer Security) to protect the privacy and integrity of the

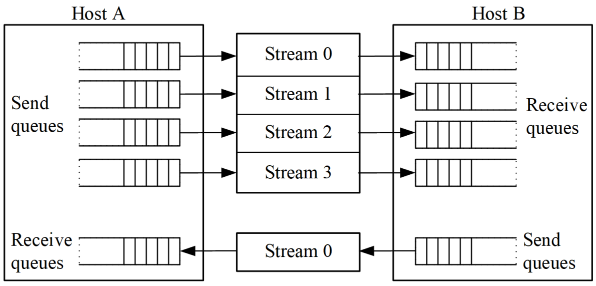


Figure 3. The multi-streaming structure of SCTP.

payload data [32]. The detailed comparison between SCTP, TCP, and UDP is shown in Table 1.

To improve QoS in real-time multimedia streaming, numerous studies have been done by many researchers [20, 32–37]. Stewart et al. proposed a partial reliable SCTP for QoS adaptation for IPTV over IP Multimedia Subsystem (IMS) Network [38]. The proposed protocol allows users to select the partial reliable service for each stream [17].

Derini et al. [24] have developed a modified version of SCTP to share bandwidth fairly and resolves network congestion by detecting the congestion state of the network and adapt the transfer rate of the video according to the congestion state.

Argyriou has developed media SCTP architecture for streaming H.264/AVC unicast video over the Internet [33]. The developed architecture has the capability of maintaining good perceptual quality under various loss conditions and maintains TCP-friendliness.

Ke and Chilamkurti have developed a content-aware packet marking scheme called Two Markers System (TMS). TMS marks the packets by pre-assigned importance levels at the video source and at the edge of DiffServ network [5]. In the event of congestion, TMS discards packets with lower importance.

Kim et al. [35] have developed a new protocol using a transmission control sublayer. In the proposed protocol, the multimedia streaming server determines whether the receiver will play the data before sending them.

Due to its multi-homing support, the SCTP protocol promises high success rates for concurrent multipath transfer (CMT) [39]. For this reason, different studies have been carried out to increase QoS such as CMT-SCTP [40, 41]. Verma et al. [42] propose a fast retransmission strategy based on delay times to improve the performance of SCTP. Arianpoo et al. [43] propose a reinforcement learning (RL) based method to learn the distribution of flows in CMT. Similarly, an approach using RL to increase throughput values is suggested by Yu et al. [44]. While these studies reduce latency and increase throughput values, they do not create a different approach to different types of upper-layer services. Another study to improve the CMT-SCTP performance with a dynamic scheduling approach was carried out by Wang et al. [45].

Services/Features	**SCTP**	**TCP**	**UDP**
Connection-oriented	yes	yes	no
Reliable data transfer	yes	yes	no
Partial-reliable data transfer	optional	no	no
Ordered data delivery	yes	yes	no
Unordered data delivery	yes	no	yes
Flow control	yes	yes	no
Congestion control	yes	yes	no
ECN (Explicit Congestion Notification) capable	yes	yes	no
Selective ACKs	yes	optional	no
Preservation of message boundaries	yes	no	yes
Path MTU (Maximum Transfer Unit) discovery	yes	yes	no
Application PDU (Protocol Data Unit) fragmentation	yes	yes	no
Application PDU bundling	yes	yes	no
Multistreaming	yes	no	no
Multihoming	yes	no	no
Protection against SYN flooding attacks	yes	no	n/a
Allows half-closed connections	no	yes	n/a
Reachability check	yes	yes	no

Table 1. Differences between TCP, UDP, and SCTP [37].

The transmission of audio and visual content via the Hypertext Transfer Protocol (HTTP) has attracted attention in recent years and with Dynamic Adaptive Streaming over HTTP (DASH) has become standard [46]. However, studies have shown a significant performance decline when streaming from a Web server that does not allow persistent connection [29]. In addition, the end-to-end latency of live streams has been increased and clients have to buffer a few fragments to ensure that their input buffers are not starved [28]. DASH itself has some shortcomings. Especially at peak times, multiple DASH clients have to share common network resources [47, 48].

Vavakananda et al. [49] proposed a priority manager for congestion avoidance in Mobile Ad hoc Networks (MANETs). They developed an application layer priority manager structure that is responsible for classifying data and effectively selecting a stream to send the data.

In literature, there is no approach that uses priority queue integrated SCTP structure on the receiver side and most of the studies have a significant overhead.

	Bits 0-7	8-11	12	13	14	15	16-31
0	Chunk type=0	**Reserved**	I	U	B	E	Chunk length
32	Transmission sequence number						
64	Stream identifier						Stream sequence number
96	Payload protocol identifier						
128	Data						

FIGURE 4. The frame structure of SCTP.

Bit value	Priority level
00	0- Retransmitted data chunks
01	1- I frames
10	2- P frames
11	3- B frames

TABLE 2. The priority levels and assigned bit values of SCTP-PQ.

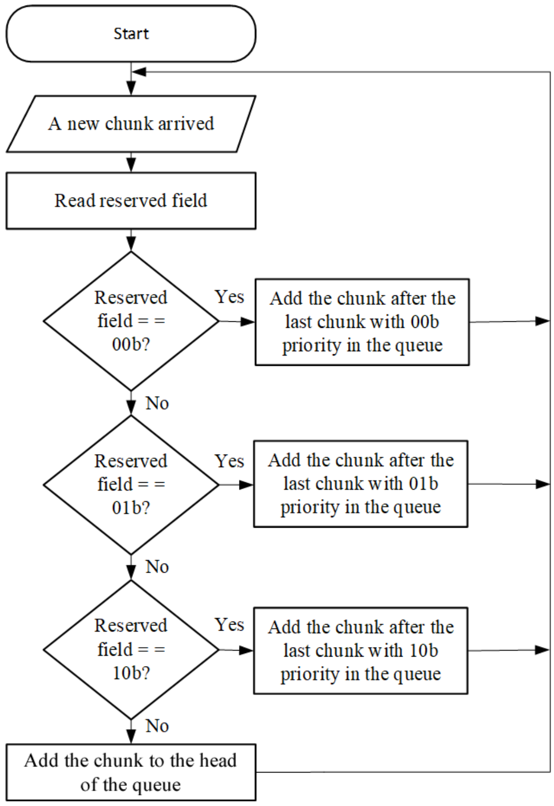


FIGURE 5. The flowchart of re-arranging algorithm of SCTP-PQ.

3. DEVELOPED SCTP-PQ PROTOCOL

This section will discuss the modification made to the conventional SCTP protocol for the proposed method. In addition, the prioritisation process on the sender side and the queue prioritisation on the receiver side depending on this process is explained in detail.

3.1. SCTP-PQ ARCHITECTURE

Real-time multimedia streaming is composed of different frames and each of them carries different types of data. For example, Moving Picture Experts Group (MPEG) I-frames carry more data than B-frames, therefore I-frames are more important than B-frames [29]. In addition, data packets to be retransmitted are more important than others because after a period of time, the data will not be played on the receiver side. SCTP sorts data packets according to Stream Sequence Number (SSN) for each flow at the receiver side buffer. As a result, a more important part of the data is waiting for less important data to be processed. For this purpose, optimising the processing order of packets by type is important. In this study, a four-level priority queue has been integrated into SCTP-PQ to optimise the processing order of the packets. The main aim of SCTP-PQ is to reduce delay and jitter for higher priority data packets, especially for the retransmitted data.

3.2. CONTENT-BASED PRIORITY IN SCTP-PQ

In SCTP-PQ, the sender such as a multimedia server determines the priorities of the data chunks. An SCTP frame has an 8-bit label on the data chunk header. The last 4 bits of the label are used for chunk flags (I, U, B, and E). The first 4 bits of the label are reserved for future use. The frame structure of SCTP is shown in Figure 4.

In this study, the first 2-bit of the reserved field has been used for the priority of the data chunks. SCTP-PQ determines 4 priority levels for MPEG-2. The priority levels and assigned bit values are shown in Table 2.

MPEG-2 format was used for the experimental studies. As can be seen in Table 2, the retransmitted data chunks have the highest priority, I-frames are second, P-frames are third, and B-frames have the fourth priority level.

3.3. RE-ARRANGING DATA CHUNKS AT RECEIVER SIDE IN SCTP-PQ

SCTP-PQ re-arranges data chunks according to priority levels which are defined in the header. The re-arranging algorithm is shown in Figure 5.

The linked list data structure has been used for the receiver-side buffer in SCTP-PQ. When a data chunk is received by the receiver side, first, SCTP-PQ extracts the priority label of the chunk using the first 2 bits of the reserved field. If the priority value equals “00”, which indicates the data chunks are retransmitted, SCTP-PQ adds the incoming data chunk to the head of the buffer. If the priority value is other than from “00”, the correct location in the buffer is found and the data chunk is inserted. The newly received chunk is placed at the relevant position in the queue according to its priority value. A well-known insertion sort algorithm has been used for the insertion process of the new data chunks. The time complexity of the insertion process is O(n). An important advantage of the SCTP-PQ is that it has no overhead because it uses the reserved fields in the header.

SCTP uses the advertised receiver window credit (a_rwnd) field on SACK. The sender adjusts the data rate according to a_rwnd. a_rwnd prevents buffer overflow at the receiver side. However, buffer overflows can still occur in rare cases. In order to prevent the important data chunks from being dropped, SCTP-PQ drops the data chunks according to the priority value in the case of a buffer overflow. SCTP-PQ discards the least important data chunk from the buffer.

Differences between the SCTP-PQ and standard SCTP are as follows:

- In SCTP-PQ, the sender such as a multimedia server determines the priority of the data chunks. In SCTP, data chunks have no priority.
- SCTP-PQ re-arranges data chunks according to priority levels that are defined in the header. SCTP sorts data chunks according to SSN.
- SCTP-PQ drops the data chunks according to priority in the case of a buffer overflow. SCTP drops data chunks from the tail of the buffer in the case of a buffer overflow.

4. Experimental results

In this study, simulations were performed on NS2 simulator [50] to compare the performance of SCTP-PQ and SCTP. Changes made to the SCTP protocol in the NS2 simulator can be found at `https://github.com/alisettar/sctp-pq`. The network topology used for experimental studies is shown in Figure 6.

In the network topology, there are seven sender nodes, seven receiver nodes, and two intermediate nodes. One 4-Mbps MPEG stream traffic has been established between S1-node and R1-node. and it has 256 kbps audio component. Simultaneously, six 0.5 Mbps background traffic connections have been generated between S2-node and R2-node, S3-node and R3-node, S4-node and R4-node, S5-node, and R5-node, S6-node and R6-node, S7-node and R7-node. In total, there is 3 Mbps background traffic. Background traffic can be defined as the traffic generated by web services, such as browsing, e-mail, etc., that may occur simultaneously with MPEG streaming. The Wi-Fi wireless carrier standard is preferred as the routing protocol for all mobile nodes.

The simulation duration in experimental studies is 120 seconds. Because the SCTP-PQ has properly sorted received packets, MPEG data rate, packet error rate, or link capacity does not affect the performance of SCTP-PQ. The performance of SCTP-PQ has been observed for different buffer sizes.

We focused on the following performance parameters:

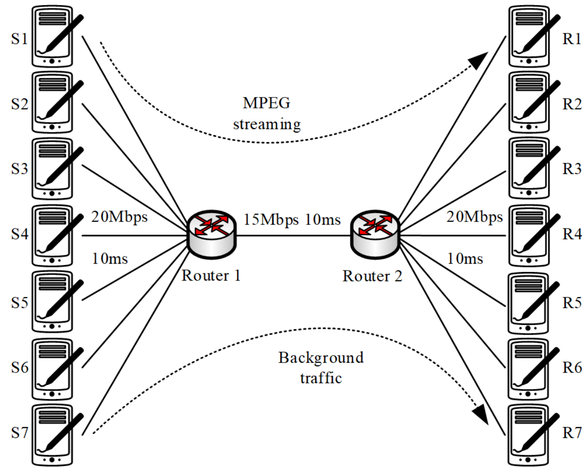
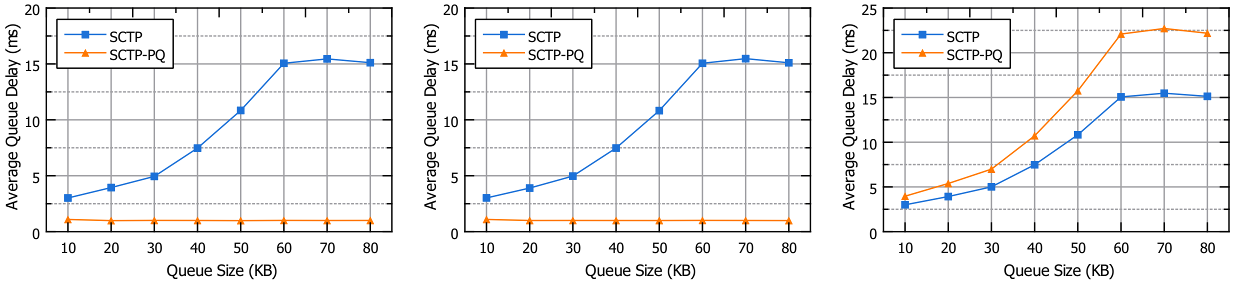


Figure 6. The network topology used for experimental studies.

- Average queue delay [ms] – Average queue delay of payload data chunks has been calculated for each priority level.
- Total queue delay [s] – The total queue delay of payload data chunks used to evaluate the correctness of the simulations. This value should be nearly the same for SCTP-PQ and SCTP.
- IP packet delay variation (IPDV) [ms] – IPDV is defined as “the difference in the one-way-delay of selected packets” in the IETF RFC 3393 [15]. IPDV has been calculated for each priority level.
- Jitter – Jitter is defined as the change (e.g., delay) of a metric relative to several reference metrics (e.g., average latency or minimum latency).
- The number of correctly received data chunks – This metric refers to the number of successfully received data chunks by the receiver. This value should be nearly the same for SCTP and SCTP-PQ.

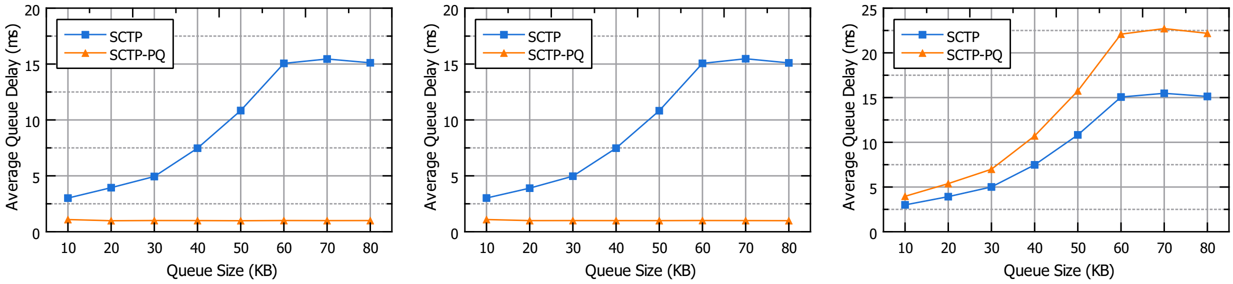
Average values of queue delay for all priority levels of data chunks are shown in Figures 7a, 7b and 7c.

As can be seen in the Figures 7a, 7b and 7c, the SCTP-PQ has more stable and much lower average queue delay for the first and the second priority level data chunks than the SCTP. Since the prioritisation process is done by the upper network layers in the proposed SCTP-PQ protocol, a performance improvement is expected on the receiver side of the prioritised packets in measurements, such as latency, IPDV values, and jitter. This improvement indirectly helps the expected packets to be processed earlier in the upper layers and to minimise the latency in the streaming process. Since there is no such prioritisation in a conventional SCTP protocol, all sent and received packets are processed in the same way. This, in turn, causes delayed processing of priority packets in the upper layers. The increase in the SCTP curve in Figures 7 and 8 is caused by the delayed processing of priority packets due to the packets created by the background



(A). First priority level. (B). Second priority level. (C). Third priority level.

FIGURE 7. Average queue delay values for different priority level data chunks.



(A). First priority level. (B). Second priority level. (C). Third priority level.

FIGURE 8. IPDV values for different priority level data chunks.

traffic in the buffer. In our simulations, the average queue delay for the SCTP for the highest priority data chunks is about 10 ms. This value is about 1 ms for SCTP-PQ. Also, SCTP has a lower average queue delay than SCTP-PQ for the third priority level data chunks. This result is caused due to the insertion of the third priority level data chunks to the end of the buffer.

I-frames have the second highest priority. SCTP-PQ adds newly received I-frame at the end of its priority frames. As opposed to SCTP, which adds all frames at the end of the queue. Let Q be receiver side queue; N_I, N_B, N_P and N_R be number of I, B, P, and R $(Retransmitted)$ frames in Q. $\mathbb{P}_I, \mathbb{P}_B, \mathbb{P}_P$ and $\mathbb{P}_R$ are the times required for the processing of I, B, P, and R frames. For a particular I frame, SCTP adds it to the end of Q. The time to process this frame is:

$$qd_1 = \sum_{k \in F} N_k \times \mathbb{P}_k \,, \tag{1}$$

where, $F = I, B, P, R$. SCTP-PQ adds this frame at the end of its priority group in Q. In SCTP-PQ, the time to process this frame is:

$$qd_2 = \sum_{k \in \{I,\ R\}} N_k \times \mathbb{P}_k \,. \tag{2}$$

As a result, $qd_2 \leq qd_1$ in the worst case. This reduces the delay for higher priority frames in the case of SCTP-PQ.

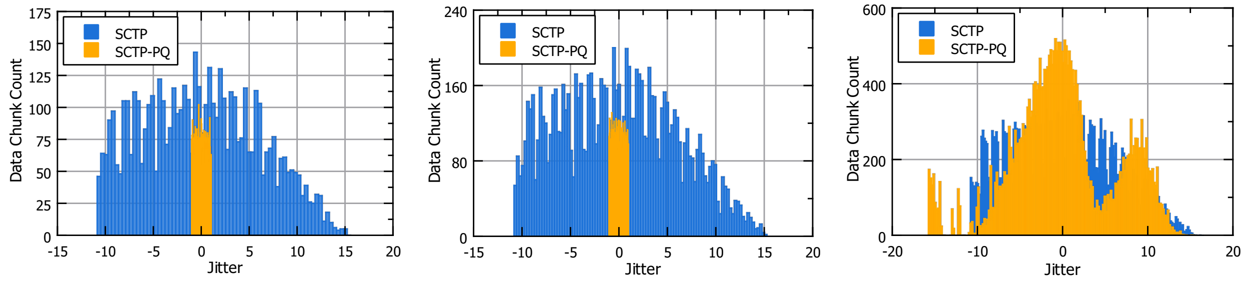
IPDV values for data chunks of all priority levels are shown in Figures 8a, 8b and 8c.

As can be seen in Figures 8a, 8b and 8c, SCTP-PQ has a more stable and much lower IPDV than standard SCTP for the first and second priority level data chunks. In the simulations, the average IPDV for SCTP for the highest priority data chunks is about 8 ms. The same value is about 1 ms for SCTP-PQ. Also, SCTP has a lower IPDV than SCTP-PQ for the third priority level data chunks. This is caused due to the insertion of the third priority level data chunks to the end of the buffer.

Jitter values for data chunks of all priority levels are shown in Figures 9a, 9b and 9c. Minimising the delay times of the packets leads to performance improvement, and the distribution of jitter values around zero indicates that the communication quality has improved. As a result, the more packets with jitter values close to zero, the smoother the communication. For SCTP-PQ, the jitter is around 0 for almost all data packets. However, for SCTP, the jitter value ranged from -10 to +15.

As shown in Figures 9a, 9b and 9c, SCTP-PQ has more stable and much lower jitter than SCTP for the first and the second priority level data chunks. Also, SCTP has less jitter than SCTP-PQ for the third priority level data chunks. This result is caused due to the insertion of the third priority level data chunks to the end of the buffer. The fourth level, namely Level-3 priority is the highest priority level. Retransmitted chunks have priority level four. Because of higher priority chunks insertion to the head of the queue, the queue delay of these chunks is very short.

Finally, the total queue delay and the number of



(a). First priority level.

(b). Second priority level.

(c). Third priority level.

Figure 9. Jitter values for different priority level data chunks.

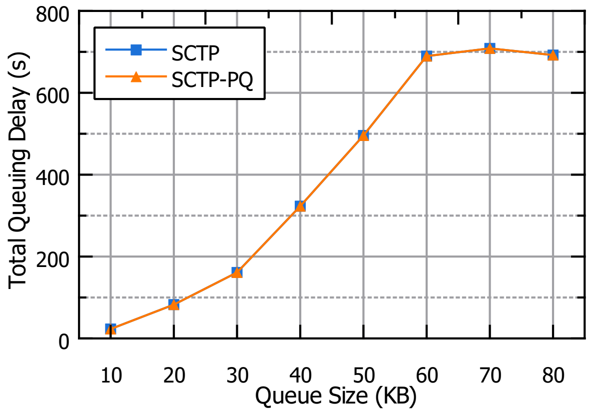


Figure 10. Total queuing delay for all data chunks.

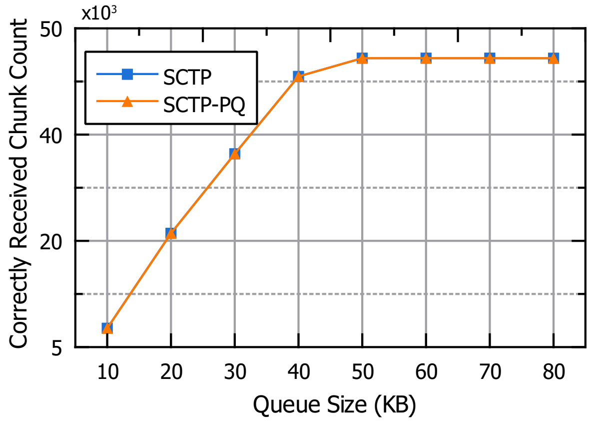


Figure 11. Number of correctly received data chunks.

correctly received data chunks have been obtained to evaluate the correctness of the simulations. These values are nearly the same for SCTP-PQ and SCTP. The total queue delay and the number of correctly received data chunks are shown in Figures 10 and 11, respectively. The following equation is used to find the total delay values (T) in the queue.

$$T = \sum_{k=1}^{n} T_k \times E_k \,, \tag{3}$$

where n is the number of priority levels, T_k is the average queuing delay for level-k, E_k is the number of data chunks in the queue for level-k.

Since the numbers of received data chunks in SCTP-PQ and SCTP simulations are the same, the lines overlap in Figures 10 and 11.

As can be seen in Figures 10 and 11, the values are very close to each other.

5. Conclusion

In this study, four-level priority queue integrated SCTP-PQ has been proposed to reduce delay and jitter for real-time multimedia streaming for mobile devices. The payload data chunks have been re-arranged in the receiver side buffer according to priority levels. The experimental results show that SCTP-PQ has reduced the average queuing delay by a factor of ten, IPDV value by a factor of eight, and jitter value by a factor of ten for high priority chunks.

Because mobile devices have limited memory and processing capacity, the SCTP-PQ is designed to be very simple. However, it significantly improves the video streaming quality. Due to mobile links being very unstable, disconnections and slowdowns occur frequently, and negatively affect the quality of the video stream. SCTP-PQ assigns the highest priority to the retransmitted chunks, improving the video streaming quality in networks with poor connection quality.

References

[1] H. Schulzrinne, S. Casner, R. Frederick, V. Jacobson. RFC3550: RTP: A transport protocol for real-time applications, 2003. https://doi.org/10.17487/rfc3550

[2] S. Fu, M. Atiquzzaman. SCTP: State of the art in research, products, and technical challenges. *IEEE Communications Magazine* **42**(4):64–76, 2004. https://doi.org/10.1109/MCOM.2004.1284931

[3] K. Ahmad, A. C. Begen. IPTV and video networks in the 2015 timeframe: The evolution to medianets. *IEEE Communications Magazine* **47**(12):68–74, 2009. https://doi.org/10.1109/MCOM.2009.5350371

[4] Z. Wang, S. Mao, W. Yang. Deep learning approach to multimedia traffic classification based on QoS

characteristics. *IET Networks* **8**(3):145–154, 2018. https://doi.org/10.1049/iet-net.2018.5179

[5] C.-H. Ke, N. Chilamkurti. A new framework for MPEG video delivery over heterogeneous networks. *Computer Communications* **31**(11):2656–2668, 2008. https://doi.org/10.1016/j.comcom.2008.02.029

[6] S. Zeadally, H. Moustafa, F. Siddiqui. Internet protocol television (IPTV): Architecture, trends, and challenges. *IEEE Systems Journal* **5**(4):518–527, 2011. https://doi.org/10.1109/JSYST.2011.2165601

[7] S. Kumar, S. Rai. Survey on transport layer protocols: TCP & UDP. *International Journal of Computer Applications* **46**(7):20–25, 2012.

[8] C. Holmberg, S. Hakansson, G. Eriksson. Web real-time communication use cases and requirements. *Request for Comments (RFC)* **7478**, 2015. https://doi.org/10.17487/rfc7478

[9] E. A. Kaur, E. P. Rani. A review on comparative analysis of different transport layer protocols in MANETs. *International Journal Of Engineering and Computer Science* **4**(06), 2015.

[10] C. Li, S. He, Z. Fen. Reliability analysis of communication IP network. In *2016 International Conference on Robots & Intelligent System (ICRIS)*, pp. 282–284. IEEE, 2016. https://doi.org/10.1109/ICRIS.2016.58

[11] H. T. Alvestrand. Overview: Real-Time Protocols for Browser-Based Applications. RFC 8825, 2021. https://doi.org/10.17487/RFC8825

[12] W. A. Hussein, S. F. Lu. Performance comparison of transport layer protocols for multimedia application in wired networks. *IOSR Journal of Computer Engineering* **18**(6):33–38, 2018.

[13] S. T. Kim, S. J. Koh, Y. J. Kim. Performance of SCTP for IPTV applications. In *The 9th International Conference on Advanced Communication Technology*, vol. 3, pp. 2176–2180. IEEE, 2007. https://doi.org/10.1109/ICACT.2007.358804

[14] T. Bach, M. Maruschke, J. Zimmermann, et al. Combination of IMS-based IPTV services with webRTC. In *Proceedings of the 9th Internation Multi-Conference on Computing in the Global Information Technology, Seville, Spain*, pp. 140–144. 2014.

[15] A. Abd, T. Saadawi, M. Lee, et al. Unequal error protection for real-time video in mobile ad hoc networks via multi-path transport. *Computer Communications* **30**(17):3293–3306, 2007. https://doi.org/10.1016/j.comcom.2007.08.042

[16] R. Stewart, M. Tüxen, P. Lei. SCTP: What is it, and how to use it? In *Proceedings of BSDCan: The Technical BSD Conference*, pp. 1–10. 2008.

[17] R. K. Goyal, S. Kaushal. A survey of mSCTP for transport layer mobility management. *Journal of Advances in Information Technology* **4**(1):20–27, 2013. https://doi.org/10.4304/jait.4.1.20-27

[18] R. Stewart, Q. Xie, K. Morneault, et al. RFC2960: Stream control transmission protocol, 2000. https://doi.org/10.17487/rfc2960

[19] J. Ortiz, E. M. Graciá, A. F. Skarmeta. SCTP as scalable video coding transport. *EURASIP Journal on Advances in Signal Processing* **2013**(1):115, 2013. https://doi.org/10.1186/1687-6180-2013-115

[20] Ł. Budzisz, J. Garcia, A. Brunstrom, R. Ferrús. A taxonomy and survey of SCTP research. *ACM Computing Surveys (CSUR)* **44**(4):1–36, 2012. https://doi.org/10.1145/2333112.2333113

[21] M. Ahmad, M. Hussain, B. Abbas, et al. End-to-end loss based TCP congestion control mechanism as a secured communication technology for smart healthcare enterprises. *IEEE Access* **6**:11641–11656, 2018. https://doi.org/10.1109/ACCESS.2018.2802841

[22] I. Rüngeler, M. Tüxen, E. P. Rathgeb. Congestion and flow control in the context of the message-oriented protocol SCTP. In *International Conference on Research in Networking*, pp. 468–481. Springer, 2009. https://doi.org/10.1007/978-3-642-01399-7_37

[23] A. Jagetiya, R. Challa. Stream control transmission protocol. *The Journal of the Computer Society of India* **40**:33–35, 2016.

[24] M. N. El Derini, A. A. Elshikh. MPEG-4 video transfer with SCTP-friendly rate control. In *The Second International Conference on Innovations in Information Technology IIT'05*, pp. 1–11. Citeseer, 2005.

[25] S. Boussen, N. Tabbane, S. Tabbane. Performance analysis of SCTP protocol in WiFi network. In *2009 First International Conference on Communications and Networking*, pp. 1–5. IEEE, 2009. https://doi.org/10.1109/ICCIT.2009.30

[26] Y. Wang, I. Rhee, S. Ha. Augment SCTP multi-streaming with pluggable scheduling. In *2011 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, pp. 810–815. IEEE, 2011. https://doi.org/10.1109/INFCOMW.2011.5928924

[27] T. D. Wallace, A. Shami. A review of multihoming issues using the stream control transmission protocol. *IEEE Communications Surveys & Tutorials* **2**(14):565–578, 2012. https://doi.org/10.1109/SURV.2011.051111.00096

[28] R. Rajput, G. Singh. Comparing stream control and datagram congestion control with traditional transmission control protocol. *International Journal of Computer Science and Mobile Computing (IJCSMC)* **4**(6):570–577, 2015.

[29] J. Eklund. *Latency Reduction for Soft Real-Time Traffic using SCTP Multihoming*. Ph.D. thesis, Karlstad University Press, 2016.

[30] L. Wang, K. Kawanishi, Y. Onozato. Achieving robust fairness of SCTP extension for MPEG-4 streaming. In *2009 IEEE 20th International Symposium on Personal, Indoor and Mobile Radio Communications*, pp. 2970–2974. IEEE, 2009. https://doi.org/10.1109/PIMRC.2009.5450179

[31] J. Filip. Analytical model of modified traffic control in an ATM computer network. *Acta Polytechnica* **41**(6):45–50, 2001. https://doi.org/10.14311/288

[32] R. Stewart, M. Tüxen, S. Loreto, R. Seggelmann. Stream schedulers and user message interleaving for the stream control transmission protocol, 2017. https://doi.org/10.17487/RFC8260

[33] A. Argyriou. A novel end-to-end architecture for H.264 video streaming over the internet. *Telecommunication Systems* **28**(2):133–150, 2005. https://doi.org/10.1007/s11235-004-5013-1

[34] R. Fracchia, C. Casetti, C.-F. Chiasserini, M. Meo. A WiSE extension of SCTP for wireless networks. In *IEEE International Conference on Communications, 2005. ICC 2005. 2005*, vol. 3, pp. 1448–1453. IEEE, 2005.

[35] K.-H. Kim, K.-M. Jeong, C.-H. Kang, S.-J. Seok. A transmission control SCTP for real-time multimedia streaming. *Computer Networks* **54**(9):1418–1425, 2010. https://doi.org/10.1016/j.comnet.2009.11.014

[36] S. Lederer, C. Müller, C. Timmerer. Dynamic adaptive streaming over HTTP dataset. In *Proceedings of the 3rd multimedia systems conference*, pp. 89–94. 2012. https://doi.org/10.1145/2155555.2155570

[37] S. S. Kallungal. A survey on adaptive video streaming techniques with cloud. *International Journal of Advanced Research in Computer Engineering & Technology (IJARCET)* **6**(3):350–352, 2017.

[38] R. Stewart, M. Ramalho, Q. Xie, et al. Rfc3758: Stream control transmission protocol (sctp) partial reliability extension, 2004.

[39] P. P. D. Amer, M. Becke, T. Dreibholz, et al. Load sharing for the stream control transmission protocol (SCTP). Internet-Draft draft-tuexen-tsvwg-sctp-multipath-25, Internet Engineering Task Force, 2023. Work in Progress.

[40] L. P. Verma, V. K. Sharma, M. Kumar, et al. DB-CMT: A new concurrent multi-path stream control transport protocol. *Journal of Network and Systems Management* **30**(4):67, 2022. https://doi.org/10.1007/s10922-022-09677-1

[41] P. Tomar, G. Kumar, L. P. Verma, et al. CMT-SCTP and MPTCP multipath transport protocols: A comprehensive review. *Electronics* **11**(15):2384, 2022. https://doi.org/10.3390/electronics11152384

[42] L. P. Verma, V. K. Sharma, M. Kumar. New delay-based fast retransmission policy for CMT-SCTP. *International Journal of Intelligent Systems and Applications* **10**(3):59–66, 2018. https://doi.org/10.5815/ijisa.2018.03.07

[43] N. Arianpoo, V. C. Leung. A smart fairness mechanism for concurrent multipath transfer in SCTP over wireless multi-hop networks. *Ad Hoc Networks* **55**:40–49, 2017. https://doi.org/10.1016/j.adhoc.2016.11.005

[44] C. Yu, W. Quan, D. Gao, et al. Reliable cybertwin-driven concurrent multipath transfer with deep reinforcement learning. *IEEE Internet of Things Journal* **8**(22):16207–16218, 2021. https://doi.org/10.1109/JIOT.2021.3101447

[45] D. Wang, M. Wang, T. Zhang, et al. An adaptive CMT-SCTP scheme: A reinforcement learning approach. *Journal of Networking and Network Applications* **1**(4):170–178, 2022. https://doi.org/10.33969/J-NaNA.2021.010404

[46] I. Sodagar. The MPEG-DASH standard for multimedia streaming over the internet. *IEEE multimedia* **18**(4):62–67, 2011. https://doi.org/10.1109/MMUL.2011.71

[47] N.-N. Dao, A.-T. Tran, N. H. Tu, et al. A contemporary survey on live video streaming from a computation-driven perspective. *ACM Computing Surveys* **54**(10s):1–38, 2022. https://doi.org/10.1145/3519552

[48] H. L. Mohamed, A. F. Mahmood. A survey of streaming protocols for video transmission. *NTU Journal of Engineering and Technology* **2**(1):23–38, 2023. https://doi.org/10.56286/ntujet.v2i1.391

[49] G. N. Vivekananda, C. Reddy, A. Ilknur. A congestion avoidance mechanism in multimedia transmission over MANET using SCTP multi-streaming. *Multimedia Tools and Applications* **79**:16823–16844, 2020. https://doi.org/10.1007/s11042-019-7260-x

[50] K. Fall, K. Varadhan. The ns manual (formerly ns notes and documentation). 2011, [2023-11-04], http://www.isi.edu/nsnam/ns/doc/ns_doc.pdf.