

Real-Time Protocols for Communication and Collaboration Environments in Telemedical Applications

Sadiq ur Rehman¹, Syeda Bushra Ahmed² and Muhammad Hasnain Raza³

¹Department of Electrical Engineering, Hamdard University, Karachi, Pakistan

²Department of Anatomy, Hamdard University, Karachi, Pakistan

³Department of Electronics Engineering, Hamdard University, Karachi, Pakistan

Correspondence Author: Sadiq ur Rehman (sadiq.rehman@hamdard.edu.pk)

Received April 01, 2021; Revised August 02, 2021; Accepted August 06, 2021

Abstract

The world is now moving in the era of high information and communication technology. Under the term telemedicine and tele-cooperation research series are carried out presently, intended to enable and expand a distributed, group-related cooperative work within the area of medicine. For this purpose, applications are needed, which support real-time control processes, distributed applications, communication and cooperation technology, and the representation of shared information and data. Real-time protocols are the basis of telemedical applications. They are needed in almost all fields of telemedicine reaching from communication transmission to reliable data transfer. In a telemedical system, how these real-time protocols are used for communications, collaboration with telemedicine applications is worth to be explored and has been discussed in this research article along with the challenges of real-time applications and their solutions by using real-time protocols.

Index Terms: Telemedical Applications, Real-Time Protocols, Robot Control, File Sharing, Instruction Via Chat.

I. INTRODUCTION

Nowadays, a lot of researches are going on to expand a distributed, group-related cooperative work within the area of medicine, supported by Information and Communication Technologies (ICT). Under the title of telemedicine, different applications are required which can support real-time control processes, distributed applications, communication, and cooperation technology, and the representation of shared information and data. For this purpose, different application-level real-time protocols are to be used for telemedical applications [1-3].

Suitable real-time protocols are investigated in this article that can be used in the telemedical applications required in the most common scenario of telemedicine as shown in figure 1. Where there is an experiment/operation going on in the near-end with the help of different telemedicine applications and devices. Two assistants are present in the near end to assist experts who are in the far end/s. Experts can monitor (EGC pulse, blood pressure, and other sensor information), guide and control the experiment i.e., this entire transmission is in real-time. Different types of delivery options (exchange of information) are possible during the experiment. Exchange of information can be in either real-time or store-and-forward communication [4] and [5].

Examples of real-time telemedical transactions include live audio/video conference, the examination of the

patient, auscultation of heart and lung sound, and reports examination, i.e., X-ray, Computed Tomography (CT) [6], Magnetic Resonance Imaging (MRI) [7], Ultrasound, etc. In the case of exchanging information in real-time, one must be careful about response time, bandwidth, data rate, delay, jitter, error and loss rate, reliability, etc.

In contrast with the real-time method, in the store-and-forward method, a set of medical information is compiled and sent to the experts for their review and interpretation as per their convenience. Examples of medical information for this method can be paper Electrocardiogram (ECG), the combination of still image frames, audio/video clips, and other related stuff [8] and [9].

In telemedicine, the quality of audio/video data and vital signal data has high importance, based on these data, doctors can make medical decisions. Lack of important information due to any reason may lead to erroneous decisions which can badly affect the overall outcome of a certain telemedical event.

To prevent this, one should always keep in mind the networks through which telemedicine applications transmit medical information (audio, video, and data). These networks must fulfill the minimum requirements (bandwidth, data rate, etc.) for the telemedical applications so that minimum transmission error/problems can be observed.

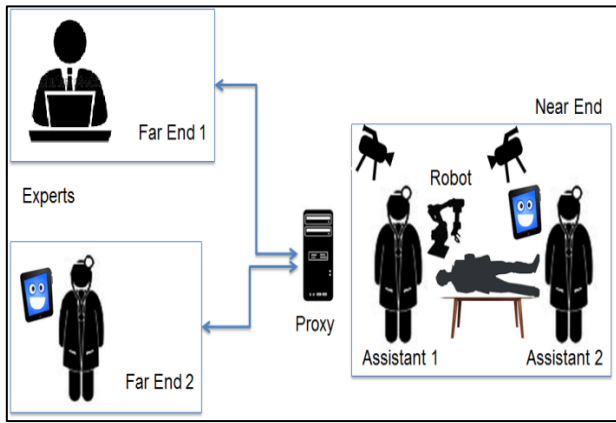


Figure 1: Near Field and Far-Field Operation

II. APPLICATION REQUIREMENT

Telemedicine applications required for a successful operation/experiment in the mentioned scenario are depicted in figure 1. Some of the aspects are given below:

A. Video/Audio

In Audio/Video (A/V), the camera and microphone are the most important devices. The feature of audio/video is very time-critical for Real-Time (RT) transmission and reception. To have a smooth and error-free transmission/reception, real-time issues, such as types of data to be transmitted (for audio/video), control over the delivery of data, correct ordering in reception, delay, jitter, synchronization, packet loss, recording, encoding/compression schemes, bandwidth issue, network problem, etc. should be minimized as much as possible.

To analyze the operation later, there will also be an option of recording so that both parties can discuss or evaluate the operation when needed or make evidence of any specific task performed in the ongoing operation.

B. Object Data (e.g., Medical Records)

Medical records like X-ray reports, patient history, blood test reports, etc. are not time-critical. However, an error-free transaction is needed. The expected issue, in this case, can be the size of data, packet loss, network problems, etc.

C. Real-Time Object Data (e.g., ECG, Sensors)

This type of data object falls in the category of real-time data, which is off-course time-critical data and a reliable connection is required. Sending real-time information to the expert about the heart rate of patients, blood pressure, and other sensor information about operated parts of the patient's body must be delivered in time without any delay and error. Possible issues for the real-time object data can be; types of data to be transmitted (audio/video), delay, jitter, synchronization, packet loss, recording, encoding/compression schemes, bandwidth issue, network problems, etc.

D. Robot Control

It is very crucial that while operating (especially in surgery) perfect cutting of skin or veins can take place. It requires accurate calculation and measurement. If this operation is performed by a human, then a big possibility of human error is there which may affect the life of the

patient. To overcome this, the need of telemedicine application that controls the robot arm is required which can perform accurate cutting or other measurements required by the operation without error/s. This robotic arm is very time-critical as it cannot afford to have a delay in instruction or any problem in a network through which it is being operated [10] and [11].

E. Instructions Via Chat

Telemedicine applications that can facilitate near and far-end sites to communicate through messages are also required in this research scenario. Here timing is not so critical, however, there are still some issues that can cause problems in the successful running of these applications, some of these issues are packet loss, encoding/compression schemes, network problems, connection reestablishment, Internet Protocol mobility, etc.

III. REAL-TIME PROTOCOLS

A. User Datagram Protocol (UDP)

To make applications in use, suitable protocols are required which are used to work on the application layer. In the case of telemedicine, we need real-time protocols that are the basis of telemedical applications. In this section, there will be a focus on only those real-time protocols that can fulfill the requirements of a given telemedicine scenario see figure 1.

In telemedical applications, User Datagram Protocol (UDP) is very well suited where quick responses/queries are required [12] and [13]. The UDP is a connection-less protocol. Data transmission is unreliable in this service it is because there will be no acknowledgment segments to the user when data is received or any information to the sender when data is lost. This will affect the integrity of data which can be suffered due to the drop of packets. As there is no error detection in this protocol, packets can be miss-sequenced. When UDP is to be used in telemedicine, it is the responsibility of the application layer to arrange to acknowledge message receipt; the proper ordering of received packets which make it into meaningful messages, get rid of duplicate packets and requesting retransmission of faulty packets. Some of the key advantages of UDP include small packet size, high speed, no retransmission delay, etc.

B. Transmission Control Protocol (TCP)

Transmission Control Protocol (TCP) is a transport layer protocol [14]. In telemedicine, this protocol is commonly used by the application that requires guaranteed end-to-end delivery of data. In other words, telemedical applications that are relatively less time-critical used to prefer TCP. This protocol is connection-oriented and creates a connection between sender and receiver and keeps it maintained until the completion of exchanging data. Data transmission in this protocol is performed with the feature of error detection/correction. This protocol also offers "flow control" whose task is to determine the need for data re-transmission, and stops the flow of data until all previous packets are successfully transferred. Key advantages of TCP include reliability, congestion control, full-duplex, etc.

C. Stream Control Transport Protocol (SCTP)

The Stream Control Transmission Protocol (SCTP) is a transport protocol, which is present at an equivalent level with UDP (User Datagram Protocol) and TCP (Transmission Control Protocol), it provides transport layer functions to many internet applications. Compare to UDP and TCP, SCTP provides services and features that are more suitable for real-time protocols, especially in the telemedicine field. However, SCTP has a big drawback, that some infrastructure on the internet takes the header as defective and drops the packet. So, it is only usable in small area networks with controlled infrastructure. The main features of SCTP include multi-homing and multi-streaming [15-17].

D. Real-Time Transport Protocol (RTP)

In telemedicine, audio and video streaming is a key feature and requires transmission to be performed in real-time. For this reason, Real-time Transport Protocol (RTP) is needed to support telemedicine applications to transport information in real-time [18]. RTP is an internet-based protocol that provides end-to-end transport service for real-time data such as audio and video stream over multicast or unicast network services [19]. RTP fulfills the need for interactive services (e.g., Internet Telephony) and media on demand (video) in telemedicine. RTP is an application layer protocol which used to run over UDP. Since RTP uses UDP, data delivery is fast. However, there is no guaranteed, Quality of Service (QoS). Furthermore, there is no reservation of bandwidth in RTP. It is the responsibility of the Real-Time Control Protocol (RTCP) to offer QoS. RTP is best suited for telemedical applications that are less sensitive to packet loss and very sensitive to packet delay. Telemedicine multimedia applications need correct timing and data playback which can be fulfilled by using RTP.

RTP takes care of timing issues by providing time-stamping and featuring sequence numbers. Therefore, the most important parameters of RTP for the telemedical scenario are time stamping, sequence numbering, payload type identifier, and source identification.

E. Real-Time Control Protocol (RTCP)

Data transmission over RTP is unreliable as it isn't playing any role in providing mechanisms for flow control, error control, quality feedback, synchronization, and congestion control which is very important for telemedical applications. To overcome these problems, Real-Time Control Protocol (RTCP) has been associated with RTP to provide reliability, quality of service, data delivery, and end-to-end monitoring [22] and [23]. RTCP is used by many applications to change sender transmission rates and also for diagnostic purposes. RTCP is best suited for telemedical applications that are sensitive to packet loss and less sensitive to packet delay. RTCP is used to monitor QoS, congestion control, session size estimation, scaling, media synchronization, and source identification. Key tasks of RTCP parameters are bandwidth-scaling, delay, jitter: packet loss, feedback, media synchronization, etc.

F. Real-Time Streaming Protocol (RTSP)

In telemedicine, reliable, controlled, on-demand delivery of real-time data is required. In telemedical systems, applications for the support of audio/video data transmissions are needed, the source of data can be live data feeds or stored clips. These applications contain features for conferencing and recording which helps to make reviews when needed and many more. To facilitate these telemedical applications, an application-level protocol named Real-Time Streaming Protocol (RTSP) is to be used.

Table 1: Comparison between Real-Time Protocols

Features	RTP	RTCP	RTSP	RTMP	RTMFP	XMPP
Transmission Type	Unicast	Multicast	Multicast	Multicast	Multicast	Multicast
Open-source	Yes	Yes	No	No	Yes	Yes
Delivery methods	UDP	UDP	UDP/TCP	TCP	UDP	TCP
Main theme	Transport	Controlling	On-Demand Access	Client-Server Communication	Client To Client Communication	Client-Server Communication
Application protocol	Yes	Yes	Yes	Yes	Yes	Yes
Reliable	No	Yes	Yes	Yes	No	Yes
Audio Support	Yes	Yes	Yes	Yes	Yes	Yes
Video Support	Yes	Yes	Yes	Yes	Yes	Yes
Data Support	Yes	Yes	Yes	Yes	Yes	Yes
HTTP support	No	No	Yes	Yes	Yes	Yes
Security	Yes	Yes	Yes	Yes	Yes	Yes
Latency	Low	Average	Low	Low	Very Low	Low
Overhead	Low	Low	Low	Low	Very Low	Very Low
Speed of delivery	Fast	Slow	Normal	Slow	Fast	Fast

RTSP is responsible to control rewinding playback, fast-forwarding, pausing, playback, repositioning playback to future or past time just like Video Cassette Recorders (VCR). The main operation of RTSP includes conference participation, live streaming, interoperability, multi and unicast, controlled delivery, playback, and recording, etc. [20] and [21].

G. Real-Time Messaging Protocol (RTMP)

As in telemedical multimedia applications, support for 'Adobe Flash' is needed which helps in adding streamed video and audio players and interactive multimedia content on web pages. This is done with the help of one real-time protocol commonly known as Real-Time Message Protocol (RTMP) [24]. RTMP is a TCP-based protocol designed to facilitate client-to-server-to-client

communications. Examples of data that can be served by using RTMP can be text chat, pre-recorded data, live video data, live audio data, and others.

The most important responsibility of RTMP is to specify different virtual channels on which receiving and sending of packets can take place. These channels should be independent of each other and may get active simultaneously at any given time.

H. Real-Time Media Flow Protocol (RTMFP)

Real-Time Media Flow Protocol (RTMFP) was introduced by Adobe [25]. This protocol is an application-level protocol and works on the principle of UDP. Hence, reduced latency and direct peer-to-peer connections are the key advantages of RTMFP. Table 1 represents the comparison between Real-Time Protocols to be used in the scenario of figure 1.

I. Extensible Messaging and Presence Protocol (XMPP)

Extensible Messaging and Presence Protocol (XMPP) is a communications protocol for message-oriented middleware. It is an open technology for real-time communication which provides the real-time equivalent of HTTP and is used in many diverse applications [26] and [27]. XMPP supports data transmission for both real-time and store-and-forward medical applications. By using XMPP, applications for the transmission of audio/video stream, data (medical records, ECG, sensors), instant messaging, robotic control and system control, and access to shared file space can be developed. The main advantages of XMPP for telemedical applications include its stability, interoperability, security, extensibility, decentralizations, scalability, equivalency with HTTP, HTML, and the other Internet, and many more.

IV. PROTOTYPE MODEL

A prototype model has been suggested to support different functions/features and data transmissions that are required in the telemedical scenario. It is to be noted that the suggested prototype model can only be used for testing purposes, which means it may or may not cover all the features/functions that are required in the given telemedical scenario presented in figure 1.

The prototype model will run as simulation software that is capable of using different systems to perform transmission and reception of data files (real-time data or simple data), instant messages, audio/video streams, and robot control. Applications to run the following systems are based on different real-time protocols (RTP, RTCP, RTSP, RTMFP, and XMPP). This simulation software needs to be installed on both sides (specialists and assistants). With the help of this simulation system which is named "UDE Telemedical Simulation System", one can analyze, visualize, specifies and records the behavior of telemedical applications.

As "UDE Telemedical Simulation System" is consisting of different applications/systems that are to be used for the given telemedical scenario. This means, by using this single simulating system, one can play with the real-time protocols and parameters of selected features to see under which conditions we can obtain the best results.

The "UDE Telemedical Simulation System" in figure 2 will work in such a way that applications that are going to be used can be controlled from a graphical user interface (GUI) of this system after passing through the security and admin checks [28].

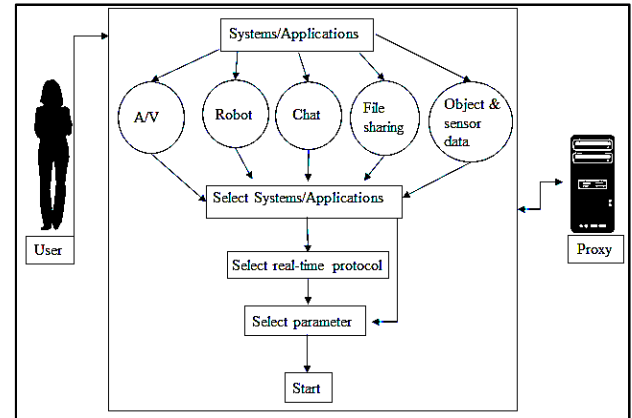


Figure 2: UDE Telemedical Simulation System

Moreover, in GUI there will be options for selecting features to be used, the available real-time protocol for those features, and altering parameters of selected features. All applications are connecting users with the devices place at near or far ends. For example, in the telemedicine case of our research, if the user (assistant) wants to send an X-ray report to another user (specialist) then just by simply clicking the browse/file button in the GUI (where the application that supports file sharing feature is running), a request will be generated at the back-end to look for the required X-ray report in the database system.

If the required file is not present in the data-base, then the request will move forward and the device (scanning machine for example) will be activated to get the X-ray report so that it can be sent to the specialist. Similarly, for the case with A/V, the user needs to simply click buttons in the GUI of the "UDE Telemedical Simulation System" to control microphone and camera options (for example zoom in/out, mute, play, stop, etc.). There will be an option in UDE Telemedical Simulation System to switch to other applications at any time so that the application with the highest priority can be used easily.

For the case of robot control, it is very much important to shift robot control between two expert doctors (if required). Therefore, if the option of robot control is selected in GUI, then there will a separate window for this feature in which the user is asked to give the user ID and password and also to type in access control information by filling "To" and "From" form. The user then needs to click apply button to activate the shifting of the robot control.

Data delivery will be based on UDP or TCP (depending on application usage) but not on SCTP as SCTP has a big drawback, that some infrastructure on the internet takes the header as defective and drops the packet. SCTP is only usable in small area networks with controlled infrastructure and we want this system/software to work not only for small area networks but also for wide area networks.

The "UDE Telemedical Simulation System" is consisting of different applications/systems that are to be used for the given telemedical scenario. In a system where there are transmission and reception of audio or video (could be A/V conferencing), devices like camera and microphones are needed to be connected on both the ends (near-end and far-end).

The feature of audio/video is very time-critical for real-time transmission and reception. The audio/video system is responsible for the successful delivery of real-time packets/streams over the internet by using protocols like XMPP, RTP/RTCP, etc. Using RTSP, audio/video stream data can be controlled. RTSP behaves just like a Video Cassette Recorder (VCR). All these protocols are based on UDP or TCP.

One can individually test the behavior of this audio/video system with the help of some useful tools (D-ITG [29], Wireshark [30], ping, etc.) to measure bandwidth, delay, jitter, data rate, etc.

Based on the different traffic classes, different QoS can be taken as a reference to compare with actual testing values. In the case of interactive audio-on-demand, approx. 128 kbps bandwidth is required for the stereo quality audio stream with the recommended delay of less than 150 ms. Furthermore, a jitter rate of less than 100 ms and a coding standard of MPEG-1 is recommended.

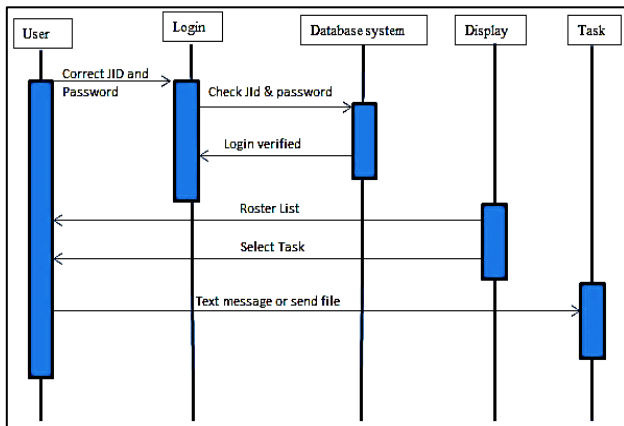


Figure 3: Transmitting Part of Proposed XMPP Application for Messages and File Sharing

If video conferencing is under test, the following are the recommendations, lip-synchronization < 100 ms, delay < 150 ms, jitter < 400 ms, loss rate and error rate < 0.01 % each. There is a possibility to use different coding standards which required different data rates and bandwidths. For example, H.320 requires bandwidth from the range of 80 Kbps to 2 Mbps and supports data rate from 64 bps to 1920 Kbps.

Different systems/applications are needed to test instructions via chat and for sending files for data (real-time data or simple data), as now in such a case we require applications that are more sensitive to packet loss and less sensitive for packet delay. Data transmission for the delivery of instant messages or for sending files must be reliable which can be achieved by using TCP.

Real-time protocols such as RTMP and XMPP are used to send sensor data, instructions via chat, access shared file

space, and for transmission of the data object (real-time or store-and-forward). However, it is suggested to use XMPP for such cases as it is an open standard protocol and provides a real-time equivalent of HTTP. XMPP also provides stability, interoperability, security scalability, presence service, and many more.

The functionality of an application based on XMPP (for instructions via chat and for sending files) just for the testing purpose needs to be divided into two parts; one part will be used for sending messages and files, while the other part mainly used for receiving messages and files which can be seen in figure 3 and figure 4. Application-based on XMPP for features like instructions via chat, accessing shared file space, etc. can also be tested individually to check its performance by using some useful tools like Iperf, D-ITG, Wireshark, ping, etc. The figure 5 represents the architecture diagram of the "UDE Telemedical Simulation System".

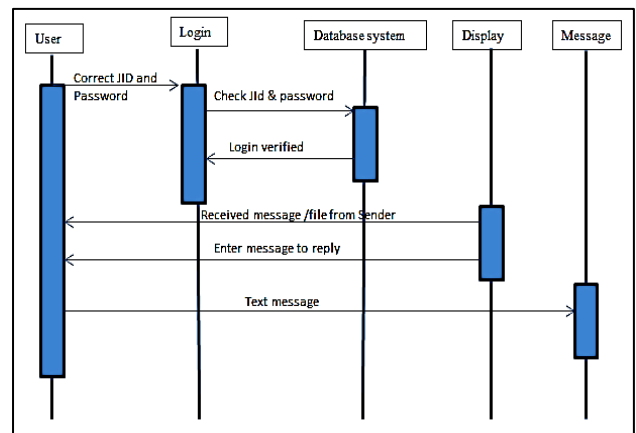


Figure 4: Receiving Part of Proposed XMPP Application for Messages and File Sharing

In figure 5, any type of data (real-time or non-real-time) will be used by this UDE Telemedical System for simulation purposes, components of this simulation system are texting units that will perform its operation when there is a need for instant messaging. Audio, video, and display/graphic unit, position control unit for robot control applications, synchronization unit will be useful when multiple features are running at the same time with the support of timer that record time of performed activity and make logs as per requirement. All of these units are controlled by a data handling/record unit which behaves like a controlling unit of this simulation system. Data will be transferred with UDP/IP or TCP/IP socket over the network. The Context diagram of the UDE Telemedical Simulation System can be seen in figure 6.

To get a deeper understanding of how this simulation system should work so that different functions and data transmission can be tested between two or more computers, we need to focus on figure 7 and figure 8.

According to figure 7 and figure 8, the user (Expert or the assistant) needs to open UDE Telemedical Simulating software on the computer, after successful opening, there will be a prompt window asking to select a model which can be the expert mode or assistant mode. If the user selects any of these modes, he then needs to provide login

details for that model. Login details will be verified from a database system and after a successful login, the user (in either mode) will get the main window on the screen in which he can select feature/s to be tested. All the features

like video/audio, instruction via chat, sending data object (real-time or non-real-time) files, and robot control (only available in expert mode) will be displayed in this main window see figure 9.

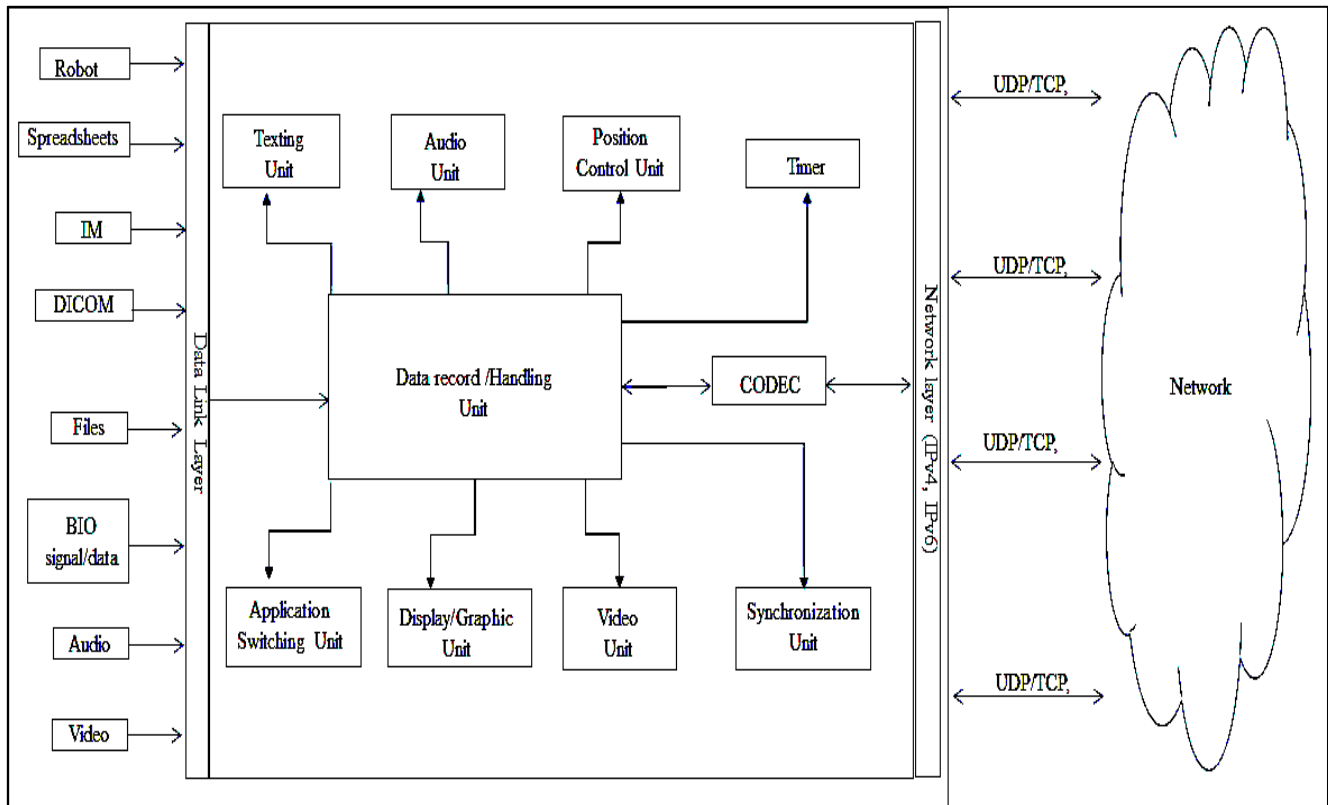


Figure 5: Architecture Diagram of UDE Telemedical Simulation System

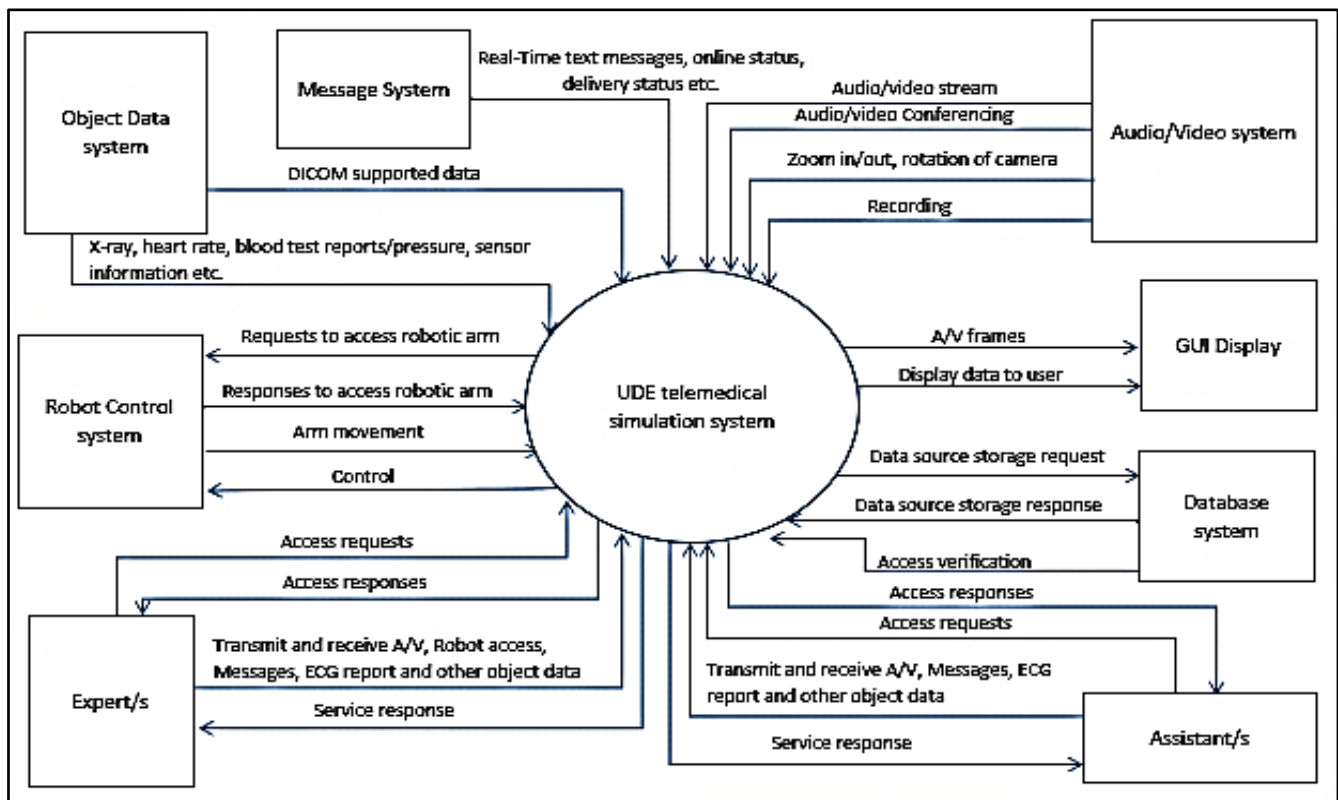


Figure 6: Context Diagram of UDE Telemedical Simulation System

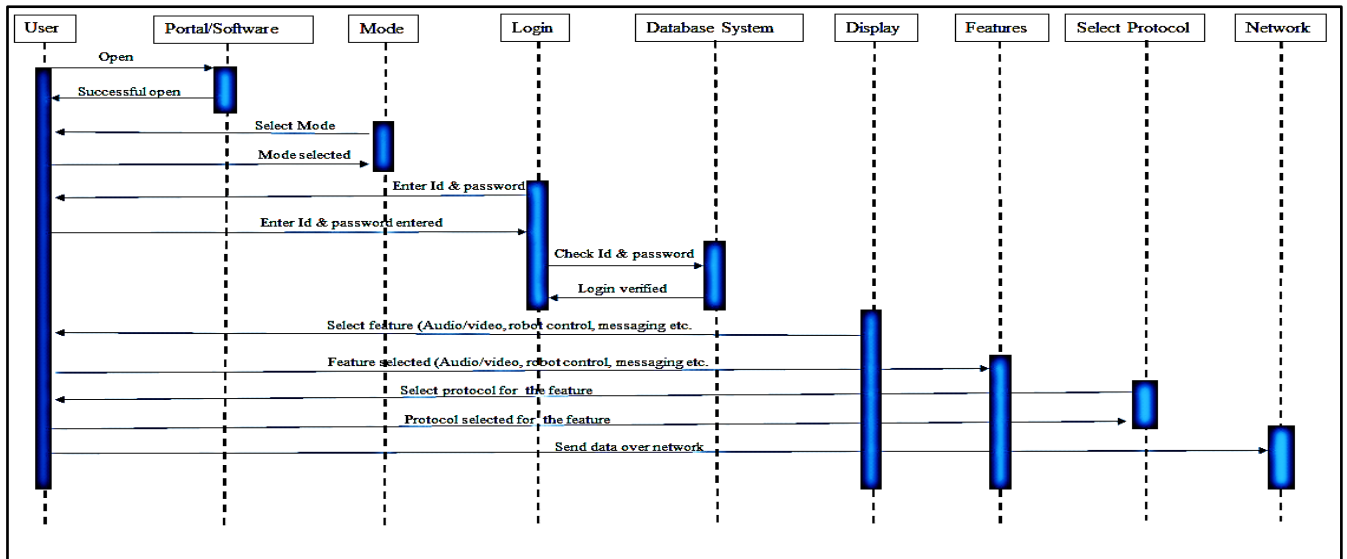


Figure 7: Sequence Diagram of UDE Telemedical Simulation System

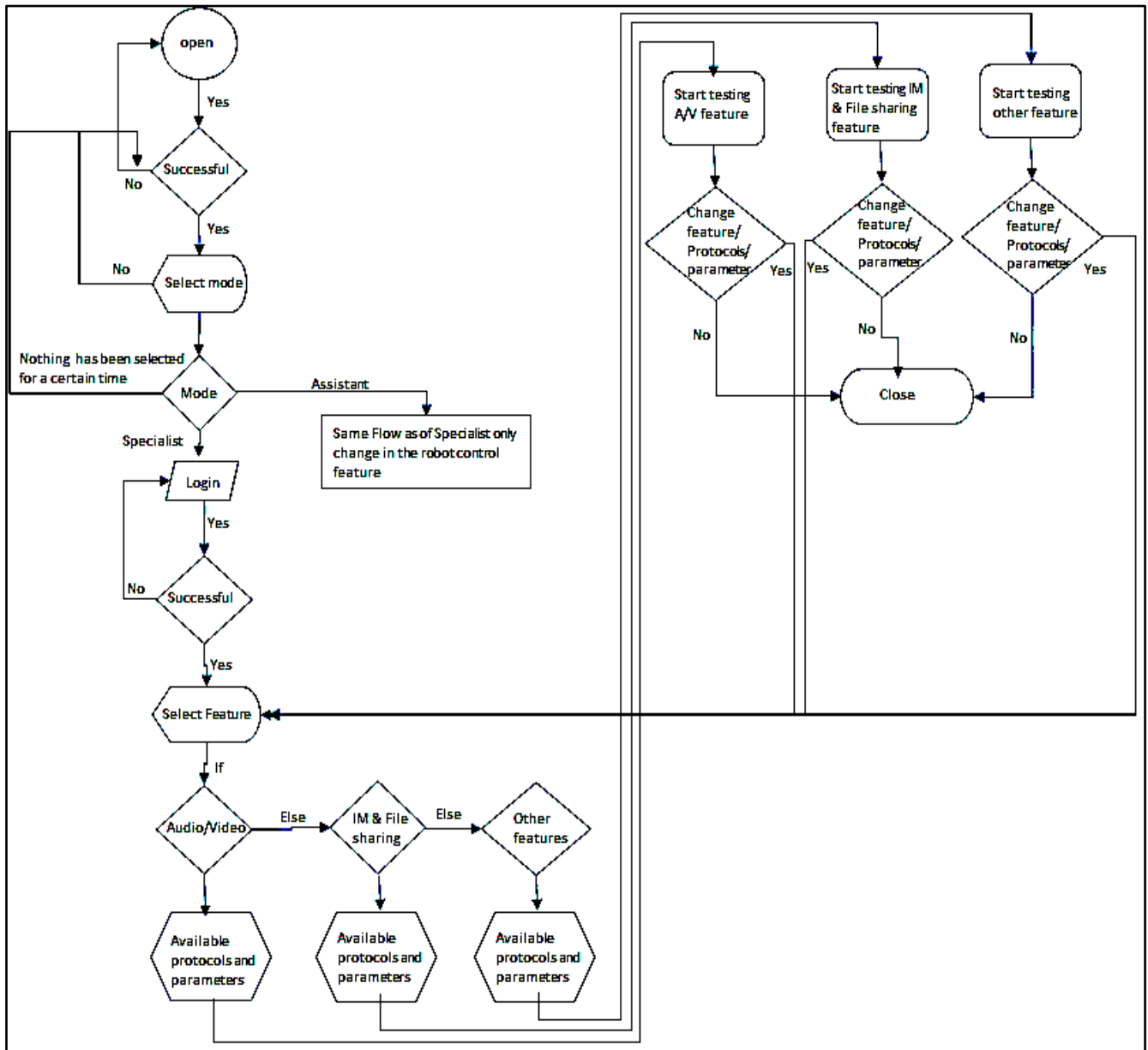


Figure 8: Data Flow Diagram of UDE Telemedical Simulation System

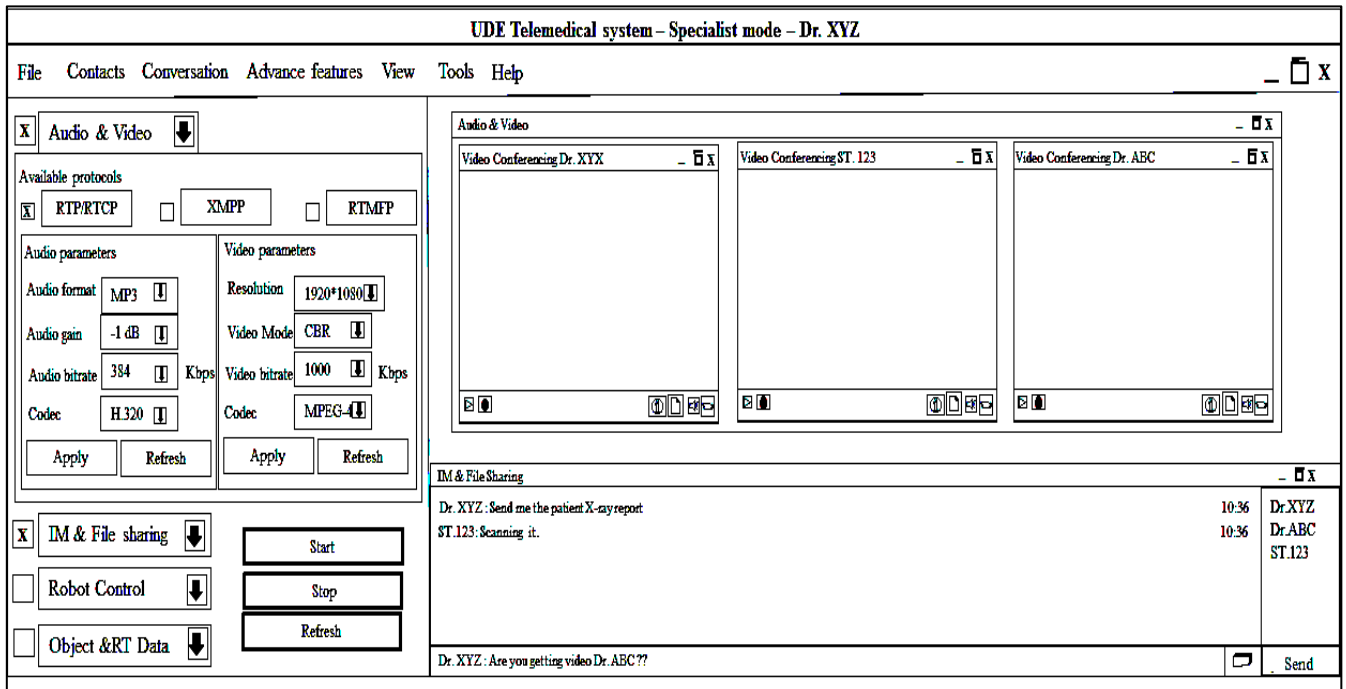


Figure 9: GUI of UDE Telemedical Simulation System

After the selection of a particular feature, the user can then select a real-time protocol for that specific feature just by clicking the pull-down button. Here at this moment, all the valid/available real-time protocols will be displayed in a small window, moreover, the user can also manually set/select other parameters for the selected feature. There will also be an option for the user to switch to any other feature (for example was using/testing A/V and now Instant messaging and file sharing is needed) if required or use different features simultaneously. Send data now over the network to start testing its parameters. Different tools can be used to measuring parameters and outcomes some of which include but are not limited to Distributed Internet Traffic Generator (D-ITG), Ethereal/ Wireshark, Available bandwidth estimator, Ping, and Traceroute.

V. CONCLUSION AND FUTURE WORK

The purpose of this research was to find suitable real-time protocols that can be used for the telemedical applications and are required in the given research scenario, for this reason, brief discussion and examination has been performed on the most important protocols that are required in a given scenario, comparison table has also been provided in table 1 which gives detail information about the offered services/features of each real-time protocol.

When performing a comprehensive comparison among different transport protocols (UDP, TCP, and SCTP), it has been noted that SCTP is the best-suited transport layer protocol as it provides services and features that are more suitable for real-time protocols, especially in the telemedicine field. However, SCTP has a big drawback, that some infrastructure on the internet takes the header as defective and drops the packet; therefore, SCTP is only usable in small area networks with controlled

infrastructure. Hence using SCTP is not a good choice and we need to use either UDP or TCP for data delivery.

All the real-time protocols that have been discussed in this research are based on either UDP or TCP. Based on the analysis and supporting features of different real-time protocols, different telemedical applications for different purposes need to be developed.

RTP/RTCP is suggested to be the best suited real-time protocol for telemedical applications (like robot control, audio/video conferencing, video streaming, etc.) that are less sensitive to packet loss and very sensitive for packet delay. The use of RTSP will be like adding VCR functionality to an application. When there is a requirement of applications (ECG, X-rays, MRI data, CT, Ultrasound, text instruction via chat, file sharing, etc.) that are more sensitive to packet loss and less sensitive for packet delay, then in such case XMPP is suggested to be used. RTMP (TCP-based) and RTMFP (UDP-based) these protocols are both used to add interactive multimedia content on web pages and require an Adobe flash player.

A prototype model (named "UDE Telemedical Simulation System") has been proposed in this research with the objective that this prototype model should enable different functions/features that are required in telemedicine scenarios and data transmissions between two computer systems. UDE Telemedical Simulation System will work as a simulation system/software in which users can play with the selection of different real-time protocols and parameters of selected applications and features under which conditions the best results can obtain from figure 9. To examine the important parameters like delay, jitter, bandwidth, etc. of different telemedical applications/systems used in the "UDE Telemedical Simulation System", some important tools like Iperf, Wireshark, available bandwidth estimator, etc. are to be used.

As future work, UDE Telemedical Simulation System needs to be implemented; the effective implementation and success of a UDE Telemedical Simulation System depend on the seamless and continuous communication transfer and its clear interaction between the various ends. Possible approaches and methods to minimize and overcome the real-time network problems to achieve maximum QoS are worth to be investigated. Techniques for reliable data connections, robust continuous transmission systems, and fully utilize the available resources of a transmission channel by enhancing the quality of service are also need to be investigated.

Acknowledgment

I would like to give my special thanks to my supervisor, Pascal A. Klein, a very resourceful and responsible researcher, who has supported me throughout my research work. I wish to thanks the University Duisburg-Essen and Prof. Dr.-Ing. Axel Hunger, to provide me an opportunity to explore the area that I was interested in.

Authors Contribution

Sadiq ur Rehman- conceived the presented idea, worked on overview, analysis, and comparison of real-time protocols with the design and development of testing scenarios and prototype model. Syeda Bushra Ahmed- wrote the manuscript as per the SSURJET Template (including re-draw of UML diagrams) in consultation with Sadiq Ur Rehman. Muhammad Hasnain Raza- proofread, checked the grammar, and approved the final manuscript with the incorporation of the latest references.

Conflict of Interest

There is no conflict of interest between all the authors.

Data Availability Statement

All data generated or analyzed during this study are included in this published article.

Funding

This research received no external funding.

References

- [1] Dorsey, E. R., & Topol, E. J. (2020). Telemedicine 2020 and the next decade. *The Lancet*, 395(10227), 859.
- [2] Calton, B., Abedini, N., & Fratkin, M. (2020). Telemedicine in the time of coronavirus. *Journal of Pain and Symptom Management*, 60(1), e12-e14.
- [3] Gorbach, V., Ali, M. L., & Thakur, K. (2020, September). A Review of Data Privacy Techniques for Wireless Body Area Networks in Telemedicine. In *2020 IEEE International IoT, Electronics and Mechatronics Conference (IEMTRONICS)* (pp. 1-6). IEEE.
- [4] Brahneborg, D., Afzal, W., Čaušević, A., & Björkman, M. (2018, July). Towards a more reliable store-and-forward protocol for mobile text messages. In *Proceedings of the 2018 Workshop on Advanced Tools, Programming Languages, and PLaforms for Implementing and Evaluating Algorithms for Distributed Systems* (pp. 13-19).
- [5] López Seguí, F., Franch Parella, J., Gironès García, X., Mendioroz Peña, J., García Cuyàs, F., Adroher Mas, C., ... & Vidal-Alaball, J. (2020). A cost-minimization analysis of a medical record-based, store and forward and provider-to-provider telemedicine compared to usual care in Catalonia: more agile and efficient, especially for users. *International journal of environmental research and public health*, 17(6), 2008.
- [6] Sera, T. (2020). Computed tomography. In *Transparency in Biology* (pp. 167-18). Springer.
- [7] Eisenberg, H. M., Shenton, M. E., Pasternak, O., Simard, J. M., Okonkwo, D. O., Aldrich, C., ... & Hayman, E. G. (2020). Magnetic resonance imaging pilot study of intravenous glyburide in traumatic brain injury. *Journal of neurotrauma*, 37(1), 185-193.
- [8] Poonguzhali, S., & Chakravarthi, R. (2020). Performance improvisation using IOT based sensor network in telemedicine assistance by electrocardiogram signal analysis for diabetes healthcare. *Biomedicine*, 40(1), 83-88.
- [9] Saeed, S. (2019). A Conceptual System on Ubiquitous Cardiovascular Health-Care System (UCHS). *Sir Syed University Research Journal of Engineering & Technology*, 9(1).
- [10] Wang, L. (2019). Optimize 2-D Path Planning of Mobile Robot by using ACO with Guidance Factor. *Sir Syed University Research Journal of Engineering & Technology*, 9(02).
- [11] kiran Bollineni, R., Menon, S. S., & Udupa, G. (2020). Design of Rover and Robotic Arm. *Materials Today: Proceedings*, 24, 1340-1347.
- [12] Aro, T. O., Adekeye, S. A., Akande, H. B., & Funso, B. O. (2019). Application of Transmission Control Protocol (TCP) and User Datagram Protocol (UDP) in Teleconference System. *Journal of Computer Science and Control Systems*, 12(2), 15-19.
- [13] Vedavathi, T., Karthick, R., Selvan, R. S., & Meenalochini, P. (2020). Data Communication and Networking Concepts in User Datagram Protocol (UDP). *International Journal of Recent Technology and Engineering (IJRTE)*, vol 8, pp. 2765-2768.
- [14] Veluswami, J. R. S., Chinnusamy, K., Kumar, K., Klinge, V. C., & Mohankumar, S. (2021). Improvement of Transmission Control Protocol for High Bandwidth Applications. *Wireless Personal Communications*, 117(4), 3359-3379.
- [15] Vivekananda, G. N., & Ilknur, A. (2020). A congestion avoidance mechanism in multimedia transmission over MANET using SCTP multi-streaming. *Multimedia Tools and Applications*, 79(23), 16823-16844.
- [16] Hercog, D. (2020). SCTP Protocol. In *Communication Protocols* (pp. 341-342). Springer, Cham.
- [17] Weinrank, F. (2021). *SCTP as an Universal Multiplexing Layer* (Doctoral dissertation, University of Duisburg-Essen, Germany).
- [18] Kumar, A. (2020). Performance Evaluation of Video Streaming Traffic in Data Centre Servers Using Real-Time Transport Protocol (RTP). *International Journal of Emerging Technology and Innovative Engineering*, vol 6, pp. 472-477.
- [19] Costin, A., & Zinca, D. (2020, November). Extending the libtins library with SIP and RTP classes. In *2020 International Symposium on Electronics and Telecommunications (ISETC)* (pp. 1-4). IEEE.
- [20] Hercog, D. (2020). Protocols RTP/RTCP. In *Communication Protocols* (pp. 343-344). Springer, Cham.
- [21] Sarker, Z., Perkins, C., Singh, V., & Ramalho, M. (2021). RTP Control Protocol (RTCP) Feedback for Congestion Control. *Internet RFC*, (8888).
- [22] Chiu, J. C., Tseng, H. Y., & Lee, Z. Y. (2020, December). Design of Multidimension-media Streaming Protocol Based on RTSP. In *2020 International Computer Symposium (ICS)* (pp. 341-347). IEEE.
- [23] Syaifudin, Y. W., Rozi, I. F., Ariyanto, R., Rohadi, E., & Adhisuwarnjo, S. (2018). Study of Performance of Real Time Streaming Protocol (RTSP) in Learning Systems. *International Journal of Engineering & Technology*, 7(4.36), 1441-1446.
- [24] Nurrohman, A., & Abdurrohman, M. (2018, May). High Performance Streaming Based on H264 and Real Time Messaging Protocol (RTMP). In *2018 6th International Conference on Information and Communication Technology (ICICT)* (pp. 174-177). IEEE.
- [25] Han, Z., Ma, J., He, X., & Fan, W. (2019). A Statistic and Analysis of Access Pattern for Online VoD Multimedia. *Journal of Signal Processing Systems*, 91(10), 1149-1157.
- [26] Nikolov, N. (2020, September). Research of MQTT, CoAP, HTTP and XMPP IoT Communication protocols for Embedded Systems.

- In 2020 XXIX International Scientific Conference Electronics (ET) (pp. 1-4). IEEE.
- [27] Chaudhari, D. A., & Umamaheswari, E. (2020). Comparative Analysis of IoT Application Layer Protocols for Healthcare Data Transmission System. In *First Doctoral Symposium on Natural Computing Research: DSNCR* (p. 313).
- [28] Oulasvirta, A., Dayama, N. R., Shiripour, M., John, M., & Karrenbauer, A. (2020). Combinatorial optimization of graphical user interface designs. *Proceedings of the IEEE*, 108(3), 434-464.
- [29] Chang, C. H., Lin, Y. D., Lai, Y. K., & Lai, Y. C. (2019). A scalable and accurate distributed traffic generator with Fourier transformed distribution over multiple commodity platforms. *Journal of Network and Computer Applications*, 144, 102-117.
- [30] Fathima, K. M., & Santhiyakumari, N. (2021, March). A Survey On Network Packet Inspection And ARP Poisoning Using Wireshark And Ettercap. In *2021 International Conference on Artificial Intelligence and Smart Systems (ICAIS)* (pp. 1136-1141). IEEE.