Evaluating Transport and Application Layer Protocols for Haptic Applications

G. Kokkonis, K. E. Psannis, M. Roumeliotis, S. Kontogiannis
Department of Technology Management,
University of Macedonia,
Loggou-Tourpali Area, Naousa 59200, Greece.
E-mail: {gkokkonis, kpsannis, manos} @uom.gr

Y. Ishibashi
Department of Scientific and Engineering Simulation,
Nagoya Institute of Technology,
Nagoya 466-8555, Japan.
E-mail: ishibasi@nitech.ac.jp

***Abstract*—Recently, a number of transport and application layer protocols for haptic and Interactive applications have been proposed. This paper presents a survey of the above protocols. It outlines the haptic data transmission characteristics. It also depicts the qualitative features that transport and application layer protocols should contain in order to carry haptic data. Based on these criteria, evaluation of the above protocols is performed.**

Keywords-Tele-Haptics, Collaborative Haptics, Tactile feedback, Transport protocols, Teleoperation, Interative applications

I. INTRODUCTION

High Quality of Experience (QoE) is an asset that all applications try to gain. The integration of haptic features in an application can highly improve its QoE. Studies [1, 2] have shown that haptic applications are sensitive to network conditions such as network delay, jitter, packet loss and out-of-order packet delivery. Due to this sensitivity, haptics could not be applied over the Internet until recently. Recent improvements in network conditions made it feasible for new services, known as tele-haptics and collaborative haptics (C-haptics), to flourish. Tele-haptics and C-haptics are a branch of interactive application that tries to carry the sense of touch and the tactile feedback through the Internet.

There are two main categories of haptic architecture [3]: The server based and the distributed architectures. The client-server architecture has the advantage of consistency. On the other hand, a peer-to-peer architecture supports parallel computation and scalability, and is less dependent on network conditions. The scene in a server-based architecture is only updated after a round-trip delay to the server while in peer-to-peer architecture the update delay is only one-way. Knowing that haptic applications are vulnerable to network delays, in collaborative applications with many participants, a peer-to-peer architecture is often chosen.

The main obstacle to the growth of haptics is the Internet's unstable network conditions. The protocols that specialize in haptic data transmission are SCTP [4], SMOOTHED-SCTP [5], IRTP [6], RTNP [7], ETP [8], and RTP/I [9]. Some of these protocols, such as IRTP and RTNP were not created specifically for haptic applications but for interactive applications in general. Both haptic and interactive applications present the same requirements for data transmission, so that these protocols can be applied to both applications. Many methods have been investigated to optimize haptics.

The rest of this paper is organized as follows. In section II, haptic data transmission characteristics and data flow requirements are described. Section III gives an evaluation of haptic transport protocols and related work, while section IV describes the requirements for haptic transport protocols. Finally, performance metrics for haptic data transmission protocols are presented in section V, while section VI identifies future research directions.

II. HAPTIC DATA TRANSMISSION CHARACTERISTICS

Research [10, 11] has shown that the update rate, in this kind of applications, must be quite high, close to 1 kHz. This nontrivial update rate is needed especially in cases where users come in contact with each other, like in the case of collaborative haptics, or with other objects. The harder the objects are, the higher the update rate should be, so as to avoid the phenomenon of unwanted intrusion and oscillation [12]. The data load per data unit is relatively small, for a motion command is about 32 bytes [6]. If the overhead of the datagram is added, then the size of the data unit is in most cases around 40 bytes, for the UDP protocol. But for update rate equal to 1 kHz the throughput is computable, most often greater than 320 kbps. Haptic system channel throughput requirements, data update rate frequencies and maximum transfer units are summarized in Table I.

Based on Table I the requirements for the control and the feedback channel are exactly the same. It should also be mentioned that the refresh rate of the video data (30 fps) is much smaller than that of the haptic data (1 kHz). Of course, the maximum transmission unit (MTU) of haptic data (64-128 Bytes) is 10 to 20 times smaller that of the video data (1500 Bytes). The throughput of a haptic system (512-1024 kbps) is a little bit bigger than that of an audio system (128 kbps) and smaller than that of a video system.

TABLE I. HAPTIC SYSTEM TRANSPORT DATA REQUIREMENTS

Haptic System Requirements	Requirements for Each Channel		
	Transmission Control Data	*Reception Feedback Data*	*MPEG4 Video Data*
Update Frequency	1 kHz	1 kHz	27-30 fps Resolution: 320x240 or 640x360 px
Throughput:	512-1024 kbps	512-1024 kbps	Video: 900-2200 Kbps Audio: 128-160 Kbps
Packet Size:	64-128 Bytes	64-128 Bytes	1500 Bytes

A. Haptic data flow requirements

Haptic data, as all real-time multimedia streaming applications, require timely delivery of information and can tolerate some packet loss to achieve this goal. Moreover, haptic applications are prone to instability. This instability can be caused by the Packet Delay Variation (PDV), also called as jitter [13]. The delay of a network specifies how long it takes for a packet to travel across the network from one node or endpoint to another. Research [14, 15, 16, 17, 18] has shown that the maximum delay should be less than 50 ms, the jitter less than 10 ms and the packet loss less than 10%. When delay and jitter get values higher than the above, then malfunctions such as unpleasant sensation of handling, delayed response, oscillations and instability occur. To avoid all these unwanted malfunctions, network conditions should satisfy table II.

TABLE II. HAPTIC DATA FLOW REQUIREMENTS

QoS	Requirements for Different Applications			
	Haptics	*Video*	*Audio*	*Graphics*
Jitter (ms)	10	30	30	30
Delay (ms)	60	400	150	100-300
Data Rate (kbps)	512	2500 - 40000	22-200	45-1200
Data Loss (%)	0,1-10	1	1	10
Update Rate (Hz)	1000	30	8000	30

It could be said that compared to the requirements that other multimedia applications have, haptic applications are more tolerant to data loss and bandwidth but are sensitive to delay, jitter and update rate.

III. RELATED WORK ON HAPTIC TRANSPORT PROTOCOLS

Several protocols have been proposed to transfer haptic data. Some of them were created purely for this purpose while others were designed for other interactive applications such as robot tele-operation [19]. Based on the qualitative features that are being presented in section IV and Table III that depicts which qualitative features every protocol fulfills, there has been a qualitative evaluation of these protocols. They are presented below with a decreasing order of suitability for real–time haptic communication:

ALPHAN [20]: It is an Application Layer Protocol for Haptic Networking (ALPHAN). It operates on top of UDP and it can easily be customized with the help of an XML-based description file. It supports prioritization with the help of multiple buffers. It uses three types of messages, partially borrowed from MPEG. I packets, known as "key updates" that are sent reliably and P and B packets, that are sent unreliably, known as "normal updates". It includes a timestamp and a sequence number to its packets. It also includes a participant and an object ID. All the above properties force the protocol to use 16 bytes of overhead at every packet.

S-SCTP [5]: Smoothed - Synchronous Collaboration Transport Protocol (s-SCTP) is widely used in haptic applications. It derived from the SCTP [4] but differs in that it includes a buffer at the receiver’s side to reduce the unwanted effect of jitter. It uses two types of packets: the “key updates” which are sent reliably and the “normal updates” that are sent unreliably. In order to avoid the implosion problem it applies negative acknowledgement. It performs congestion control by scaling the transmission rate, depending on the received Ack and Nack messages. It uses interaction streams to carry “differential” packets. Key updates are placed especially in the first and in the last slot of the stream.

HMTP [21]: It is an efficient Hybrid Multicast Transport Protocol (HMTP) for collaborative virtual environments. It derives from the combination of four protocols: Scalable Reliable Multicast (SRM) [22], Reliable Multicast Transport Protocol (RMTP) [23], Selective Reliable Transmission Protocol (SRTP) [24] and Synchronous Collaboration Transport Protocol (SCTP). It enforces scalability like RMTP and reliability with NACK approach like SRM. Its architecture resembles a multicast tree, like RMTP. It has two types of messages, “key updates” and “normal updates” in an interaction stream, like SCTP. The packets are sent with three modes of transport depending on their type like SRTP. This protocol is focusing on collaborative applications with a lot of users in which case multicast is necessary.

IRTP [6]: Another protocol that applies to interactive applications is the Interactive Real-time Protocol (IRTP). It is connection oriented and is located at the transport layer. For its implementation it imitates the transport protocols TCP and UDP, the TCP for the transport of “crucial data” and the UDP for the transport of the “remaining data”. It implements flow and congestion control with the windows size scheme, same as TCP, and error control with the help of a buffer at the sender's side and two buffers at the receiver's side. The main advantage of this protocol is that it uses a very little overhead of only 9 bytes.

RTP/I [9]: The Real Time Application Level Protocol For Distributed Interactive Media (RTP/I) is also a protocol that can be used for haptic applications. It also uses UDP for the “unreliable data” and TCP for the “reliable data”. It contains a participant identifier, a priority field, a sequence number and a timestamp in its packets. It has four types of messages: “event”, “state”, “delta state” and “state query” messages. Its main drawback is that it is not designed purely for haptic application but for general interactive applications. This means that it has a big overhead of 28 bytes at every datagram in order to support a wide spectrum of applications. Taking into account that the data rate in haptic applications is near 1 kHz, a 224 kbps bandwidth is required only for the overhead information, which is mostly unnecessary.

ETP [8]: The Efficient Transport Protocol (ETP) is designed especially for haptic applications. Its main target is to optimize the available bandwidth by trying to minimize the Inter Packet Gap (IPG) and the round trip time (RTT). For this goal it has six states: FAST DECREASE IPG, LOOK, INCREASE IPG, SLOW DECREASE IPG, STABILITY IPG and STABILITY MAX. By adjusting the IPG it performs congestion control. It uses the UDP protocol for transporting its data and it operates the control and the feedback channel as two independent data flows. The protocol that has almost the same features as ETP is the Bidirectional Transport Protocol (BTP) [25] which is specialized in tele-operated robots and has been developed from the same authors.

UDP: This protocol is in transport layer and provides the smallest delay, if no congestion occurs in the network. It does not provide reliability and congestion control and it is very poor to Inter-Protocol and Intra-Protocol Fairness. It does not use buffer and produces large jitter. It creates the smallest overhead, 8 bytes, and it is characterized as a “best effort” protocol.

RTNP [7]: Another protocol for haptic data transmission is the Real Time Network Protocol (RTNP). It was designed for teleoperation but its main drawback is that it can be implemented only in unix environments. Its main feature is that it uses priority for its packet. The scale of the priority is designated inside the packets. The priority changes the queue in which the packets stand at the sender's or in an intermediate buffer. If a packet is marked as a “real-time” packet, it gets higher priority.

STRON [26]: The Supermedia Transport for Teleoperations over Overlay Networks (STRON) is an transport scheme that uses forward error correction encoding to provide a reliable and fast transmission service. The main attribute of STRON is that it takes advantage of multiple disjoint overlay network paths (one of them is the IP path) to transport its packets. With the use of Reed-Solomon codes provides a rather reliable transport service without using acknowledgments and retransmissions. For congestion control it enforce the TCP Friendly Rate Control (TFRC) [27].

TCP: This is a protocol that creates a virtual connection between sender and receiver. It is a reliable protocol with a congestion control mechanism. It creates bigger delay than UDP and it is TCP-friendly with other connections. It uses a bandwidth optimization control with a window size scheme. It creates very big overhead, 20 bytes, that is a disadvantage at a very high update rate. As TCP is optimized for accurate delivery rather than timely delivery, it is understood that it should not be applied in haptic applications.

A summary of the properties of all the above protocols is presented in Table III.

IV. HAPTIC SYSTEM TRANSPORT PROTOCOL REQUIREMENTS

Using TCP as transport protocol for haptic systems is difficult to meet the requirements of Table II. In order to implement so big update frequencies (1 kHz) with so little delay (60 ms) and jitter (<10 ms), protocols should meet certain constrains and contain some qualitative features. Table III presents all the protocols that could be used in a haptic transmission system. It also depicts the qualitative features that a protocol should fulfill in order to meet requirements of Table II. These qualitative features are presented below in a decreasing order of importance:

Prioritization [28]: Haptic transport protocols should give higher priority to real-time interactive data. If different kinds of data (haptic, video and voice) are transmitted simultaneously, priority should be given to haptic data as they are more sensitive to delay and jitter. As haptic data contain control commands, they are more crucial than video and voice. Video and audio have lower requirements on delay and jitter which means that they should have lower priority. This prioritization can be supported with the IPv4 ToS octet or the IPv6 traffic class octet in a DiffServ architecture.

Key Updates [4]: Many interactive protocols divide packets into key updates which are sent reliably and normal updates which are sent unreliably with best effort techniques. Furthermore, some protocols, like SCTP, support "differential messages". They are update messages that carries only the difference between the current and the previous state of an entity's. This technique achieves higher compression and faster data transmission.

Interaction Stream [4]: In an interaction stream a series of update messages are grouped into one stream. All the messages from the same stream are referred to the same interaction. With this grouping “differential messages” can be applied, higher compression can be achieved, less bandwidth can be required and less computational power for packet forwarding is needed.

Minimum Overhead: Haptic protocols have to keep as minimum overhead as possible. Haptics requires very high update rate, around 1 kHz. This means that each byte at the header is being multiplied by a factor of 1000 due to the update rate. The result of this multiplication is the required bandwidth. Since the packet payload is quite small, usually 32-128 bytes, each byte of overhead sensibly reduces protocol efficiency [6].

Error Correction and Data Integrity: Reliability is a feature that must be supported from a haptic protocol. A small part of the messages, such as the last update message, of the control and feedback channel should be sent reliably. A proposed mechanism that enforces reliability and reduces network traffic is the Negative Acknowledgement (NACK) [29] and the Selective Acknowledgment (SACK) [30]. Both of the above mechanisms avoid the Acknowledgement “Implosion problem”. Nevertheless both mechanisms increase protocol overhead. A tradeoff threshold should be set in order to satisfy minimum overhead requirement, error correction and data integrity.

Jitter Control [31] **and Congestion Control** [32, 33]: Jitter control can avoid long delay and jitter due to network overload. When the network is under congestion, data transfer should be reduced or even terminated in order to solve congestion and consequently reduce delay and jitter. Algorithms that could be applied for the congestion control are the Additive Increase/Multiplicative Decrease AIMD [34], the Rate Based Congestion Control RAP [35] and the TCP Friendly Rate Control (TFRC) [27].

Real-time Flow Control: In an IP network, packets can be lost, duplicated, or delivered out of order. A sequence number and a buffer, at the receiver's side will help to set these packets in the correct order. Moreover, the sequence number helps in understanding if any packet has been dropped so that a retransmission can be required. Furthermore, each packet should carry a timestamp. The resolution of the clock must be sufficient for the desired synchronization accuracy. It must allow more precise RTT and jitter estimation. When a packet delays, it means that the network is congested.

Synchronization [36]: As mentioned before, different kinds of data are being transmitted simultaneously over the same channel. Video, voice and haptic media streams have to be synchronized in order maximum QoE to be achieved. For this synchronization buffer schemes, timestamp fields and time adjustment algorithms are used. It is worth pointing out that media synchronization control is one of key techniques for realizing distributed multimedia applications. Media synchronization control falls into three types: intra-stream, inter-stream, and group (or inter-destination) synchronization control. Group synchronization control as well as the first two types of control is needed in multicast communications. The purpose of the control is to output each MU of media streams simultaneously at different destinations for the fairness among the destinations [37], [38], [39] and [40].

Multicast [21]: In the case of a multiuser-collaboration application, the protocol should support Multicast. This property will save users from unnecessary retransmission of identical information from the same source to different recipients. Multicast results in a better utilization of network resources and avoids network congestion. The multicast property can be supported through IP Multicast routing technique.

Bandwidth Estimation [41]: The protocol should reduce the transfer rate if it detects network overload so as to avoid congestion. In the opposite case, however, if network resources are free for use, it should increase the transfer rate in order to fulfill transmission requirements. Methods for determining an appropriate sending rate of the transport protocol can be found in [42], [43] and [44].

Multiplexing [14]: In a haptic application different kinds of data, such as voice, video, graphics and haptic data, are being transmitted simultaneously. Each kind of data has different Quality of Service (QoS) requirements. The statistical multiplexing of this data can achieve better bandwidth utilization.

Scalability- Adaptability [45]: Since network conditions change quite often, the protocol should have the ability to adapt the throughput of the transfer data to the network conditions. When conditions allow it, the protocol should send more detailed information. When a network is heading to congestion, the protocol should have the ability to lower the transferred data either by lowering the data rate or by omitting some packets-informations that are not really necessary.

Receiver Buffer Optimization [36]: Many protocols use buffers to reduce the unwanted effects of jitter and the out of order arrival of packets. With the help of an intermediate buffer the receiver can use packets at the right time regardless of the time of origin. The important drawback introduced by the buffer is that it increases the mean time of the delay. This means that an appropriate receiver buffer length threshold should be applied.

Minimum TCP Friendly Capability: As tele-haptics refer to the Internet, haptic protocols should try to be TCP friendly [46, 47]. There should be a tradeoff between fair network resource allocation and bandwidth optimization so transmission requirements are met.

V. HAPTIC TRANSPORT PROTOCOL PERFORMANCE METRICS

To evaluate the behaviour of haptic protocols, performance metrics should be attained as well. Measures of the performance of a haptic transport protocol should rely on tangible attributes.

Performance metrics used in order to evaluate performance of haptic transport protocols are similar to metrics used for TCP and UDP transport protocols. That is, throughput or goodput at the receiver end, packet loss, network delay, and packet delay variation also called as jitter. As already mentioned, jitter is more crucial for a haptic system (HS) as it causes system instability. Throughput or goodput are less significant since the media unit of a HS is rather small. Haptic protocols should avoid packet delays with big variation in order to function properly, even if this means to operate in a higher RTT or less throughput environment.

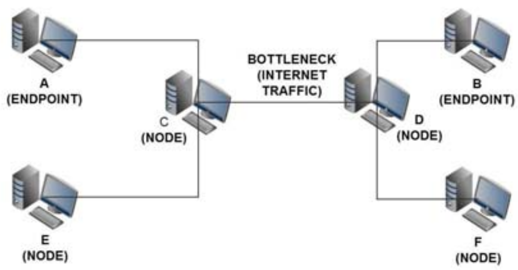

Figure 1. Simulation Scenario.

In order to evaluate each protocol based on the above tangible attributes, a simulation scenario must be set. Some open source, popular network simulators that offer this ability are ns2, ns3 and omnet++. The simulation scenario must contain at least, two endpoints (A, B) that exchange haptic data, some intermediate nodes (C, D) that forward the passing information and some other nodes (E, F) that communicate with each other based on protocols as TCP and UTP as depicted in Figure 1. Intermediate nodes C, D will be the bottleneck of the network as they represent the Internet

Network. Nodes E and F will generate the Internet traffic. The simulator tool will record all the important performance metrics as delay, jitter, packet loss and throughput. After running the same simulation scenario with all haptic protocols of section III, the evaluation of protocols will be completed.

VI. CONCLUSIONS AND FUTURE WORK

It is worth pointing out that the research area of haptic protocols is fairly new and open to research and development. The QoS for transferring haptic data is particularly demanding and challenging in order to achieve the maximum QoE. The protocols that are used for haptic transmission are continuously evolving. This paper analyzes data transmission characteristics and the data flow requirements of haptic data. It presents the protocols that are used for the transmission of haptic data and analyzes the qualitative features that these protocols should fulfill. Table III summarizes all the above information.

The next step is to compare the above protocols in a simulation environment. From the simulation, it will be possible to derive accurate statistical and qualitative data that will allow in-depth evaluation of the reported protocols. Taking the above information in consideration, it is possible to either suggest corrections to the relevant protocols or create a new haptic protocol.

TABLE III. HAPTIC DATA FLOW REQUIREMENTS AND TRANSPORT PROTOCOLS CAPABILITIES

Protocol Properties	Protocols									
	ALPHAN	S-SCTP	HMTP	IRTP	SCTP	RTP/I	ETP	UDP	RTNP	TCP
LAYER	APP	TRNSP	TRNSP	TRNSP	TRNSP	APP	TRNSP	TRNSP	NTW	TRNSP
PRIORITIZATION	YES	NO	NO	NO	NO	YES	NO	NO	YES	NO
KEY UPDATES	YES	YES	YES	NO	YES	NO	NO	NO		NO
PACKET HEADER (BYTES)	16			9		28		8		20
RELIABLE	PARTIAL	PARTIAL	PARTIAL	PARTIAL	PARTIAL	PARTIAL	NO	NO		YES
CONGESTION CONTROL	NO	ACK-NACK	NACK	Cwnd	ACK-NACK		RATE BASED	NO		Cwnd
BANDWITH OPTIMIZATION	NO	NO	NO	YES	NO	PARTIAL	YES	NO		YES
SEQUENCE NUMBER	YES	YES	YES	YES	YES	YES	NO	NO		YES
TIMESTAMP	YES	YES	NO	NO	NO	YES	NO	NO		YES
BUFFERS	YES	YES	YES	YES	NO	PARTIAL	NO	NO		YES
CONNECTION ORIENTED	NO	YES	NO	YES	NO	PARTIAL	NO	NO		YES
TCP FRIENDLY	NO	NO	NO	YES	NO	PARTIAL	YES	NO		YES
MULTI-PLATFORM	YES	YES	YES	YES	YES	YES	YES	YES	NO	YES

REFERENCES

[1] D. Wang, K. Tuer, M. Rossi, L. Ni, and J. Shu, "The effect of time delays on tele-haptics," in Haptic, Audio and Visual Environments and Their Applications, 2003. HAVE 2003. Proceedings. The 2nd IEEE Internatioal Workshop on, Sept. 2003, pp. 7 – 12.

[2] K. Hikichi, H. Morino, I. Arimoto, K. Sezaki, and Y. Yasuda, "The evaluation of delay jitter for haptics collaboration over the internet," in Global Telecommunications Conference, 2002. GLOBECOM '02. IEEE, vol. 2, Nov. 2002, pp. 1492 – 1496.

[3] J. Marsh, M. Glencross, S. Pettifer, and R. Hubbold, "A network architecture supporting consistent rich behavior in collaborative interactive applications," Visualization and Computer Graphics, IEEE Transactions on, vol. 12, no. 3, pp. 405 –416, May-June 2006.

[4] S. Shirmohammadi and N. D. Georganas, "An end-to-end communication architecture for collaborative virtual environments," Computer Networks, vol. 35, no. 3, pp. 351 – 367, 2001.

[5] S. Dodeller and N. D. Georganas, "Transport layer protocols for telehaptics update message," in Proc. 22nd Biennial Symposium on Communications, Queen's Univeristy, Canada, May 31-June3 2004.

[6] L. Ping, L. Wenjuan, and S. Zengqi, "Transport layer protocol reconfiguration for network-based robot control system," in Networking, Sensing and Control, 2005. Proceedings. 2005 IEEE, Mar. 2005, pp. 1049 – 1053.

[7] Y. Uchimura and T. Yakoh, "Bilateral robot system on the real-time network structure," Industrial Electronics, IEEE Transactions on, vol. 51, no. 5, pp. 940 – 946, Oct. 2004.

[8] R. Wirz, M. Ferre, R. Marin, J. Barrio, J. Claver, and J. Ortego, "Efficient transport protocol for networked haptics applications," in Haptics: Perception, Devices and Scenarios, ser. Lecture Notes in Computer Science, M. Ferre, Ed. Springer Berlin / Heidelberg, 2008, vol. 5024, pp. 3–12.

[9] M. Mauve, V. Hilt, C. Kuhmunch, and W. Effelsberg, "Rtp/i-toward a common application level protocol for distributed interactive media," Multimedia, IEEE Transactions on, vol. 3, no. 1, pp. 152–161, Mar. 2001.

[10] W. A. McNeely, K. D. Puterbaugh, and J. J. Troy, "Six degree-of-freedom haptic rendering using voxel sampling," in ACM SIGGRAPH 2005 Courses, ser. SIGGRAPH '05. New York, NY, USA: ACM, 2005.

[11] C. Mendoza and C. Laugier, “Realistic haptic rendering for highly deformable virtual objects,” in Virtual Reality, 2001. Proceedings. IEEE, Mar. 2001, pp. 264–269.

[12] N. Diolaiti, G. Niemeyer, F. Barbagli, and J. Salisbury, “Stability of haptic rendering: Discretization, quantization, time delay, and coulomb effects,” Robotics, IEEE Transactions on, vol. 22, no. 2, pp. 256 –268, April 2006.

[13] C. Demichelis and P. Chimento, IP packet delay variation metric for IP performance metrics (IPPM), IETF RFC 3393 Std., November 2002.

[14] M. Eid, J. Cha, and A. El Saddik, “Admux: An adaptive multiplexer for haptic–audio–visual data communication,” Instrumentation and Measurement, IEEE Transactions on, vol. 60, no. 1, pp. 21 –31, Jan. 2011.

[15] K. Iwata, Y. Ishibashi, N. Fukushima, and S. Sugawara, “Qoe assessment in haptic media, sound, and video transmission: Effect of playout buffering control,” Comput. Entertain., vol. 8, no. 2, pp. 1–12, Dec. 2010.

[16] K. M. Yap, A. Marshall, and W. Yu, “Providing qos for distributed haptic virtual environments in ip networks,” in Proceedings of the First International Conference on Immersive Telecommunications, ser. ImmersCom ’07. ICST, Brussels, Belgium, Belgium: ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2007, pp. 13:1–13:6.

[17] A. Boukerche, S. Shirmohammadi, and A. Hossain, “Moderating simulation lag in haptic virtual environments,” in Simulation Symposium, 2006. 39th Annual, April 2006, pp. 269 –277.

[18] C. Jay, M. Glencross, and R. Hubbold, “Modeling the effects of delayed haptic and visual feedback in a collaborative virtual environment,” ACM Trans. Comput.-Hum. Interact., vol. 14, no. 2, Aug 2007.

[19] R. Wirz, R. Marin, J. M. Claver, M. Ferre, R. Aracil, and J. Fernandez, “End-to-end congestion control protocols for remote programming of robots, using heterogeneous networks: A comparative analysis,” Robotics and Autonomous Systems, vol. 56, no. 10, pp. 865 – 874, 2008.

[20] H. Al Osman, M. Eid, R. Iglesias, A. El Saddik, , "ALPHAN: Application Layer Protocol for HAptic Networking," *Haptic, Audio and Visual Environments and Games, 2007. HAVE 2007. IEEE International Workshop on* , pp.96-101, 12-14 Oct. 2007

[21] A. Boukerche, H. Maamar, and A. Hossain, “An efficient hybrid multicast transport protocol for collaborative virtual environment with networked haptic,” Multimedia Systems, vol. 13, pp. 283–296, 2008.

[22] S. Floyd, V. Jacobson, S. McCanne, C.-G. Liu, and L. Zhang, “A reliable multicast framework for light-weight sessions and application level framing,” SIGCOMM Comput. Commun. Rev., vol. 25, no. 4, pp. 342–356, Oct. 1995.

[23] S. Paul, K. Sabnani, J.-H. Lin, and S. Bhattacharyya, “Reliable multicast transport protocol (rmtp),” Selected Areas in Communications, IEEE Journal on, vol. 15, no. 3, pp. 407 –421, April 1997.

[24] J. Pullen and V. Laviano, “A selectively reliable transport protocol for distributed interactive simulation,” in Proceedings of the 13th Workshop on Standards for the Interoperability of Distributed Simulations, J. Pullen and V. Laviano, Eds., 1995.

[25] R. Wirz, R. Marin, M. Ferre, J. Barrio, J. Claver, and J. Ortego, “Bidirectional transport protocol for teleoperated robots,” Industrial Electronics, IEEE Transactions on, vol. 56, no. 9, pp. 3772 –3781, Sept. 2009.

[26] Z. Cen, M. Mutka, D. Zhu, and N. Xi, “Supermedia transport for teleoperations over overlay networks,” in NETWORKING 2005. Springer Berlin / Heidelberg, 2005, vol. 3462, pp. 111–163.

[27] M. Handley, S. Floyd, J. Padhye, and J. Widmer, “Tcp friendly rate control (tfrc): Protocol specification,” RFC, 2003.

[28] A. Alturki and M. Alnuem, “Study of the effects of using some qos mechanisms on haptic transmission - using opnet modeler,” in Haptic Audio Visual Environments and Games (HAVE), 2011 IEEE International Workshop on, pp. 94 –101, Oct. 2011.

[29] A. Roach, A negative acknowledgement mechanism for signaling compression, IETF Network Working Group Std., May 2005.

[30] M. Smith and K. Ramakrishnan, “Formal specification and verification of safety and performance of tcp selective acknowledgment,” Networking, IEEE/ACM Transactions on, vol. 10, no. 2, pp. 193 –207, April 2002.

[31] Y. Mansour and B. Patt-Shamir, “Jitter control in qos networks,” IEEE/ACM Trans. Netw., vol. 9, no. 4, pp. 492–502, Aug. 2001.

[32] S. Floyd and V. Jacobson, “Link-sharing and resource management models for packet networks,” Networking, IEEE/ACM Transactions on, vol. 3, no. 4, pp. 365 –386, Aug. 1995.

[33] Y.-C. Huang, C.-H. Chu, and E. H.-K. Wu, “A novel congestion control mechanism on tfrc for streaming applications over wired-wireless networks,” in Proceedings of the 6th ACM workshop on Wireless multimedia networking and computing, ser. WMuNeP ’11. New York, NY, USA: ACM, pp. 9–16, 2011.

[34] D.-M. Chiu and R. Jain, “Analysis of the increase and decrease algorithms for congestion avoidance in computer networks,” Computer Networks and ISDN Systems, vol. 17, no. 1, pp. 1 – 14, 1989.

[35] R. Rejaie, M. Handley, and D. Estrin, “Rap: An end-to-end rate-based congestion control mechanism for realtime streams in the internet,” in INFOCOM ’99. Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE, vol. 3, pp. 1337 –1345, Mar. 1999.

[36] O. Wongwirat, S. Ohara, and N. Chotikakamthorn, “An adaptive buffer control using moving average smoothing technique for haptic media synchronization,” in Systems, Man and Cybernetics, 2005 IEEE International Conference on, vol. 2, , pp. 1334 –1340, Oct. 2005.

[37] Y. Miyashita, Y. Ishibashi, N. Fukushima, S. Sugawara, and K. E. Psannis, “Quality of experience assessment of media synchronization in chorus among multi-points,” The Journal of the Institute of Image Information and Television Engineers, vol. 66, no. 4, pp. J114–J118, April 2012.

[38] Y. Miyashita, Y. Ishibashi, N. Fukushima, S. Sugawara, and K. Psannis, “Qoe assessment of group synchronization in networked chorus with voice and video,” in TENCON 2011 - 2011 IEEE Region 10 Conference, pp. 192 –196, Nov. 2011.

[39] K. Hosoya, Y. Ishibashi, S. Sugawara, and K. Psannis, “Group synchronization control considering difference of conversation roles,” in Consumer Electronics, 2009. ISCE ’09. IEEE 13th International Symposium on Consumer Electronics, pp. 948 –952, May 2009.

[40] K. Hosoya, Y. Ishibashi, S. Sugawara, and K. E. Psannis, “Effects of group synchronization control in networked virtual environments with avatars,” in Proceedings of the 2008 12th IEEE/ACM International Symposium on Distributed Simulation and Real-Time Applications. IEEE Computer Society, pp. 119–127, Oct. 2008.

[41] K. Nahrstedt, A. Arefin, R. Rivas, P. Agarwal, Z. Huang, W. Wu, and Z. Yang, “Qos and resource management in distributed interactive multimedia environments,” Multimedia Tools and Applications, vol. 51, pp. 99–132, 2011.

[42] I. Psaras and V. Tsaoussidis, “On the properties of an adaptive tcp minimum rto,” Computer Communications, vol. 32, no. 5, pp. 888 – 895, 2009.

[43] V. T. Ioannis Psaras, “On the properties of an additive increase rate accelerator,” Computer Networks, vol. 53, no. 5, pp. 583 – 595, 2009.

[44] P. Sarolahti, M. Allman, and S. Floyd, “Determining an appropriate sending rate over an underutilized network path,” Computer Networks, vol. 51, no. 7, pp. 1815 – 1832, 2007.

[45] M. Rocha, M. Maia, I. Cunha, J. Almeida, and S. Campos, “Scalable media streaming to interactive users,” in Proceedings of the 13th annual ACM international conference on Multimedia, ser. MULTIMEDIA’ 05, pp. 966–975,2005.

[46] H.-P. Shiang and M. van der Schaar, “Content-aware tcp-friendly congestion control for multimedia transmission,” in Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on, pp. 2356 –2359, May 2011.

[47] N. Singhal and R. Sharma, “Survey on tcp friendly congestion control for unicast and multicast traffic,” International Journal of Computer Applications, vol. 26, no. 1, pp. 23–30, July 2011.