Network-Aware Adaptive Sampling for Low Bitrate Telehaptic Communication

Vineet Gokhale[1], Jayakrishnan Nair[2], Subhasis Chaudhuri[2] and Suhas Kakade[2]

[1] University of South Bohemia, České Budějovice, Czech Republic
[2] Indian Institute of Technology Bombay, Mumbai, India.

Abstract. While the adaptive sampling technique for kinesthetic signal transmission offers a phenomenal reduction in the time-average data rate, it does not guarantee a meaningful upper bound on the instantaneous rate, which can occasionally be comparable to the peak rate. This implies that for Quality of Service (QoS) compliance, a network bandwidth equal to the peak rate must be reserved apriori for the telehaptic stream at all times. On a shared network with unknown and time-varying cross-traffic, this is not always feasible. In order to address the intermittently high bandwidth demand as well as the network-obliviousness of adaptive sampling, we propose *NaPAS: Network-aware Packetization for Adaptive Sampling*. The idea is to intelligently merge multiple haptic samples generated by adaptive sampling, depending on the changing network conditions. This results in an *elastic* telehaptic traffic that can adapt to the available network bandwidth. Through qualitative and quantitative measures, we evaluate the performance of NaPAS and demonstrate that it outperforms standard adaptive sampling (SAS) in terms of maintaining the haptic perceptual quality and QoS compliance, while also being friendlier to the network cross-traffic.

Keywords: *Telehaptic communication, adaptive sampling, QoS, shared network*

1 Introduction

The possibility of transmitting touch signals (in addition to audio and video) over a network has unlocked the doors to a new realm of *telehaptic* applications, like telesurgery [1] and distributed touch therapy [3]. In order to perform such sensitive tasks in a seamless manner, strict Quality of Service (QoS) requirements, reported in Table 1, need to be satisfied [13]. It is to be noted that the presence of haptic feedback makes the *teleoperation* extremely vulnerable to the irregularities of the communication network, viz., delay, jitter and packet loss. Non-compliance to these QoS constraints can adversely affect the stability of the haptic control loop, albeit there exist control architectures, such as [16], that alleviate the effect to some extent.

In order to realize a QoS-compliant teleoperation, one typically needs to leverage an existing shared network (like the internet), since deploying dedicated networks solely for the purpose of teleoperation may be practically infeasible. However, ensuring telehaptic QoS compliance on a shared network is challenging, since the cross-traffic is both unknown as well as time-varying. Indeed, network congestion (overloading) can lead to high delays, jitter and packet losses, often resulting in QoS violations. Moreover, certain networks tend to be resource constrained in nature; for instance, rapidly deployed adhoc networks for emergency operations and connections in rural areas. Therefore, the communication scheme in a teleoperation should be *network-aware* in nature, with the ability to relieve congestion by dynamically tuning the traffic rate to match the available network bandwidth in real time.

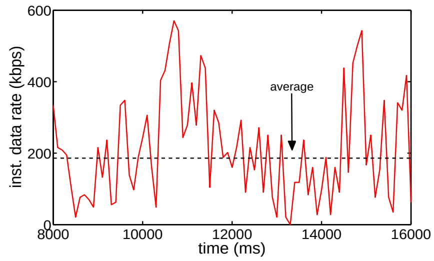
Media	Delay (ms)	Jitter (ms)	Loss (%)
Haptic	30	10	10
Audio	150	30	1
Video	400	30	1

Table 1: QoS specifications to be satisfied for carrying out seamless teleoperation.

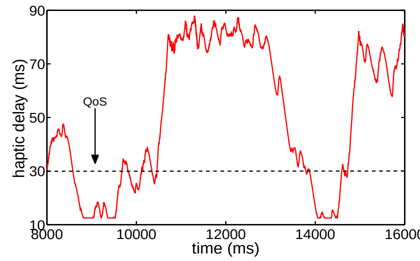
The past two decades have witnessed rapid advancements in the design and development of communication techniques for haptic based teleoperation. Amongst them, the most widely accepted model is *adaptive sampling* – a human perception based compression scheme for haptic signals [6,11,15]. Adaptive sampling classifies a haptic sample as *perceptually significant* if the percentage change in its amplitude with respect to a certain reference exceeds a pre-defined threshold δ. The work in [11] demonstrated that transmission of only the perceptually significant samples leads to a substantial reduction in the telehaptic data rate of up to 90%, without hampering the human perception. It is worth remarking that the data rate of the adaptive sampling technique depends purely on the haptic signal profile; a fast varying signal results in a high data rate, and vice-versa. This makes the adaptive sampling scheme *network-oblivious* in nature.

It is important to note that the rate reduction of adaptive sampling is in a *time-average* sense. In Figure 1a, we plot the instantaneous rate of the adaptive sampling scheme for a real haptic trace recorded during a haptic activity [2]. It can be seen that despite the low time-average rate (186 kbps in this case), the instantaneous rate exhibits rapid fluctuations, occasionally reaching the peak value of around 600 kbps.

As per the recommendations of the references [6,11,15], reserving an amount of network bandwidth equal to the time-average rate guarantees QoS compliance, and hence smooth teleoperation. On the contrary, we demonstrate in Figure 1b that the above network provisioning strategy (reserving 186 kbps for telehaptic stream) results in severe QoS violations, particularly when the instantaneous rate surpasses the average rate; see, for example, the range [11000, 13000] ms. Such violation of QoS results in impairment of haptic perception, as we demonstrate through subjective experiments in Section 4.2. Hence, we conclude that despite guaranteeing low time-average rate, adaptive sampling scheme provides no meaningful economies from the standpoint of network bandwidth requirement, since for QoS-compliance the network should be provisioned for the peak telehaptic rate. Henceforth, we refer to the adaptive sampling scheme described above as *standard adaptive sampling* (SAS).



(a)



(b)

Fig. 1: Temporal evolution of (a) instantaneous data rate of SAS (b) haptic delay when the network is provisioned for the time-average data rate.

To summarize, SAS suffers two major limitations: (i) it is *network-oblivious*, and (ii) it lacks fine-grained control on the instantaneous transmission rate.

A recent work [9] attempts to address the aforementioned drawbacks of SAS through the design of *Dynamic Packetization Module (DPM)*. DPM is a lossless protocol that transmits every haptic sample, irrespective of its significance, in a *network-aware* manner. The idea is to merge k haptic samples into a single packet based on the changing network conditions.This packetization parameter k is dynamically tuned to match the instantaneous rate to the available network bandwidth. Note that a higher k corresponds to a lower transmission rate, due to a reduction in the packet header overhead.[3] The usage of k as the control lever enables DPM to generate steady traffic at multiple resolutions, where each resolution corresponds to a particular value of k, thereby offering a fine-grained control on the instantaneous rate. Note that DPM's data rate is insensitive to the haptic signal profile. Hence, DPM transmits even the perceptually insignificant samples (90%) leading to an improper utilization of the network resources.

To summarize, SAS provides a significantly low average rate, compared to the peak rate, but is network-oblivious and lacks control on the instantaneous rate. On the other hand, DPM is network-aware and provides a fine-grained control on the instantaneous rate, but transmits unnecessary samples leading to a higher data rate. The question we ask in this paper is the following: Can we leverage the benefits of both SAS and DPM to obtain the best of both worlds?

In this paper, we propose NaPAS (Network-aware Packetization for Adaptive Sampling) for transmitting only the perceptually significant samples in a network-aware manner, characterized by a fine-grained control on the instantaneous rate. Like DPM, NaPAS responds to network congestion by aggressively cutting its transmission rate, thereby minimizing the chances of a QoS violation. Additionally, it compresses the generated haptic signal by transmitting

---

[3] Given the high sampling rate of the haptic stream (typically 1 kHz), packet headers can account for upto 73% of the transmission rate on the forward channel when each haptic sample is packetized separately [9]. As a result, there is considerable room for data rate adaptation by varying the control parameter k (which determines the telehaptic packetization rate).

only perceptually significant samples, freeing up the network resources for other cross-traffic flows (as demonstrated in Section 4).

We carry out quantitative and qualitative assessments of our proposal through extensive simulations and bilateral telehaptic experiments, respectively. Our investigations reveal that NaPAS outperforms standard adaptive sampling in terms of telehaptic QoS compliance, and also in terms of preserving the quality of telehaptic interaction even under heavily congested network conditions. Further, we demonstrate that the dynamics of our technique are friendly to exogenous cross-traffic streams, more so than SAS and DPM.

2 Telehaptic Communication

In this section, we briefly explain the general framework of a point-to-point telehaptic communication framework, and then move to the detailed design of the proposed NaPAS framework.

2.1 Point-to-Point Framework

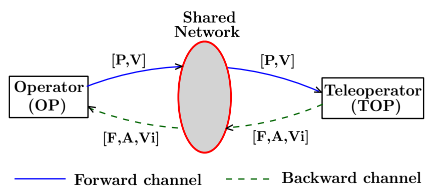


Fig. 2: Communication framework of a typical point-to-point teleoperation. Notations: [P, V]: [position, velocity], [F, A, Vi]: [force, audio, video].

The point-to-point teleoperation system, shown in Figure 2, consists of a human operator (OP) controlling the remote robotic manipulator known as the teleoperator (TOP). The OP transmits the current position and velocity commands on the forward channel. The TOP follows the trajectory of the OP through execution of the received commands, and in response transmits the captured audio and video signals along with the force feedback on the backward channel. This configuration is generally referred to as *two-channel position-force architecture* [12]. Note that the communication is inherently bidirectional and asymmetric in a teleoperation paradigm.

In the remainder of this section, we present the design details of the proposed communication framework; see Figure 3. For the ease of presentation, we consider the standard haptic sampling rate of 1 kHz. In this work, we restrict our focus to rate control on the forward channel. On the backward channel, due to the presence of audio and video, the haptic data constitutes only a small portion of the overall payload. Hence, from the standpoint of data rate reduction, SAS on the backward channel is not as effective as it is on the forward channel. Accordingly, we perform standard DPM [9] on the backward channel.

2.2 Network Feedback

In order to accurately monitor the network under asymmetric conditions, we adopt the delay-based *network feedback* scheme proposed in [9]. The scheme in [9]

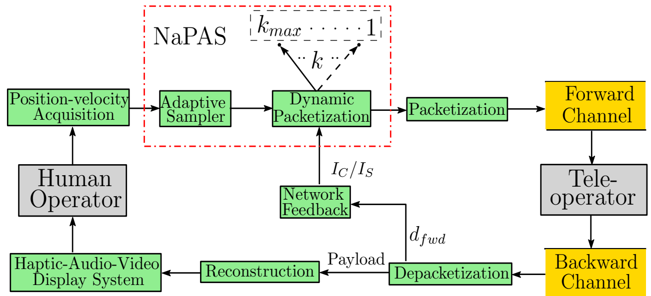


Fig. 3: Proposed communication framework for teleoperation featuring NaPAS on the forward channel. d_{fwd} denotes the forward channel delay that is piggybacked on the backward channel packets.

exploits the bidirectional property of telehaptic communication for conveying to each transmitter the network delays on its channel by piggybacking this information on packets sent on the reverse channel. Specifically, the TOP piggybacks the end-to-end delays from the forward channel (d_{fwd}) on the backward channel packets, and the OP piggybacks the end-to-end delays on the backward channel on the forward channel packets. The OP extracts the piggybacked delay from the packet header to analyze the congestion state on the forward channel. Note that end-to-end delays increase during congestion, and remain steady otherwise. Thus, if N successive, non-duplicate delay samples exhibit an increasing trend, then the OP infers that congestion is present in the forward channel.[4] In this case, a congestion trigger I_C is generated. On the other hand, if N successive, non-duplicate delay samples exhibit a steady trend, then the OP infers that the forward channel is uncongested. In this case, a steady trigger I_S is generated. NaPAS performs rate adaptation based on these triggers, as described next.

2.3 NaPAS Rate Control

The goal of NaPAS is to transmit the perceptually significant samples with a fine-grained control on the instantaneous rate in a network-aware manner. As shown in Figure 3, NaPAS subjects the perceptually significant haptic samples to a dynamic packetization process that results in an adaptive transmission rate.

The working principle of NaPAS is as follows: The time dimension is divided into continuous, non-overlapping blocks, each of length k-milliseconds, as shown in Figure 4. Let the time interval $t \in [t_s, t_s + k)$ indicate the current block, where t_s and $t_s + k$ denote the time instants of the start and the end of the current block. The perceptually significant haptic samples, generated from adaptive sampling in the range $t \in [t_s, t_s + k)$ are merged into a single packet for transmission on the forward channel. Note that a packet can carry at most k haptic samples; see, for example, the second block in Figure 4. No packet is generated if all the k samples in a block are perceptually insignificant; see, for example, the third block in Figure 4. This is the primary departure from DPM,

[4] The TOP transmits duplicate copies of a delay measurement if it transmits multiple packets in between adjacent receptions.

which essentially merges all haptic samples generated in a block, whether or not they are perceptually significant. Note that the parameter k allows for a tradeoff between packetization delay and transmission rate. Since haptic samples suffer a maximum packetization delay of $k-1$ ms, an increase in k results in an overall increase in packetization delay. However, an increase in k also results in a lower packet rate, and consequently a lower data rate.

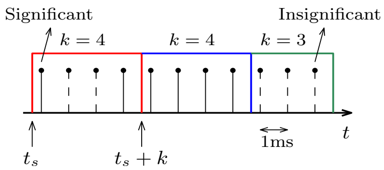


Fig. 4: Demonstration of the working principle of NaPAS based on dividing the time into blocks.

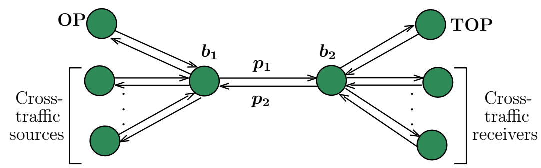


Fig. 5: Network topology used for the simulations. p_1 and p_2 - bottleneck links; b_1 and b_2 - intermediate nodes.

We perform rate control by using k as the control parameter based on the generated triggers. When the congestion trigger I_C is generated (indicating congestion), the update $k_{max} \leftarrow k$ is executed, where k_{max} denotes the maximum permissible value of k. This results in a drastic reduction in the instantaneous rate, facilitating the rapid draining of queues at the intermediate routers (which are either overflowing or fast filling). This minimizes the risk of QoS violations. When the steady trigger I_S is generated (indicating an uncongested network), the framework executes the update $k-1 \leftarrow k$, as long as $k > 1$. This results in a prudent increment in the instantaneous rate, seeking to avoid congestion while probing if the network has the capacity to accommodate the rate increment. This gradual increase / aggresive decrease in transmission rate is in line with the classical additive increase / multiplicative decrease (AIMD) principle [4] in the congestion control literature. Note that when $k=1$, NaPAS is equivalent to SAS. Hence, when the network is uncongested NaPAS reverts to SAS to minimize the packetization delay encountered by the significant samples. Note that during congestion, the data rate of NaPAS is at most equal to the data rate of DPM, since NaPAS transmits only perceptually significant samples while DPM transmits all samples.

At the TOP, the perceptually insignificant samples position-velocity samples are interpolated using the zero-order hold strategy. We choose $k_{max} = 4$, thereby restricting the maximum length of a block to 4 ms [9]. In our implementation, we set the parameter $N = 10$.

We conclude with a remark regarding the rendering of the received haptic signal at the TOP. Due to block processing, the TOP receives the latest (perceptually significant) haptic sample along with a few (upto $k_{max} - 1$) previous perceptually significant samples simultaneously. For reliable teleoperation, it is important to ensure that the TOP replicates the OP's movements as accurately as possible. Therefore, it is crucial to play-out all the received samples at the TOP sequentially, rather than render only the latest sample. This approach of transmitting/displaying a significant haptic sample even after the generation of more recent ones has also been advocated in the literature; see, for example, [5,7].

3 Experimental Setup

In this section, we give a detailed description of the setups used for qualitative and quantitative evaluation of the proposed NaPAS framework.

3.1 Simulation Testbed

We perform quantitative evaluation using NS3 - a discrete event network simulator [14]. We simulate a single bottleneck dumbbell network topology, as shown in Figure 5. We introduce cross-traffic sources on the forward channel. Hence, p_1 is the bottleneck link on the forward channel. We set the capacity of p_1 to 1500 kbps. The access links to the intermediate nodes b_1 and b_2 have high capacity. The propagation delay of each link is set to 4 ms. Hence, the end-to-end propagation (one-way) delay between OP and TOP is 12 ms. The details pertaining to the cross-traffic streams are reported in Section 4.

For our simulations, we use real haptic traces recorded during the telepottery task described in Section 3.2, and also from the trace repository of TU Munich [2]. We set the adaptive sampling threshold $\delta = 10\%$ as prescribed in [11], but can be tuned for an individual user. For brevity, we report the results corresponding to a single trace. However, we note that our findings remain consistent across different traces.

3.2 Subjective Evaluation

For qualitative evaluations, we use the telepottery setup in which the human subject manipulates a remote, virtual clay model through haptic and video feedback transmitted over a real network. It includes a network emulator tool for reproducing the effects of a shared network [9]. The task for the subject is to carve the clay model into a nice looking pot. Initially, the subject undergoes appropriate training involving a detailed explanation and hands-on demonstration of the telepottery task for familiarization with the experiments. The training is performed on a high bandwidth (100 Mbps) network to avoid the impact of network congestion. After the training, the subject is moved to a test setup where the bandwidth and one-way latency are set to 1500 kbps and 12 ms, respectively, as in simulations.

The testing phase consists of performing telepottery task twice: once with SAS, and once with NaPAS. On a scale of 5, the subjects grade the telepottery perception in each of the two test cases relative to the training phase based on three standard perceptual parameters: *transparency*, *smoothness*, and *overall experience* [9]. The subject provides a grade for each perceptual parameter in each test case relative to training experience based on the following grading scale: 5 - imperceptible; 4 - slight disturbance, but not annoying; 3 - slightly annoying; 2 - annoying; 1 - very annoying.

We performed the subjective evaluations with 20 human subjects (10 male and 10 female) belonging to the age group of 20 to 32 years, none suffering from any known neurophysiological disorders. Out of them, 3 were regular users of haptic devices and the rest were novices. Nevertheless, all subjects underwent meticulous training prior to the test experiments. The subjects provided informed consents prior to the experiments.

4 Experimental Results

In this section, we report the quantitative (Section 4.1) and qualitative (Section 4.2) performances of NaPAS, standard adaptive sampling, and DPM.

4.1 Simulation Results

We consider two classes of cross-traffic flows that are typically seen in a shared network: Constant Bitrate (CBR) cross-traffic (Section 4.1.1), and Transmission Control Protocol (TCP) cross-traffic (Section 4.1.2).

For a standard haptic sampling rate of 1 kHz, we get a forward channel peak rate of 688 kbps with a packet of size 86 bytes (including telehaptic payload and packet headers) transmitted every millisecond (see [9] for details). Through analysis, it can be shown that under heavily congested conditions, NaPAS can guarantee an upper bound of 316 kbps on the instantaneous data rate (this corresponds to $k = 4$).

4.1.1 Constant Bitrate (CBR) Cross-Traffic: It is important to remark that under CBR cross-traffic the performance of DPM and NaPAS are comparable. Hence, in this section we only report the performances of SAS and NaPAS. We introduce a CBR cross-traffic source with rate R_{cross} on the forward channel. For the test signal, we observe that for R_{cross} >1250 kbps the congestion level is beyond the control of the proposed technique. Hence, we conduct the simulations over the range $R_{cross} \in [0, 1250]$ kbps.

We begin by presenting the dynamics of k and the corresponding end-to-end haptic delay measured at the TOP for NaPAS in Figures 6 and 7, respectively. For this experiment, we set $R_{cross} = 1200$ kbps starting at $t = 0$, so that the telehaptic stream gets a bandwidth of 300 kbps. It can be seen that as the delay increases due to congestion, NaPAS quickly switches to $k = 4$ resulting in an effective congestion control, and thereby guaranteeing a strict QoS-compliance. During intervals of steady delays, the OP reduces k in a step-wise manner. It should be noted that the duration of the steady delay regions is not a constant. For example, the steady delay region corresponding to $k = 4$ starting at $t =$ 1000 ms is longer than that ending at $t = 800$ ms. This is an artifact of the haptic signal profile; during periods when the rate of generation of perceptually significant haptic samples is greater, the rate of packet transmissions on the forward channel is also greater. This results in a greater reception rate of (non-duplicate) delay measurements at the OP, in turn leading to faster updates in k.

Under the same cross-traffic setting, we compare the end-to-end haptic delays resulting from SAS and the proposed NaPAS scheme. Note that SAS has a much lower average data rate (186 kbps) compared to the available bandwidth (300

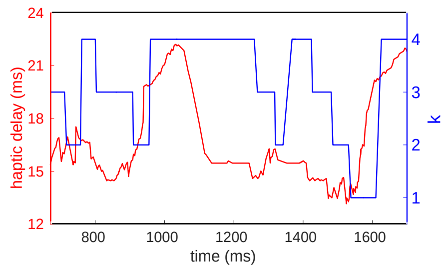


Fig. 6: Temporal evolution of end-to-end haptic delay and the corresponding value of k for $R_{cross} = 1200$ kbps.

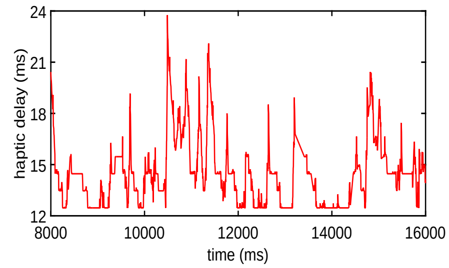


Fig. 7: Temporal evolution of end-to-end haptic delay for NaPAS, with $R_{cross} =$ 1200 kbps.

kbps). However, due to its network-obliviousness, the standard adaptive sampling scheme faces severe violations of the haptic delay QoS conditions, comparable to Figure 1b. This can lead to perceptual artifacts that can cause significant degradation in the quality of teleoperation (as demonstrated by the results of our subjective evaluations in Section 4.2). On the other hand, due to the timely congestion detection and control measures, NaPAS comfortably adheres to the delay QoS requirements of teleoperation, as shown in Figure 7. The above observations demonstrate that from a network provisioning and QoS standpoint, it is essential to control the *instantaneous* data rate in a network-aware manner. We note that the haptic packet loss in the above experiment for both schemes is zero.

As per the standard definition, haptic jitter refers to the variation in the delay encountered by successive haptic samples. However, we note that for an adaptive sampling based communication, the haptic packets are generated irregularly in time. Thus, the standard definition of jitter does not apply, and so we do not report on haptic jitter in this paper.

Until now all our measurements were carried out in presence of a steady cross-traffic. We now use vary R_{cross} over time to demonstrate the ability of NaPAS to adapt to changing network conditions. Specifically, we introduce a time-varying cross-traffic profile as shown below.

$$R_{cross} = \begin{cases} 700 \text{ kbps}, & \text{for } t \leq 10s \\ 1250 \text{ kbps}, & \text{for } 10 < t \leq 11.5s \\ 900 \text{ kbps}, & \text{for } 11.5 < t \leq 12.5s \\ 700 \text{ kbps}, & \text{for } t > 12.5s \end{cases}$$

In Figure 8, we report the temporal variation of the instantaneous rates for SAS and NaPAS. While the standard adaptive sampling is insensitive to the cross-traffic variations, the NaPAS scheme senses the level of congestion and tunes the instantaneous rate appropriately to below the available bandwidth. During heavy congestion ($R_{cross} = 1250$ kbps) it achieves peak congestion control, and when the network is uncongested ($R_{cross} = 700$ kbps) it reverts to SAS. Indeed, we note that NaPAS achieves a *network-aware haptic signal compression* that beats SAS. The maximum haptic delays in this experiment for SAS and

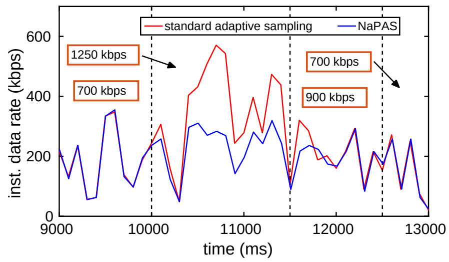


Fig. 8: Temporal evolution of the instantaneous data rate of SAS (red curve) and NaPAS (blue curve) against a time-varying cross-traffic.

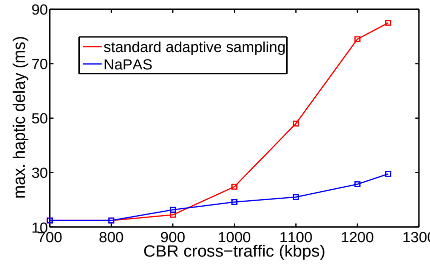


Fig. 9: Comparison of performances of SAS (red curve) and NaPAS (blue curve) in terms of maximum haptic delay over a wide range of R_{cross}.

NaPAS are 85.19 ms and 29.42 ms, respectively. Note that NaPAS is able to ensure QoS compliance in spite of heavy cross-traffic ($R_{cross} = 1250$ kbps) during part of the experiment. For both schemes, haptic packet losses are measured to be zero in this experiment.

Finally, we report the variation of the the maximum haptic delay (Figure 9) for both techniques across the considered range of R_{cross}. When $R_{cross} < 800$ kbps, the network is uncongested and hence there is no notable difference in the performances of the two techniques. As R_{cross} increases, SAS continues to transmit packets in a network-oblivious manner. This causes extremely high end-to-end haptic delays, severely violating the QoS needs. On the other hand, the network-aware behavior of NaPAS ensures QoS compliance even under heavily congested network conditions. Further, the back-off behavior with increasing R_{cross} suggests that the proposed technique is friendly to CBR cross-traffic streams, which are themselves network-oblivious in nature. Once again, haptic packet losses are zero for both schemes under consideration.

4.1.2 Transmission control protocol (TCP) Cross-Traffic: We now turn to Transmission control protocol (TCP) cross-traffic. TCP is the dominant rate control protocol on the internet, employed by over 90% of all internet traffic [17]. Our goal is to analyze the performance of SAS, DPM, and NaPAS in presence of TCP, and get a sense of the improvement of NaPAS over the rest. For our experiments, we consider TCP NewReno [8] which is the most widely deployed variant of TCP. In these experiments, we add a TCP source on the forward channel, and switch off the CBR cross-traffic source. We configure the queue size at b_1 to 2.7 kB as prescribed in [10].

The maximum haptic delays are measured to be 26.51 ms, 29.97 ms, and 29.95 ms for SAS, DPM, and NaPAS, respectively. Note that the dynamic packetization process in DPM and NaPAS results in a marginally higher haptic delay (around 3 ms) compared to SAS. However, it is worth remarking that all of the above schemes satisfy the haptic delay QoS conditions. Further, we note that the telehaptic stream suffers zero losses in all three cases.

Since the impact of TCP stream on telehaptic stream is comparable for the above three schemes, we move on to evaluating the impact of the haptic stream on the TCP cross-traffic stream. We consider the TCP throughput as the performance metric for this evaluation. The TCP throughput under SAS, DPM and NaPAS are measured to be 1160 kbps, 1114 kbps, and 1270 kbps, respectively. It can be observed that NaPAS yields significant improvement in TCP performance over both SAS (due to its lower peak transmission rate) and DPM (due to the signal compression obtained by transmitting only the perceptually significant samples). We conclude that NaPAS is friendlier to TCP cross-traffic than DPM and SAS.

4.2 Subjective Evaluation

We now report the qualitative evaluation of SAS and NaPAS in presence of $R_{cross} = 1180$ kbps. Note that for this value of R_{cross}, both DPM and NaPAS guarantee QoS compliance. Both these schemes transmit the perceptually significant samples in addition to employing the same congestion control scheme. Hence, from the standpoint of perception, we expect their behavior to be similar. We validate this argument through our subjective evaluation. However, for the purpose of brevity, in this section we do not report the findings for DPM. Table 2 presents the mean opinion score (MOS) and the standard deviation (SD) for the subject grades. It can be seen that while SAS introduces significant perceptual degradation, NaPAS is capable of preserving the perceptual quality of telepottery (in comparison with training) even under heavily congested network conditions. We perform paired t-test in order to statistically validate our findings. The results for the three perceptual parameters are as follows: (i) *transparency* - $t(19) = 7.29$, $p < 0.001$; (ii) *smoothness* - $t(19) = 10.43$, $p < 0.001$; (iii) *overall experience* - $t(19) = 10.50$, $p < 0.001$. This further confirms that NaPAS outperforms SAS under heavy CBR cross-traffic conditions in terms of preserving the perceptual quality of telepottery.

	Transparency		Smoothness		Overall Exp.	
	MOS	SD	MOS	SD	MOS	SD
SAS	2.05	0.88	1.85	0.87	1.90	0.71
NaPAS	3.95	0.68	4.30	0.47	4.20	0.52

Table 2: Mean opinion scores (MOS) and standard deviation (SD) of subject grades for the perceptual parameters corresponding to SAS and NaPAS.

5 Conclusions

In this paper, we demonstrated that provisioning the network for time-average data rate given by SAS causes severe QoS violations. In order to overcome this drawback, we proposed NaPAS, a network-aware refinement of SAS for teleoperation in resource constrained networks. Our simulations revealed that the proposed technique outperforms SAS and DPM in terms of telehaptic QoS compliance, as well as friendliness to network cross-traffic. Through subjective evaluations we demonstrated that NaPAS outperforms SAS in preserving the perceptual quality of telepottery even under heavy cross-traffic scenarios.

References

1. Anderson, R., Spong, M.: Bilateral control of teleoperators with time delay. Automatic Control, IEEE Transactions on 34(5), 494–501 (1989)
2. Bhardwaj, A., Cizmeci, B., Steinbach, E., Liu, Q., Eid, M., Araujo, J., El Saddik, A., Kundu, R., Liu, X., Holland, O., Luden, M., Oteafy, S., Prasad, V.: A candidate hardware and software reference setup for kinesthetic codec standardization. In: International Symposium on Haptic, Audio and Visual Environments and Games (HAVE) (2017)
3. Bonanni, L., Vaucelle, C., Lieberman, J., Zuckerman, O.: Taptap: a haptic wearable for asynchronous distributed touch therapy. In: CHI'06 extended abstracts on Human factors in computing systems. pp. 580–585. ACM (2006)
4. Chiu, D.M., Jain, R.: Analysis of the increase/decrease algorithms for congestion avoidance in computer networks. Computer Networks and ISDN Systems 17(1), 1–14 (1989)
5. Cizmeci, B., Chaudhari, R., Xu, X., Alt, N., Steinbach, E.: A visual-haptic multiplexing scheme for teleoperation over constant-bitrate communication links. In: Haptics: Neuroscience, Devices, Modeling, and Applications, pp. 131–138. Springer (2014)
6. Clarke, S., Schillhuber, G., Zaeh, M.F., Ulbrich, H.: Telepresence across delayed networks: a combined prediction and compression approach. In: Haptic Audio Visual Environments and their Applications, 2006. HAVE 2006. IEEE International Workshop on. pp. 171–175. IEEE (2006)
7. Condoluci, M., Mahmoodi, T., Steinbach, E., Dohler, M.: Soft resource reservation for low-delayed teleoperation over mobile networks. IEEE Access (2017)
8. Floyd, S., Gurtov, A., Henderson, T.: The newreno modification to tcp's fast recovery algorithm (2004)
9. Gokhale, V., Nair, J., Chaudhuri, S.: Congestion control for network-aware telehaptic communication. ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM) 13(2), 17 (2017)
10. Gokhale, V., Nair, J., Chaudhuri, S.: Teleoperation over a shared network: When does it work? In: International Symposium on Haptic, Audio and Visual Environments and Games (HAVE) (2017)
11. Hinterseer, P., Hirche, S., Chaudhuri, S., Steinbach, E., Buss, M.: Perception-based data reduction and transmission of haptic data in telepresence and teleaction systems. IEEE Transactions on Signal Processing 56(2), 588–597 (2008)
12. Lawrence, D.A.: Stability and transparency in bilateral teleoperation. IEEE transactions on robotics and automation 9(5), 624–637 (1993)
13. Marshall, A., Yap, K.M., Yu, W.: Providing qos for networked peers in distributed haptic virtual environments. Advances in Multimedia (2008)
14. ns3: The network simulator (2011), `http://www.nsnam.org/`
15. Sakr, N., Georganas, N.D., Zhao, J.: Human perception-based data reduction for haptic communication in six-dof telepresence systems. IEEE Transactions on Instrumentation and Measurement 60(11), 3534–3546 (2011)
16. Xu, X., Cizmeci, B., Schuwerk, C., Steinbach, E.: Haptic data reduction for time-delayed teleoperation using the time domain passivity approach. In: IEEE World Haptics Conference (WHC). pp. 512–518 (2015)
17. Yao, S., Xue, F., Mukherjee, B., Yoo, S.B., Dixit, S.: Electrical ingress buffering and traffic aggregation for optical packet switching and their effect on tcp-level performance in optical mesh networks. IEEE Communications Magazine 40(9), 66–72 (2002)