5G-Based Low-Latency Teleoperation: Two-Way Timeout Approach

Burak Kizilkaya$^{(\boxtimes)}$, Olaoluwa Popoola , Guodong Zhao ,
and Muhammad Ali Imran

School of Engineering, University of Glasgow, Glasgow, UK
`b.kizilaya.1@research.gla.ac.uk,`
`{olaoluwa.popoola,guodong.zhao,muhammad.imran}@glasgow.ac.uk`

Abstract. Recent advances in communications and robotics enable myriad teleoperation applications, empowering real-time remote operation in various application areas such as healthcare, education, manufacturing, and aerial manipulation. The main problems in teleoperation systems are the time delay and packet loss caused by poor network conditions. Ultra-reliable Low-Latency Communication (URLLC), one of the key services of fifth-generation cellular communications (5G), is expected to enable real-time teleoperation by mitigating latency and reliability issues of pre-5G communications. In this study, we develop a 5G-enabled teleoperation testbed to conduct experiments on communication latency and packet loss, demonstrating the current capabilities of 5G communications for teleoperation applications. Furthermore, we propose a two-way timeout approach to reduce the communication latency. The proposed approach reduces the end-to-end (E2E) latency by limiting the waiting time for new packet reception. Extensive latency and packet loss experiments are conducted to demonstrate the superiority of the proposed approach compared to the benchmark approach without a timeout. The experimental results corroborate that the proposed approach can reduce E2E latency by up to 65% and improves overall reliability.

Keywords: Teleoperation · Robotics · 5G · URLLC · Low-Latency

1 Introduction

Advances in communications and robotics uncover myriad teleoperation applications in various areas such as medical diagnosis and surgery [6], aerial manipulation [22], learning and education [12], nuclear industry [17], and space teleoperation [19]. A typical bilateral teleoperation system enables the remote environment manipulation over communication systems while empowering the sense of touch and telepresence over haptic communications. In general, such systems utilize a controller with a haptic interface that a human operator uses to control the remote robot/teleoperator. The operator's inputs are sampled and transmitted over the communication system to be executed by the teleoperator to

accomplish a task. In bilateral teleoperation, the operator receives the feedback signal, which closes the control loop. One of the main problems in teleoperation is the time delay introduced by hardware, computation, and communication systems which degrades the teleoperation performance [13], and real-time control becomes unstable. This issue has been investigated extensively in existing literature regarding communications and control theory perspectives.

Regarding communications, current advancements in 5G provide high data rates, low communication latency, and ultra-reliable exchange of information, with teleoperation being one of the most important scenarios of ultra-reliable low-latency communications (URLLC) [2]. Communication studies optimize communication system parameters to provide better solutions in terms of communication reliability and latency, which in turn enables better teleoperation performance [7,8,18]. In [18], prediction is performed to mitigate time delays in control systems, where authors jointly optimize prediction and communication systems to minimize wireless resource consumption. Similarly, a predictive actuation framework is proposed in [7], where lost packets are predicted at the receiver to minimize the age of information (AoI) and transmit power. In [8], communication and prediction systems are optimized jointly to achieve better user-experienced delay and system reliability in haptic feedback enabled teleoperation considering communication quality of service (QoS).

From a control theory perspective, control parameters are optimized under perfect and imperfect wireless channel assumptions [3,15,20]. In [20], a control scheme switching approach is proposed based on Quality of Experience (QoE). The proposed switching approach selects the best-performing control scheme adaptively considering QoE. In [15], a passivity-based control approach is proposed for bilateral teleoperation, which achieves stable control of a legged manipulator under time delays caused by the communication system. A reinforcement learning approach is proposed in [3] and integrated with Model Mediated Teleoperation (MMT) to mitigate the effects of long time delays.

On the other hand, various other approaches are proposed to mitigate latency considering User Datagram Protocol (UDP) and Transmission Control Protocol (TCP) for real-time applications [11,14,16,21]. In [11], a UDP-based streaming method is proposed with Forward Error Correction (FEC) for real-time video streaming applications to guarantee timeliness and low packet loss rate. In [21], a novel permutation-based encapsulation method is proposed for Internet of Things (IoT) applications where UDP and TCP are considered for application layer data encapsulation. In [14], the impacts of 5G New Radio (NR) numerologies on E2E latency are discussed, where UDP and TCP-based simulations are conducted to reveal the relationship. In [16], the performance of UDP and TCP are compared in terms of throughput, data rate, packet loss, and network latency on both IEEE-802.11p and 5G networks, where the superiority of UDP is shown in terms of latency and data rate in Vehicle-to-Vehicle (V2V) and Vehicle-to-Infrastructure (V2I) scenarios.

Considering existing literature, most of the studies either consider communication or control systems separately, and some lack E2E real-life experiments.

Therefore, the feasibility of the proposed solutions remains unclear in practical E2E teleoperation systems.

Motivated by the aforementioned issue, we develop a 5G-based teleoperation testbed with haptic feedback and conduct extensive latency and reliability experiments to investigate the current capabilities of 5G communications. With the inspiration of the information freshness concept [9], we propose a two-way timeout approach in which packets are discarded if they are not received before the timeout period to reduce the waiting time for stale information. We conduct additional experiments on the 5G network to compare conventional UDP with our proposed approach. The results of the experiments show that the proposed approach achieves better latency performance with the cost of reliability compared to benchmark UDP without a timeout. The main contributions of this study can be listed as follow.

– We design and build a 5G-enabled teleoperation testbed with haptic feedback, in which a remote operator controls a robotic arm to complete a task over 5G communications. The operator can feel the remote environment with haptic feedback enabled controller, allowing for more precise operation and increasing the sense of telepresence.
– We propose a two-way timeout approach for teleoperation systems to improve latency and reliability performance.
– We implement the proposed approach on our teleoperation testbed, where we conduct latency and reliability experiments over a 5G network.
– We present latency and packet loss measurements for the proposed approach and compare them to the benchmark protocol without a timeout. Experiment results show that the proposed approach can reduce average communication latency by up to 65%.

The rest of the paper is organized as follows. In Sect. 2, we provide system descriptions and propose a two-way timeout approach for low-latency teleoperation. In Sect. 3, we present our 5G-enabled teleoperation testbed and experiment setup. In Sect. 4, we provide our experimental results, which show the superiority of our approach, and we conclude the study in Sect. 5.

2   System Description and Two-Way Timeout Approach

In this section, we provide the system description and propose the two-way timeout approach. As seen in Fig. 1, a typical teleoperation system consists of three main domains. At the operator domain, a controller equipped with a haptic interface is used by a human operator to control a robotic arm at the teleoperator domain. To accomplish a task, the haptic controller samples the operator's inputs (i.e., control commands) and transmits them via the communication domain. In the communication domain, a server is utilized to exchange packets between the operator and teleoperator domains. In the teleoperator domain, a remote robotic arm receives control commands and provides force feedback over the communication domain. Generally, UDP is used to guarantee the required
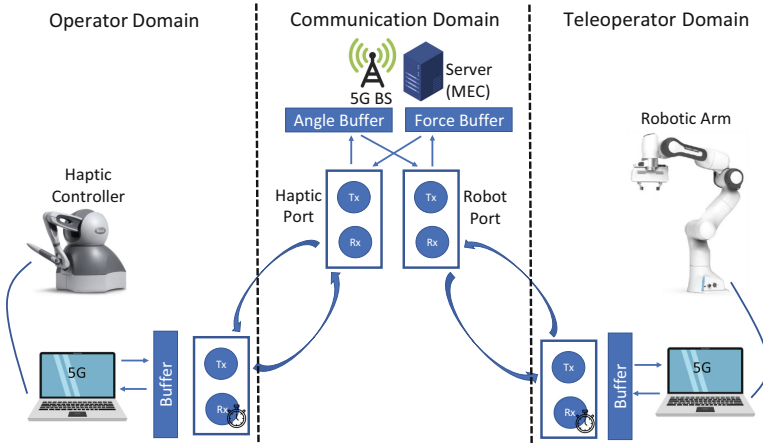
Fig. 1. 5G-enabled teleoperation system with haptic feedback

control frequency since it is more suitable than TCP for real-time applications [10].

In conventional UDP, clients request packets and wait for the response from the server. Waiting time is related to the latency and packet loss experienced in the communication domain. Since UDP is unreliable, packet losses are to be expected, which may result in communication deadlock. Deadlock can be described as the event that both the client and the server are in the receiving (Rx) state. In most cases, this is caused by packet loss in the communication domain. For example, a client transmits a packet to the server and waits for the response in the Rx state. If the transmitted packet drops, the server will also be in the Rx state, resulting in a packet exchange deadlock. In the sequel, we propose a technique to minimize communication latency and the event of a deadlock.

2.1 Two-Way Timeout Approach

As shown in Fig. 1, two clients, a haptic device and a robotic arm, communicate with the UDP server to exchange control commands and haptic feedback, respectively. In this scenario, each client sends the most recent sample to the server and waits for the most recent feedback or control command. As a result, clients can be in either the Rx or transmitting (Tx) state. We use a timeout technique at each client end (i.e., two-way timeout) to limit the waiting time in the Rx state and to mitigate the event of a deadlock. In other words, the client transmits the most recent sample and waits until the timeout threshold, τ, is reached. If no packet is received before then, it switches to the Tx state to transmit a new sample. As a result, the client is not required to wait in the Rx state for any longer than the timeout threshold τ. Please note that the timeout threshold, τ, must be adjusted according to application requirements. For exam-

ple, if the required latency for an application is 10 ms, then waiting more than 10 ms is a waste of time because any packet received after the threshold will be stale. In other words, if a packet is not received before the timeout, the receiver assumes it is lost. Fresh information can thus be retrieved without wasting time waiting for stale information.

Let us denote the communication latency from the haptic device to the server as d_{hs} and the communication latency from the server to the haptic device as d_{sh}. Similarly, denote the communication latency from the robotic arm to the server and from the server to the robotic arm as d_{rs} and d_{sr}, respectively. Then, E2E communication latency (i.e., round trip latency) can be given as

$$d_{e2e} = d_{hs} + d_{sr} + d_{rs} + d_{sh}. \tag{1}$$

The proposed timeout approach limits d_{sr}, and d_{sh} by the timeout threshold τ, i.e., $d'_{sr} \leq \tau$ and $d'_{sh} \leq \tau$. In this case, the E2E communication latency becomes

$$d'_{e2e} \leq d_{hs} + d'_{sr} + d_{rs} + d'_{sh}, \tag{2}$$
$$\leq d_{hs} + d_{rs} + 2\tau. \tag{3}$$

While the timeout strategy ensures that there is no deadlock in the communication channel, it also means that packets may be discarded or dropped if they do not arrive before the timeout expires. This implies that the proposed approach has a lower level of reliability. To account for this, we denote the reliability as γ and compute the reliability using Packet Delivery Ratio (PDR), which can be expressed as

$$\gamma = R/T, \tag{4}$$

where R is the total number of packets received by the receiver and T is the total number of packets transmitted by the transmitter. To capture the overall performance, we consider a joint Cost Function (CF), defined as the cost of successfully transmitting a packet. For given E2E latency d'_{e2e}, and reliability γ, CF can be defined as

$$CF = \frac{1}{\gamma} d'_{e2e} \ (ms), \tag{5}$$

where $\frac{1}{\gamma}$ is the number of repetitions required to transmit a packet with probability 1, and d'_{e2e} is the time required to transmit a packet.

3   5G-Enabled Teleoperation Testbed with Haptic Feedback

In this study, we develop a 5G-enabled teleoperation testbed with haptic feedback[1] to enable real-time teleoperation with a sense of touch as seen in Fig. 1. At the operator domain, 3D systems haptic device is deployed as a haptic controller,

---

[1] System description: https://youtu.be/c3onK5Vh6QE.

which has the capability of 6-degree-of-freedom (DoF) positional sensing and 3-DoF force feedback. The human operator uses the haptic device to control the remote robotic arm on the teleoperator domain via the communication domain. We deployed Franka Emika Panda robotic arm as a teleoperator which is a 7-DoF serial manipulator with 1kHz control and sensor sampling capabilities. In the sequel, we explain the flow of information in each domain and describe the measurement setup.

3.1   Operator Domain

A local computer at the operator domain samples 3D positions, $\mathbf{p} = \{p_x, p_y, p_z\}$ of the haptic device's end effector at the haptic device working space. Since the coordinate systems of the robot and haptic device are different, we investigate their local coordinate systems and match their movement axes such that the Y-axis of the haptic device corresponds to the X-axis of the robotic arm (i.e., towards up), Z-axis of the haptic device corresponds to Y-axis of the robotic arm (i.e., towards to user), and X-axis of the haptic device corresponds to Z-axis of the robotic arm (i.e., towards right). Then, positions are scaled up to match the movement space of the robotic arm since one unit movement in the haptic workspace does not correspond to one unit movement in robot space, necessitating mapping. Computed position values with fixed quaternion rotations (since we only consider position control) are then fed into the inverse kinematics (IK) solver of the robotic arm, which is based on the Denavit-Hartenberg [4] parameters of the Franka Emika Panda robotic arm [1] which can be given as follows

$$\boldsymbol{\theta} = f_{ik}\left(\mathbf{p}, \mathbf{q}\right) \tag{6}$$

where $\boldsymbol{\theta} = \{\theta_1, \theta_2, \theta_3, \theta_4, \theta_5, \theta_6, \theta_7\}$ are computed joint angles, $\mathbf{q} = \{q_w, q_x, q_y, q_z\}$ are fixed quaternion rotations, and $f_{ik}(.)$ is the inverse kinematics solver function. The same computation is performed in each sampling cycle, and the computed joint angles $\boldsymbol{\theta}$ are written into a buffer to be transmitted to the server.

3.2   Communication Domain

We deploy a UDP server in the communication domain at the 5G Base Station (BS) Mobile Edge Computing (MEC) unit. The haptic controller and the robotic arm computers are equipped with 5G dongles to access the local BS. At the UDP server, we utilize two separate ports to communicate with two clients (i.e., the haptic device computer and the robotic arm computer). In addition, we use two buffers to store the most recently received packets from the robot and haptic device. As discussed in the preceding subsection, the haptic device computer transmits the most recently computed joint angles to the server, and then the server writes them into the angle buffer as seen in Fig. 1. Most recent joint angles are transmitted to the robotic arm when a new packet request is received. New packets are requested by transmitting the most recent sample to the server. In

other words, each client transmits its most recent sample, and the server receives it and updates the corresponding buffer. In exchange, the server transmits the buffer's content to the client. If the client is the haptic device, the angle buffer is updated, and the content of the force buffer is returned. If the client is the robotic arm, the force buffer is updated, and the content of the angle buffer is returned.

3.3   Teleoperator Domain

Franka Emika Panda robotic arm and a local computer reside in the teleoperator domain. Received joint angles from the server are written into a buffer for execution. The main reason for using the buffer is that the frequency of the real-time control loop is 1 kHz, which means that the control loop must be fed with new joint angles in every 1 ms. This necessitates the arrival of a new packet in every 1 ms, which is not possible in physical implementation due to communication and computation latencies. As a result, we utilize a buffer at the robot end to serve as an information source for the control loop, allowing us to run the control loop at a frequency of 1 kHz. On the other hand, we update the buffer with the latest received joint angles whenever a new packet arrives from the server. We employ a proportional integral derivative (PID) control algorithm on a local computer which computes joint angular velocity commands for received joint angles considering the difference between received joint angles and the current state of the robotic arm. Computed commands are executed using provided functions of *libfranka*, which is a C++ implementation of Franka Control Interface (FCI) [1]. This process is repeated in each control loop to enable real-time control. In addition, the robotic arm's built-in sensors are used to sample external forces applied to the robot, which are transmitted as feedback to the server, allowing us to control the robot in real-time with haptic feedback (i.e., sense of touch).

3.4   Latency Measurement Setup

One of the main challenges in latency measurements is the unsynchronized end devices. Since the E2E latency is at a milliseconds level, ensuring that the end devices are synchronised to the same clock is critical to have reliable results. One method for synchronizing end devices is to use *Network Time Protocol (NTP)* servers, in which each device synchronizes its local clock with the same NTP server, allowing time synchronization. However, even this solution has a synchronization error of tens to hundreds of milliseconds (ms) depending on the NTP server load and other communication channel characteristics [5], which is insufficient for our experiments. To overcome this difficulty, we collect exchanged packets with timestamps at the server, allowing us to perform the measurements on the same machine and eliminate the time synchronization issue. Let us denote t_c^i as the time the i-th packet received by the server from the client c. Then, the E2E latency between the server and the client c for the i-th packet becomes

$$d_{e2e,c}(i) = t_c^{(i)} - t_c^{(i-1)}, \tag{7}$$

where $d_{e2e,c}$ is the time difference between two consecutive packets, which measures the round trip time required from server to client and client to server. Then, the mean E2E latency for received N packets from client c becomes

$$d_{mean,c} = \frac{1}{N} \sum_{i=1}^{N} d_{e2e,c}(i). \tag{8}$$

On the server side, control and feedback packets are exchanged over two dedicated ports. Furthermore, two threads are running concurrently to provide real-time communication. One thread handles server and haptic device communication. Another thread runs for server and robot communication. The server buffers the received control packets from the haptic device and transmits the most recent feedback packet back. Similarly, received feedback packets are buffered, and the server returns the most recent available control packet to the robot. Since both the control and feedback buffers have a capacity of one packet, the server always has the most recently received control and feedback packets. Furthermore, the server records packet traffic from both the haptic device and the robot. Every received packet is recorded with a timestamp, providing a chronological record of packet exchanges.
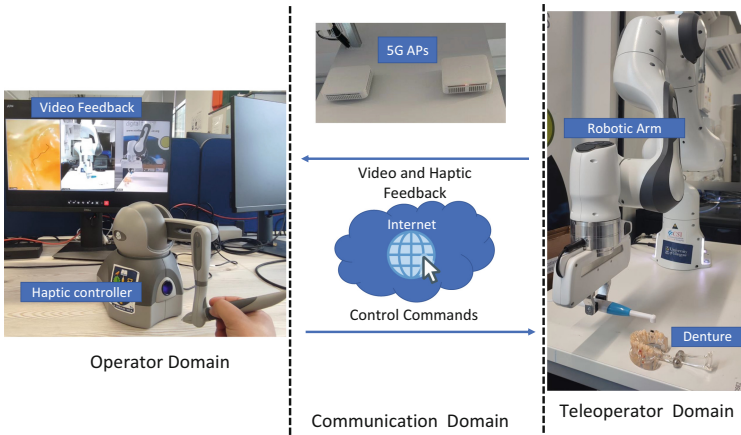

Fig. 2. Experimental setup considering remote dental inspection use case.

4   Experimental Results

In this section, we provide our experiment results considering remote dental inspection use case[2] in Fig. 2. Latency and packet loss measurements are conducted as discussed in Sect. 3-D.

---

[2] Remote dental inspection use case: https://youtu.be/afJFUwUW6Dg.

In Fig. 3a, we provide an average E2E latency comparison between the proposed approach and the benchmark UDP without a timeout. We conduct experiments for different timeout values and for the benchmark UDP, where we exchanged 10,000 packets in every experiment. As seen from the figure, the proposed approach outperforms the benchmark UDP in terms of E2E latency, where the proposed approach achieves mean E2E latency of around 5 ms while the benchmark UDP achieves mean E2E latency of around 50 ms. This is reasonable since the proposed approach reduces the waiting time in the Rx state and mitigates the long waiting time delays. On the other hand, E2E latency increases with increasing timeout period τ. This is also reasonable since the E2E latency is related to the communication latency and the timeout period.
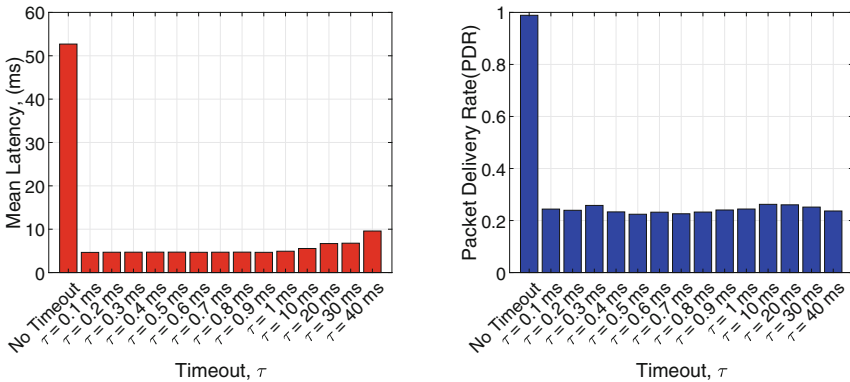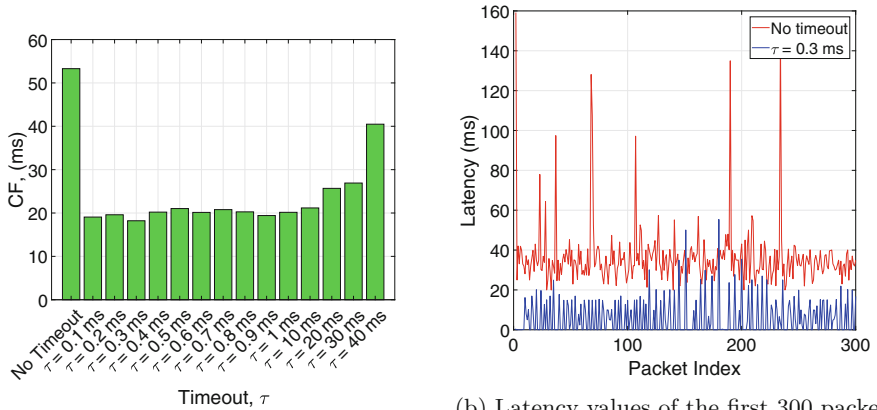

(a) Latency comparison between the proposed approach and the benchmark UDP without a timeout.

(b) Packet delivery rate comparison between the proposed approach and the benchmark UDP without a timeout.

Fig. 3. Latency and packet delivery rate comparisons.

In Fig. 3b, we provide the reliability comparison in terms of PDR. As discussed in Sect. 2, the proposed approach has a negative impact on reliability. This is because the waiting time in the Rx state is limited by the timeout period τ, which allocates very little time for packet reception. However, we must consider a joint metric CF to reveal overall performance. CF results are provided in Fig. 4a to compare the performance of the proposed approach with the benchmark UDP without a timeout. As seen from the figure, the proposed approach outperforms the benchmark UDP. In other words, the proposed approach requires less than 20 ms to successfully transmit a packet, whereas the benchmark UDP requires more than 50 ms. The results show that the proposed approach can achieve up to 65% less E2E latency under identical network conditions.

Furthermore, we provide the latency measurements in Fig. 4b for the first 300 packets to demonstrate the difference between the proposed approach and the benchmark UDP in terms of jitter, which is the deviation of the latency. The

(a) Cost function comparison between the proposed approach and the benchmark UDP without a timeout.

(b) Latency values of the first 300 packets for the proposed approach with $\tau = 0.3$ ms and the benchmark UDP without a timeout.

Fig. 4. Cost function comparison and latency values for first 300 packets.

figure shows that the proposed approach reduces jitter and provides more stable and predictable network conditions.

To further evaluate the latency deviations, we provide latency distributions in Fig. 5 for the benchmark UDP (see Fig. 5a) and the proposed approach (see Fig. 5b). As seen from the figures, the proposed approach reduces the standard deviation of latency, σ, from 14.84 to 10.28 compared to the benchmark UDP which is consistent with the results in Fig. 4b and justifies them.
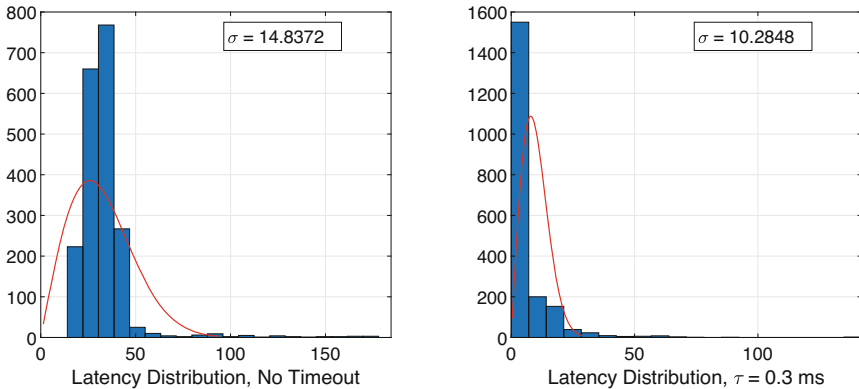
(a) Benchmark UDP without timeout

(b) Proposed approach with $\tau = 0.3$ ms

Fig. 5. Latency distributions of the proposed approach the benchmark UDP without a timeout.

5 Conclusions

In this study, we design and develop a 5G-enabled teleoperation testbed with haptic feedback, where we conduct extensive latency and reliability measurements to demonstrate the current capabilities of 5G communications. Furthermore, we propose a two-way timeout approach to maintain QoS in teleoperation systems for improved latency and reliability performance. We reduce the E2E latency by reducing waiting time in the packet reception process. The proposed approach allows for the transition between Tx and Rx states by introducing a timeout threshold for the Rx state, which improves latency and reliability performance. Experimental results are presented and compared with the benchmark UDP without a timeout. The presented results show that the proposed approach can reduce mean E2E latency by up to 65% compared to the benchmark UDP with identical communication networks.

As a future work, we will conduct more experiments to show the superiority of the proposed approach in terms of control performance, such as stability and transparency in teleoperation.

References

1. Franka control interface documentation (2017). https://frankaemika.github.io/docs/
2. Antonakoglou, K., Xu, X., Steinbach, E., Mahmoodi, T., Dohler, M.: Toward haptic communications over the 5G tactile internet. IEEE Commun. Surv. Tutor. **20**(4), 3034–3059 (2018)
3. Beik-Mohammadi, H., et al.: Model mediated teleoperation with a hand-arm exoskeleton in long time delays using reinforcement learning. In: IEEE International Conference on Robot and Human Interactive Communication (RO-MAN), pp. 713–720. IEEE (2020)
4. Denavit, J., Hartenberg, R.S.: A kinematic notation for lower-pair mechanisms based on matrices (1955)
5. Durairajan, R., Mani, S.K., Sommers, J., Barford, P.: Time's forgotten: using NTP to understand internet latency. In: ACM Workshop on Hot Topics in Networks, pp. 1–7 (2015)
6. Feizi, N., Tavakoli, M., Patel, R.V., Atashzar, S.F.: Robotics and AI for teleoperation, tele-assessment, and tele-training for surgery in the era of COVID-19: existing challenges, and future vision. Front. Robot. AI 16 (2021)
7. Girgis, A.M., Park, J., Bennis, M., Debbah, M.: Predictive control and communication co-design via two-way gaussian process regression and AOI-aware scheduling. arXiv preprint arXiv:2101.11647 (2021)
8. Hou, Z., She, C., Li, Y., Zhuo, L., Vucetic, B.: Prediction and communication co-design for ultra-reliable and low-latency communications. IEEE Trans. Wireless Commun. **19**(2), 1196–1209 (2019)
9. Kaul, S., Gruteser, M., Rai, V., Kenney, J.: Minimizing age of information in vehicular networks. In: IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks, pp. 350–358. IEEE (2011)

10. Kebria, P.M., Abdi, H., Dalvand, M.M., Khosravi, A., Nahavandi, S.: Control methods for internet-based teleoperation systems: A review. IEEE Trans. Hum.-Mach. Syst. **49**(1), 32–46 (2018)
11. Kim, S., Shin, S., Moon, J.: UDP-based extremely low latency streaming. In: IEEE Annual Consumer Communications & Networking Conference (CCNC), pp. 94–99. IEEE (2022)
12. Kizilkaya, B., Zhao, G., Sambo, Y.A., Li, L., Imran, M.A.: 5g-enabled education 4.0: Enabling technologies, challenges, and solutions. IEEE Access **9**, 166962–166969 (2021)
13. Liu, C.: Adaptive control of teleoperation systems with uncertainties: a survey. In: The 3rd International Conference on Robotics Systems and Automation Engineering (RSAE), pp. 5–9 (2021)
14. Patriciello, N., Lagen, S., Giupponi, L., Bojovic, B.: 5G new radio numerologies and their impact on the end-to-end latency. In: IEEE International Workshop on Computer Aided Modeling and Design of cCommunication Links and Networks (CAMAD), pp. 1–6. IEEE (2018)
15. Risiglione, M., Sleiman, J.P., Minniti, M.V., Cizmeci, B., Dresscher, D., Hutter, M.: Passivity-based control for haptic teleoperation of a legged manipulator in presence of time-delays. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 5276–5281. IEEE (2021)
16. Tahir, M.N., Katz, M.: ITS performance evaluation in direct short-range communication (IEEE 802.11p) and cellular network (5G) (TCP vs UDP). In: Hamid, U.Z.A., Al-Turjman, F. (eds.) Towards Connected and Autonomous Vehicle Highways. EICC, pp. 257–279. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-66042-0_10
17. Tokatli, O., et al.: Robot-assisted glovebox teleoperation for nuclear industry. Robotics **10**(3), 85 (2021)
18. Tong, X., Zhao, G., Imran, M.A., Pang, Z., Chen, Z.: Minimizing wireless resource consumption for packetized predictive control in real-time cyber physical systems. In: IEEE International Conference on Communications Workshops (ICC Workshops), pp. 1–6. IEEE (2018)
19. Wang, Z., Lam, H.K., Xiao, B., Chen, Z., Liang, B., Zhang, T.: Event-triggered prescribed-time fuzzy control for space teleoperation systems subject to multiple constraints and uncertainties. IEEE Trans. Fuzzy Syst. **29**(9), 2785–2797 (2020)
20. Xu, X., Zhang, S., Liu, Q., Steinbach, E.: QoE-driven delay-adaptive control scheme switching for time-delayed bilateral teleoperation with haptic data reduction. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 8838–8844. IEEE (2021)
21. Yang, Y., Hanzo, L.: Permutation-based TCP and UDP transmissions to improve goodput and latency in the internet of things. IEEE Internet Things J. **8**(18), 14276–14286 (2021)
22. Zhang, D., Tron, R.: Stable haptic teleoperation of UAVs via small L2 gain and control barrier functions. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 8352–8357. IEEE (2021)