Statistical Characterization of Closed-Loop Latency at the Mobile Edge

Suraj Suman, *Member, IEEE*, Federico Chiariotti, *Member, IEEE*, Čedomir Stefanović, *Senior Member, IEEE*, Strahinja Došen, *Member, IEEE*, and Petar Popovski, *Fellow, IEEE*

***Abstract*—The stringent timing and reliability requirements in mission-critical applications require a detailed statistical characterization of end-to-end latency. Teleoperation is a representative use case, in which a human operator (HO) remotely controls a robot by exchanging command and feedback signals. We present a framework to analyze the latency of a closed-loop teleoperation system consisting of three entities: an HO, a robot located in remote environment, and a Base Station (BS) with Mobile edge Computing (MEC) capabilities. A model of each component is used to analyze the closed-loop latency and optimize the compression strategy. The closed-form expression of the distribution of the closed-loop latency is difficult to estimate, such that suitable upper and lower bounds are obtained. We formulate a non-convex optimization problem to minimize the closed-loop latency. Using the obtained upper and lower bound on the closed-loop latency, a computationally efficient procedure to optimize the closed-loop latency is presented. The simulation results reveal that compression of sensing data is not always beneficial, while system design based on average performance leads to under-provisioning and may cause performance degradation. The applicability of the proposed analysis is much wider than teleoperation, including a large class of systems whose latency budget consists of many components.**

***Index Terms*—Mission-critical communications, teleoperation, real-time systems, telerobotics, human-machine interaction, mobile edge computing, low-latency high-reliability**

I. Introduction

The Tactile Internet (TI) [1] is a fairly recent concept that involves the transmission of tactile sensations along with data, text, and multimedia content. The ability to receive multiple sensory inputs enhances the immersion of the user in Virtual Reality (VR), and improves control performance in teleoperation, in which a human operator (HO) controls and manipulates a remotely located robot or object [2]. Specifically, teleoperation involves a two-way exchange of data: commands from the operator and sensory feedback from the remote environment, creating a closed-loop system [3]. Advanced Human-to-Machine Interaction (HMI) in teleoperation involves the exchange of abundant sensory data, which ensures that the HO can have an intuitive and precise interaction with the remote environment, improving the task execution accuracy and efficiency, but also putting a significant strain on the communication system.

Due to its interactive nature, teleoperation is highly sensitive to communication impairments. The concept of motion-to-photon delay [4], often used in VR, is extremely relevant here: if we measure the closed-loop latency between the moment an action is performed by the operator and the moment that they get the related feedback, we can gauge the Quality of Experience (QoE) for the operator, as well as the final control performance. Besides low latency, teleoperation requires a high reliability, as communication channel losses in both directions may degrade feedback and control precision. Moreover, reliable low-latency systems are effectively *transparent* [5], i.e., the operator should not notice the remote nature of the environment, and should feel as if they were controlling the robot directly. This ensures the immersion of the operator in the remote environment, improving both QoE and control performance. On the other hand, long delays and high jitter can both deteriorate the user experience and jeopardize the stability of the closed-loop teleoperation system, as the HO's reactions to events in the remote environment are delayed and sluggish [6]. The comfort zone for performing remote surgery requires a closed-loop latency up to around 300 ms [7], while slower tasks in VR-based haptic teleoperation can tolerate up to around 800 ms[8].

Furthermore, communication is not the only bottleneck in the system, as closed-loop latency also includes computation and processing times, and the robot might have limited onboard computation capabilities. As the decoding and execution of the operator's commands are part of the teleoperation control loop along with the compression and processing of the sensing data, these operations can have a significant effect on the latency. In effect, the success of teleoperation applications strongly depends on the performance of both the communication and the computational segments. In order to fully optimize the teleoperation system to meet the final application's closed-loop latency constraints, which can be very stringent in mission-critical industrial scenarios, we need to consider all steps of the process.

Latency in networked, closed-loop control systems is not easy to control, as the randomness of the wireless channel and the variable amount of available bandwidth and computation resources make the closed-loop latency a stochastic quantity. While the effects of a higher latency and jitter are understood from laboratory experiments [9]–[11], existing schemes optimize only for the *average* latency [12]–[14], without providing

This work is supported by the European Horizon 2020 project TACTILITY (grant agreement number 856718). The work of Petar Popovski is supported by the Villum Investigator grant "WATER" from the Velux Foundation, Denmark.

Suraj Suman (ssu@iitp.ac.in) is with the Department of Electrical Engineering, IIT Patna, Bihar, India. Federico Chiariotti (chiariot@dei.unipd.it) is with Department of Information Engineering, University of Padova, Italy. Čedomir Stefanović (cs@es.aau.dk) and Petar Popovski (petarp@es.aau.dk) are with Department of Electronic Systems, Aalborg University, Denmark. Strahinja Došen (sdosen@hst.aau.dk) is with Department of Health Science and Technology, Aalborg University, Denmark.

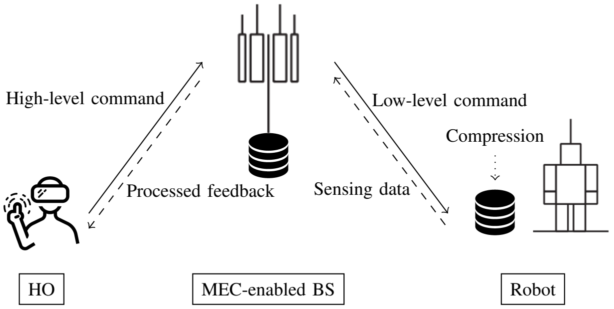


Figure 1: System model for a closed-loop MEC-enabled teleoperation system.

any reliability guarantees. Furthermore, most teleoperation system designs [15]–[18] do not take into account the computational delay when computing latency, effectively leaving out part of the control loop and potentially disregarding an important component of the closed-loop latency.

In this work, we present an analysis of a Mobile Edge Computing (MEC) [19], [20] system for teleoperation, in which the most computationally-intensive tasks are offloaded to an MEC-enabled Base Station (BS), as shown in Fig. 1. A preliminary version of this work was presented in [21], focusing only on the latency of an uplink connection, rather than closed-loop latency, from a multi-sensor robot to the BS. This work generalizes the framework to analyze the latency of the closed-loop teleoperation system and formulates an optimization problem to minimize the closed-loop latency with high reliability under statistical constraints. The key contributions of the paper are as follows:

1) The system model for closed-loop MEC-enabled teleoperation system is presented, where the command and sensing data are exchanged over wireless network through a BS. An MEC server on the BS processes the command and sensing data in order to be sent to the robot and the feedback for the HO, respectively, enabling the closed-loop operation.
2) Two system design possibilities are considered and compared. In the first one, the robot located in the remote environment compresses the raw sensing data and then transmits it to the BS, which decompresses and processes it to extract the user readable feedback that is transmitted back to the HO. In the second scenario, the robot transmits the raw sensing data to the BS without compression, where it is processed by the MEC server.
3) The closed-loop latency for both scenarios is analyzed by dividing it into three components: the duration of the compression operation on the sensing data, the transmission delays for commands and feedback, and the decompression and computation at the MEC server. All these latency components are modelled as independent random variables (RVs). Thus, the closed-loop latency, which is the sum of all these components, is also an RV. We characterize the nature of its Probability Density Function (PDF) and Cumulative Distribution Function (CDF) and obtain tractable upper and lower bounds on the closed-loop latency for both scenarios.
4) A non-convex optimization problem is formulated, aiming to minimize the the closed-loop latency in the statistical sense. Using the obtained upper and lower bounds on the closed-loop latency, we present a computationally efficient procedure to estimate the optimal closed-loop latency and solve the problem.
5) We analyze the performance of the schemes by simulation, and find that the simulation results reveal that compression of sensing data is not always beneficial. The decision on data compression depends on the communication system parameters, as well as the computational capability of the robot. The proposed approach is also compared with the system design that is optimized in average sense, as reported in the prior works. The comparative analysis reveals that system design in average sense leads to under-provisioning and causes a significant performance degradation.

While this work focuses on the latency analysis for teleoperation, the basic framework we propose can be used for any cascaded system with random latency components, such as Open Radio Access Network (O-RAN) [22] systems. The O-RAN architecture is envisioned to execute networking processes in software, making network components' behavior programmable. Telecom operators will use the standardized interfaces to control multi-vendor infrastructures. In the context of O-RAN, the proposed framework will be very useful to analyze the latency incurred across multiple software and hardware components from multiple vendors in order to maximize user QoE.

The rest of this paper is organized as follows. Section II presents related work on the subject, while the basic model of an MEC-enabled teleoperation system is presented in Section III. The different delay components of the closed-loop latency and their distributions are discussed in Section IV, and the overall closed-loop latency distribution is estimated and optimized in Section V. Finally, the numerical simulation results are discussed in Section VI, followed by the concluding remarks in Section VII.

II. Related Work

Latency is a major issue in teleoperation systems, and the literature in the field [9]–[11] has extensively investigated its impact on control performance. The study in [9] describes an experiment that uses a haptic device to generate feedback, presenting the visual three-dimensional environment to the user on a monitor and studying the effect of latency between the participant's actual action and the visible movement on the monitor. A commercial haptic teleoperation system [23] was used in [10], which allowed HOs to touch and grasp the computer-generated virtual objects. This experiment demonstrated that the average latency increases significantly with the network load. A closed-loop compensatory tracking task is performed using tactile input in [24], where the feedback is encoded to the user using frequency and amplitude modulation schemes. A significant time delay, on the order of several hundred milliseconds, has been noticed in this experiment.

Most existing low-latency teleoperation schemes have tried to minimize the *average* latency, which is an easier target, and neglected the required reliability targets in a statistical sense. Even in systems with high-reliability fiber-wireless (FiWi) networks, existing optimization works are limited to average guarantees [14]. In this scenario, the limiting factors are the availability, skill set, distance to task location, and remaining energy of robots [25], or the association between tasks and HOs [26]. The study in [27] presents a task allocation strategy by combining suitable host robot selection and computation task offloading onto collaborative nodes in the FiWi infrastructure. The conventional Cloud, decentralized cloudlets, and neighboring robots as collaborative nodes are used for computation offloading. Cross-layer techniques for low-latency teleoperation have also been considered in the literature [28]–[30]. TI cross-layer transmission optimization is investigated in [28] by considering the transmission delay, error probability, and statistical queuing delay requirements, using a proactive packet dropping mechanism to limit latency. A resource allocation mechanism to maximize the uplink sum rate of traditional data while satisfying the delay requirements for tactile data is presented in [30] using sparse code multiple access. The study in [29] estimates the average latency from a hub to an access point for tactile body-worn devices connected using an IEEE 802.11 network.

In general, support for teleoperation applications based on Network Function Virtualization (NFV) is included in the 5G network architecture [12], [13], [31]. The study in [12] presents a utility function based model to evaluate the performance of the NFV-based TI by considering the human perception resolution and the network cost of completing services. The utility function depends on the average round-trip delay, network link bandwidth, and node virtual resource consumption. The joint radio and NFV resources for a heterogeneous network are allocated in [13] by guaranteeing average end-to-end delay of each tactile user, including the queuing, transmission, and computation delays. MEC offloading is another possibility for TI applications [16], [18]. A trade-off between the average service response time and power usage efficiency is investigated in [15] for local and cooperative MEC. This can be optimized according to QoE metrics as well [16], or using more advanced caching techniques [17]. Finally, a real-time network architecture for remote surgery application is presented in [18], employing cloud and MEC networks to satisfy the timing constraints, which are still expressed in terms of the average end-to-end delay.

Overall, existing experimental works have mostly considered simple links with controllable latency, while the existing architectures only dealt with average latency, and often disregarded the contribution of the computational component of the closed-loop latency. The main novelty of our work is in the modeling and consideration of these factors, estimating the complete distribution of the closed-loop latency in a public Internet setup, along with optimizing it for the worst-case scenario by considering statistical guarantees to ensure a more stable performance than average latency minimization.

III. System Model

The basic schematic of the considered MEC-enabled teleoperation system is shown in Fig. 1: the robot, which is located in the remote environment, is instructed by the HO, who receives feedback information that guides his decisions. The instruction sent to the robot is referred to as the *command signal*, whereas the the information received from the robot is referred to as the *sensing data*. The command signal from the HO and the sensing data from the robot are exchanged over a wireless connection via the BS. The volume of the generated sensing data is much larger than the command signal, because it can include throughput-intensive formats such as video and tactile data, along with other types of media such as audio, images, text, or numeric values. On the other hand, the command signal is usually selected from a discrete and limited set of instructions, such as direction, speed, trajectory, or applied force, which are not data intensive and can be represented with only a few bytes. The sensing data acts as feedback to the HO for further command instructions to the remote robot. The transmission of this potentially large volume of data over a wireless connection is expensive in terms of required radio resources, and may require compression before the data is transmitted.

The alternative is to process the sensing data at the robot itself and extract a user-readable feedback signal, but this may be very computationally intensive, often far beyond the capabilities of the robot. On the other hand, commands from the HO are typically not compressed, and represent high-level instructions to the robot. The MEC server then translates these high-level commands to low-level commands to the robot's actuators, which can be directly executed. As the size of command signals is typically much smaller than the sensing data from the robot, the compression of command data is not within the scope of our work[1].

The MEC server at the BS acts as a decision-support system that handles the data-intensive computation task by processing the sensing data from the robot. Such processed data is communicated to the HO. In the system model shown in Fig. 1, the robot compresses the sensing data locally and transmits this compressed data to the BS. The MEC server at the BS first decompresses the data, processes it, and sends the processed data having user readable feedback to the HO. Based on the received feedback, the HO decides on the command signal for the robot located in the remote environment. Thus, the command and sensing signals exchanged over wireless medium through a BS form a closed-loop teleoperation system connected over two communication links.

Remark 1. *The focus of this work is on communication and computation aspects of the teleoperation system. Therefore, the latency due to executing the command signal at the robot is not taken into account, as it depends on the mechanical properties of the robot and the application at hand. Likewise, the latency incurred in the HO's reaction is not considered and is beyond the scope of the work.*

[1] Note that the analytical framework presented here can be straightforwardly extended if the command data also gets compressed at the HO side before transmission.

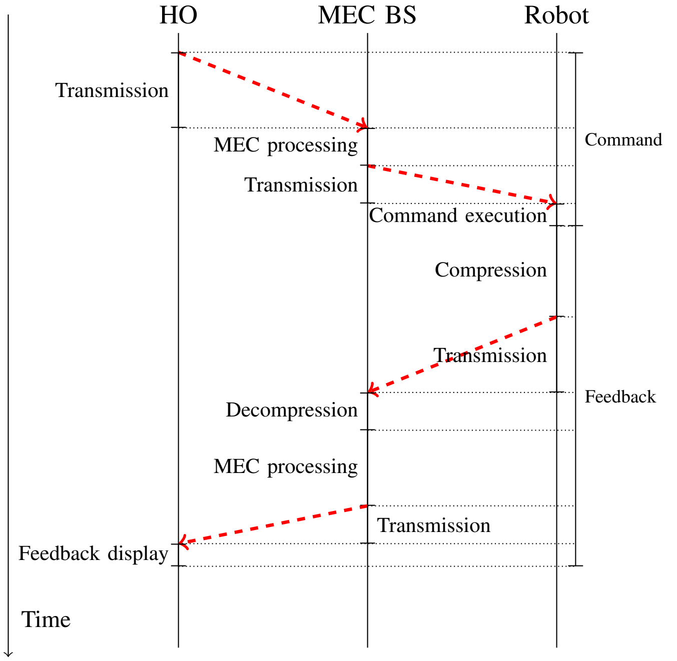


Figure 2: Depiction of the components of the closed-loop latency.

Remark 2. *In the following, we assume that the data related to teleoperation application is allocated resources in a private slice by the BS avoiding queuing delay, as its latency-constrained nature requires network support.*

The different delay components of the closed-loop latency, which constitute the delay incurred in exchanging command and sensing signals, are shown in Fig. 2 for the system model in Fig. 1. The feed-forward delay consists of the delay incurred in the transmission of the high-level command signal, the MEC processing (if required), and the transmission of processed data (i.e., the low-level command). The feedback delay consists of the delay incurred in (potential) compression of the sensing data at the robot, the transmission of the compressed data, the decompression and processing at the MEC BS, and finally, the transmission of this processed user-readable feedback signal to the HO [2].

IV. Latency Components

In the given context of an MEC-enabled teleoperation system, the closed-loop latency is mainly due to the delay incurred in transmission of command and sensing data, compression at the robot, and decompression and computation at the MEC server. The data transmission delay is random in nature due to uncertainty in the wireless channel, while the randomness of the computation time at the MEC server is related to the uncertainty in the amount of resources allocated for processing. Thus, the overall closed-loop latency is also an RV. In the following, we characterize the probability distributions of different delay components of the closed-loop MEC-enabled teleoperation system.

A. Latency Incurred in Data Transmission

We consider a wireless channel with Rayleigh block fading, such that the channel gain h is constant over the length of the packet. Hence, the fading gain $g = |h|^2 \sim \exp(1)$, and the gain over subsequent packets is independent and identically distributed. We consider a transmitter that uses power P_{tx} located at a distance d from the receiver. As teleoperation applications are extremely sensitive to delay and require dedicated resources, we consider a slicing-enabled 5G system in which a bandwidth B is reserved for the transmission [32]. The signal-to-noise ratio (SNR) γ at the receiver is then

$$\gamma(P_{\text{tx}}, d, B) = K_0 \frac{P_{\text{tx}}|h|^2}{d^\ell N_0 B} = \gamma_0(P_{\text{tx}}, d, B) g, \tag{1}$$

where K_0 is the Friis equation parameter, ℓ is a path loss exponent depending on the propagation scenario, N_0 is the noise power spectral density, and $\gamma_0(P_{\text{tx}}, d, B) = K_0 \frac{P_{\text{tx}}}{d^\ell N_0 B}$ is the average SNR.

We assume that the transmitter can choose the transmission rate, guaranteeing ε-outage of the communication link at the receiver [33] by using the Shannon bound. The outage probability ε characterizes the probability of packet loss in case of deep fading, when the transmission cannot be decoded. We assume that the data is correctly received if the instantaneous received SNR is higher than γ_{th}. Thus, for a threshold SNR γ_{th} with outage probability ε, the rate $R(\varepsilon)$ is

$$R(\varepsilon) = B \log_2(1 + \gamma_{\text{th}}). \tag{2}$$

The outage probability ε in a Rayleigh fading channel is:

$$\varepsilon = \Pr\{\gamma < \gamma_{\text{th}}\} = \Pr\left\{ g < \frac{\gamma_{\text{th}}}{\gamma_0} \right\} = 1 - \exp\left(-\frac{\gamma_{\text{th}}}{\gamma_0}\right), \tag{3}$$

where $\Pr\{\cdot\}$ denotes the probability of an event. From (2) and (3), $R(\varepsilon)$ can be rewritten as

$$R(\varepsilon) = B \log_2(1 + \gamma_{\text{th}}) = B \log_2\left(1 + \gamma_0 \ln\left(\frac{1}{1-\varepsilon}\right)\right). \tag{4}$$

Remark 3. *$R(\varepsilon)$ is a monotonically increasing function of ε.*

If the transmitter divides data into packets with a constant length n_p and uses a pass-band modulation, the time t_p to transmit a packet is simply given by

$$t_p = \frac{n_p}{2BR(\varepsilon)}. \tag{5}$$

As the erasure probability for a packet is ε, the total time until correct reception is a geometrically distributed RV. The time elapsed in receiving acknowledgement is very small compared to the data transmission time, and hence it is ignored. Thus, the probability mass function (PMF) of the time $\mathcal{T}$ required to transmit a packet is then given by

$$\Pr(\mathcal{T} = kt_p) = \varepsilon^{k-1}(1-\varepsilon), \quad k \geq 1. \tag{6}$$

[2] The propagation delay is not taken into consideration here, as it will be several orders of magnitude lower than the other components of the closed-loop delay for the distances relevant for the considered teleoperation scenario.

The mean and variance of $\mathcal{T}$ are then

$$\mathbb{E}[\mathcal{T}] = \frac{1}{1-\varepsilon}t_p, \;\; \mathbb{E}[(\mathcal{T} - \mathbb{E}[\mathcal{T}])^2] = \frac{\varepsilon}{(1-\varepsilon)^2}t_p^2. \quad (7)$$

If a data block is composed of N packets, the total transmission time for the block is:

$$T_{\text{tx}}(N, \varepsilon) = \sum_{i=1}^{N} \mathcal{T}_i. \quad (8)$$

We note that T_{tx} is the sum of N identical and independent geometrically distributed RVs. It's PMF is a negative binomial distribution as follows

$$\Pr(T_{\text{tx}} = kt_p | N, \varepsilon) = \binom{k+N-1}{N-1}\varepsilon^{k-N}(1-\varepsilon)^N, \;\; k \geq N. \quad (9)$$

In most practical cases, N will be relatively large, and we can use the Central Limit Theorem to approximate this distribution to a Gaussian RV as follows:

$$T_{\text{tx}}(N, \varepsilon) \sim \mathcal{N}(\mu_{\text{tx}}, \sigma^2_{\text{tx}}), \quad (10)$$

where $\mu_{\text{tx}} = N\frac{1}{1-\varepsilon}t_p$ and $\sigma^2_{\text{tx}} = N\frac{\varepsilon}{(1-\varepsilon)^2}t_p^2$. This approximation substitutes the transmission time from a discrete domain with only positive values with the real domain. However, the approximation error is negligible for $N \gg 1$. The Cumulative Distribution Function (CDF) of T_{tx} is given as follows:

$$F_{T_{\text{tx}}}(t) = \frac{1}{2}\left[1 + \text{erf}\left(\frac{t - \mu_{\text{tx}}}{\sqrt{2}\sigma_{\text{tx}}}\right)\right], \quad (11)$$

where $\text{erf}(x) = \frac{2}{\sqrt{\pi}}\int_0^x \exp(-z^2)dz$ is the error function.

Remark 4. *The transmission time, modeled here using the Gaussian distribution, should be non-negative, but the domain of the Gaussian distribution is* $(-\infty, \infty)$*. Since we are interested in the worst-case scenario, i.e., in cases in which the latency is higher than average, the extremely small probability of the Gaussian approximation resulting in a negative value on the left tail of the distribution (we have noted that* $\mu_{tx} - 4\sigma_{tx} \geq 0$ *for the numerical values considered in this paper, implying that more than 99.995% data points will be non-negative) has a negligible effect on the analysis and the considered reliability levels.*

Remark 5. *The PDF of Gaussian distribution is neither a convex nor a concave function. It is symmetric and exhibits a unimodal variation. Further, it is also a log-concave function.*

B. Latency Incurred in Data Processing

The time required by computational tasks, including compression and decompression of data, depend upon the number of central processing unit (CPU) cycles required to process one bit of data, the clock frequency of the CPU, and the volume of the data to be processed. In the context of the considered closed-loop teleoperation system (see Fig. 1), the processing capability of the MEC-enabled BS will be much higher than the robot's [34]. Therefore, the computation to extract the low-level command from the raw data is performed by the BS rather than at the robot. Here, we model the time elapsed in different computational processes, which will be necessary to characterize the closed-loop latency.

1) Latency incurred in computation: The time elapsed in computation T_c to compute a volume of data D_0 is given as

$$T_c = \frac{D_0 X_c}{f_0}, \quad (12)$$

where X_c is the number of CPU cycles and f_0 is the frequency of the CPU clock.

A recent study [35] shows that the number of cycles allocated to compute one bit is stochastic in nature. This is because the CPU cycles are allocated to different ongoing tasks simultaneously. The number of CPU cycles required to compute one bit of data is modeled in the relevant literature [36], [37] as an RV following a Gamma distribution. Thus, the PDF of $X_c \sim \text{Gamma}(\kappa_1, \beta_1)$ is given as

$$f_X(x; \kappa_1, \beta_1) = \frac{x^{\kappa_1 - 1}}{(\beta_1)^{\kappa}_1\Gamma(\kappa_1)}\exp(-x/\beta_1), \quad (13)$$

where κ_1 is the shape parameter and β_1 is the scale parameter. $\Gamma(s) = \int_0^\infty t^{s-1}exp(-t)dt$ is the Gamma function. Note that $\mathbb{E}[X_c] = \kappa_1\beta_1$. Now, from (12), (13), and the transformation of the PDF of X_c, the distribution of the computation time $T_c \sim \text{Gamma}\left(\kappa_1, \frac{D_0\beta_1}{f_0}\right)$ is given by

$$f_{T_c}(t; \kappa_1, \beta_1, D_0, f_0) = \left(\frac{f_0}{D_0\beta_1}\right)^{\kappa_1}\frac{t^{\kappa_1 - 1}}{\Gamma(\kappa_1)}\exp\left(\frac{-tf_0}{D_0\beta_1}\right). \quad (14)$$

The expected computation delay $\bar{T}_c$ is obtained as follows:

$$\bar{T}_c(\kappa_1, \beta_1, D_0, f_0) = \mathbb{E}[T_c] = \frac{D_0\kappa_1\beta_1}{f_0}. \quad (15)$$

The CDF of T_c is given by:

$$F_{T_c}(t) = \frac{\Gamma\left(\kappa_1, \frac{t}{D_0\beta_1/f_0}\right)}{\Gamma(\kappa_1)}, \quad (16)$$

where $\Gamma(s, x) = \int_0^x t^{s-1}exp(-t)dt$ is lower incomplete Gamma function. It may be noted that the lower incomplete gamma function is usually denoted by $\gamma(s, x)$. However, use of this notation would introduce ambiguity with the notation used in the paper to denote SNR – see (1).

2) Latency incurred in compression: The latency of data compression depends on the data volume and computational properties of the device's processor. Specifically, T_{cp}, the time elapsed in compressing volume of data D_0 is given as [38]

$$T_{\text{cp}} = \frac{D_0 X_{\text{cp}}}{f_0}, \quad (17)$$

where X_{cp} is the number of CPU cycles required to compress one bit of data, and f_0 is the frequency (i.e., clock speed) of the processor. Analogously to the previous case, X_{cp} is stochastic in nature and follows the Gamma distribution given

$$X_{\text{cp}} \sim \text{Gamma}(\kappa_2, \beta_2), \quad (18)$$

where κ_2 and β_2 are respectively the shape and scale parameters. Note that $\mathbb{E}[X_{\text{cp}}] = \kappa_2\beta_2$.

Thus, T_{cp} is also an RV and its PDF is given as

$$T_{\text{cp}} \sim \text{Gamma}\left(\kappa_1, \frac{D_0\beta_2}{f_0}\right). \quad (19)$$

We consider lossless data compression, so that the original data can be perfectly reconstructed from the compressed data without error[3] [40]. For the lossless compression, the average number of CPU cycles required to compress one bit of raw data is given as [38], [41]

$$\mathbb{E}[X_\text{cp}] = \kappa_2\beta_2 = \exp(Q\psi) - \exp(\psi) = C(Q), \tag{20}$$

where $Q \geq 1$ is the compression ratio (i.e., the ratio of the sizes of raw and compressed data) and ψ is a positive constant. Using (20), the PDF of compression time T_cp is also a Gamma RV

$$T_\text{cp} \sim \text{Gamma}\left(\kappa_2, \frac{D_0 C(Q)}{\kappa_2 f_0}\right). \tag{21}$$

The expected value of T_cp for the compression ratio Q is given as

$$\bar{T}_\text{cp}(\kappa_2, \beta_2, D_0, f_0, Q) = \mathbb{E}[T_\text{cp}] = \frac{D_0\mathbb{E}[X_\text{cp}]}{f_0} = \frac{D_0 C(Q)}{f_0}. \tag{22}$$

3) Latency incurred in decompression: Decompression refers to the process of restoring compressed data to its original form. It is also a type of computation, and can be performed on MEC server. The latency incurred T_d in decompressing D_0 amount of data is given as

$$T_d = \frac{D_0 X_d}{f_0}, \tag{23}$$

where X_d denotes the number of cycles required to decompress one bit of data, which will also follow the Gamma distribution given as

$$X_d \sim \text{Gamma}(\kappa_3, \beta_3), \tag{24}$$

where κ_3 and β_3 are the shape and scale parameters, respectively. Note that $\mathbb{E}[X_d] = \kappa_3\beta_3$.

Recent works in [39], [42], [43] show that the decompression process is faster than compression for the same volume of data. Thus, the average number of cycles required in decompression and compression process is:

$$\mathbb{E}[X_d] = \zeta\mathbb{E}[X_\text{cp}], \tag{25}$$

where $0 < \zeta < 1$ is a constant. Using (25), the following can be written:

$$\kappa_3\beta_3 = \zeta\kappa_2\beta_2 = \zeta C(Q). \tag{26}$$

Thus, the decompression time T_d is also an RV, $T_\text{d} \sim \text{Gamma}\left(\kappa_3, \frac{D_0\zeta C(Q)}{f_0\kappa_3}\right)$, and its PDF with compression ratio Q is given as

$$T_\text{d} \sim \text{Gamma}\left(\kappa_3, \frac{D_0\zeta C(Q)}{f_0\kappa_3}\right). \tag{27}$$

The expected value of T_d, $\bar{T}_d$, for the compression ratio Q, $\bar{T}_d$, is obtained as

$$\bar{T}_d(\kappa_3, \beta_3, D_0, f_0, Q) = \mathbb{E}[T_d] = \frac{D_0\kappa_3\beta_3}{f_0} = \frac{D_0\zeta C(Q)}{f_0}. \tag{28}$$

Remark 6. *The PDF of the Gamma distribution is neither convex nor concave, but it has unimodal variation. Also, it is not a symmetric distribution [44]. Thus, the PDFs of T_c, T_{cp}, and T_d are neither convex nor concave functions, but are all unimodal. In addition, the Gamma PDF is log-concave, and hence so are the PDFs of T_c, T_{cp}, and T_d.*

V. Closed-Loop Latency Analysis

Using the results from the previous section, we can develop the analytical framework to estimate the closed-loop latency of MEC-enabled teleoperation system shown in Fig. 1. We assume that the HO transmits all the command data to the BS for processing at the MEC server. On the other hand, two scenarios are analyzed regarding the processing of the raw sensing data at the robot. In the first case, the robot located in the remote environment compresses the raw sensing data first and then transmits these compressed data. The MEC server then decompresses the data to recover the original version, which is processed to extract the user readable command to be transmitted to the HO. In the second case, the robot does not compress the sensing data, but transmits them in raw form to the BS for further processing at the MEC server to extract the user readable command for the HO.

Let D_c and D_s be the volume of command and sensing signal, respectively. Let the distance between the HO and the BS be d_{ho}, and the same between BS and the robot be d_r. The transmission powers of the HO, the BS, and the robot are P_tx^{ho}, P_tx^{bs}, and P_tx^{r}, respectively. We also assume that a bandwidth B is dedicated for this closed-loop operation, and the HO, the BS, and the robot transmit over this bandwidth. We now consider the case in which the sensing data is compressed by the robot to avoid excessive transmission delays, while the BS decompresses and processes the data to generate a human-readable feedback signal.

The shape parameter in (13), (18), and (24)) will remain the same for all tasks performed by the same processor. On the other hand, the scale parameter will be different for different tasks (computation, compression, or decompression), as different amounts of resources in terms of CPU cycles need to be allocated. Let the shape parameter of the MEC-enabled BS be κ_MEC, and the scale parameter for computation and decompression at the BS be β_c and β_d, respectively. Further, let the shape parameter of the robot's embedded processor be κ_r, and the scale parameter for the compression process be β_cp. Finally, let the frequency of the MEC-enabled BS and robot be f_MEC and f_R, respectively. Thus, from (20) and (26), we get

$$\kappa_r\beta_\text{cp} = C(Q), \quad \kappa_\text{MEC}\beta_d = \zeta C(Q). \tag{29}$$

A. Case 1: Data Compression at Robot

The closed-loop latency is the sum of the latency of the command data (from the HO to the robot) and the sensing data (from the robot to the HO). The latency of the command data is the time required to transmit the HO's command, extract the low-level command on the MEC-enabled BS, and transmit the low-level command from the BS to the robot. Thus, referring

[3] Huffman, run-length, Lempel-Ziv, and bzip2 are some of the most commonly used compression techniques to achieve lossless data compression [39].

to Fig. 1, the latency involved in transmitting the command signal T_1^c composed of N_c data packets with outage probability ε is given as

$$T_1^c = T_{\text{tx}}^c(N_c, \varepsilon) + T_c^c(\kappa_{\text{MEC}}, \beta_c, D_c, f_{\text{BS}}) + T_{\text{tx}}^{pc}(N_c^p, \varepsilon), \quad (30)$$

where T_{tx}^c is the time elapsed in transmitting the command signal as given by (10) and T_c^c is the time taken by MEC server to estimate the low-level command, given by (14). T_{tx}^{pc} denotes the time elapsed in transmitting the low-level command (consisting of N_c^p packets) extracted from command signal as in (10). All the constituents of T_1^c are independent RVs.

The latency of the sensing data is the time required by compression on the robot, transmission of the compressed data, decompression the compressed data by the MEC-enabled BS, extraction of the low-level command from the sensing data, and transmission of the human-readable feedback to the HO. Thus, referring to Fig. 1, the feedback latency T_1^f when N_f data packets are to be transmitted with outage ε is given as

$$\begin{aligned} T_1^f =& T_c^f(\kappa_{\text{MEC}}, \beta_c, D_s, f_{\text{BS}}, Q) + T_d^f(\kappa_{\text{MEC}}, \beta_d, \frac{D_s}{Q}, f_{\text{BS}}, Q), \\ &+ T_{\text{cp}}^f(\kappa_r, \beta_{\text{cp}}, D_s, f_R, Q) + T_{\text{tx}}^f(N_f, \varepsilon) + T_{\text{tx}}^{pf}(N_f^p, \varepsilon), \end{aligned} \quad (31)$$

where T_r denotes the time elapsed in compressing the raw sensing data with compression ratio Q, given by (21), T_{tx}^f denotes the transmission delay of the compressed data, given by (10), T_d^f denotes the decompression delay for the compressed sensing data, given by (27), and T_c^f denotes the processing time for the sensing data, given by (19). T_{tx}^{pf} is the transmission time of the human-readable feedback (consisting of N_f^p packets) extracted from the sensing data to the HO, given by (10). All the constituents of T_1^f are independent RVs. The volumes of the low-level command and human-readable feedback are much lower than for the original raw data, i.e., $N_c >> N_c^p$ and $N_f >> N_f^p$, and will not contribute significantly in the closed-loop latency. Using this fact, T_1^c and T_1^f can be written as,

$$\begin{aligned} T_1^c \approx& T_{\text{tx}}^c(N_c, \varepsilon) + T_c^c(\kappa_{\text{MEC}}, \beta_c, D_c, f_{\text{BS}}) \\ T_1^f \approx& T_d^f(\kappa_{\text{MEC}}, \beta_d, \frac{D_s}{Q}, f_{\text{BS}}, Q) + T_c^f(\kappa_{\text{MEC}}, \beta_c, D_s, f_{\text{BS}}, Q) \\ &+ T_{\text{cp}}^f(\kappa_r, \beta_{\text{cp}}, D_s, f_R, Q) + T_{\text{tx}}^f(N_f, \varepsilon). \end{aligned} \quad (32)$$

Using (32), we can estimate the closed-loop latency T_1 as

$$\begin{aligned} T_1 =& T_{\text{cp}}^f(\kappa_r, \beta_{\text{cp}}, D_s, f_R, Q) + T_d^f(\kappa_{\text{MEC}}, \beta_d, \frac{D_s}{Q}, f_{\text{BS}}, Q) \\ &+ T_{\text{tx}}^c(N_c, \varepsilon) + T_c^c(\kappa_{\text{MEC}}, \beta_c, D_c, f_{\text{BS}}) + T_{\text{tx}}^f(N_f, \varepsilon) \\ &+ T_c^f(\kappa_{\text{MEC}}, \beta_c, D_s, f_{\text{BS}}, Q). \end{aligned} \quad (33)$$

This can be rewritten as

$$\begin{aligned} T_1 =& T_{\text{tx}}^c(N_c, \varepsilon) + T_{\text{tx}}^f(N_f, \varepsilon) + T_{\text{cp}}^f(\kappa_r, \beta_{\text{cp}}, D_s, f_R, Q) \\ &+ T_c^c(\kappa_{\text{MEC}}, \beta_c, D_c, f_{\text{BS}}) + T_d^f(\kappa_{\text{MEC}}, \beta_d, \frac{D_s}{Q}, f_{\text{BS}}, Q) \\ &+ T_c^f(\kappa_{\text{MEC}}, \beta_c, D_s, f_{\text{BS}}, Q). \end{aligned} \quad (34)$$

The expected value of T_1, μ_{T_1}, is given by

$$\begin{aligned} \mu_{T_1} =& \frac{(N_c + N_f)t_p}{1-\varepsilon} + \frac{(D_c + D_s)\kappa_{\text{MEC}}\beta_c}{f_{\text{BS}}} + \frac{D_s C(Q)}{f_R} \\ &+ \frac{\zeta D_s C(Q)}{Q f_{\text{BS}}}. \end{aligned} \quad (35)$$

T_1's constituent distributions T_c^c, T_d^f, and T_c^f follow Gamma distributions with different scale and shape parameters, whereas T_{tx}^c and T_{tx}^f follow Gaussian distributions. It is very difficult to estimate the closed-form expression of the PDF of the RV T_1. Therefore, it is very important to characterize its properties for further analysis.

Lemma 1. *The PDF of the sum of two independent RVs is convex if and only if at least one of them is convex. In the same way, the PDF of the sum of two independent RVs is concave if and only if at least one of them is concave.*

Proof. See Appendix A. □

Remark 7. *The PDF of T_1 is neither a convex nor a concave function of t, because none of its constituent distributions are either convex or concave (see Lemma 1).*

Theorem 1. *The CDF of T_1 is neither a convex nor a concave function.*

Proof. See Appendix B. □

B. Case 2: Raw Data Offloading to MEC

In this case, no data compression happens at the robot, and the whole raw sensing data is transmitted to the BS, which processes it in order to extract the human-readable feedback. As the only difference with Case 1 is in the feedback latency, the latency T_2^c to transmit the command signal from HO to the robot is simply given by

$$T_2^c = T_1^c.$$

Now, referring to Fig. 1, the latency T_2^f involved in transmitting M_f raw sensing data packets from the robot to the HO is given as

$$T_2^f = T_{\text{tx}}^f(M_f, \varepsilon) + T_c^f(\kappa_{\text{MEC}}, \beta_c, D_s, f_{\text{BS}}) + T_{\text{tx}}^{pf}(N_f^p, \varepsilon), \quad (36)$$

where the details of the parameters are mentioned in (31).

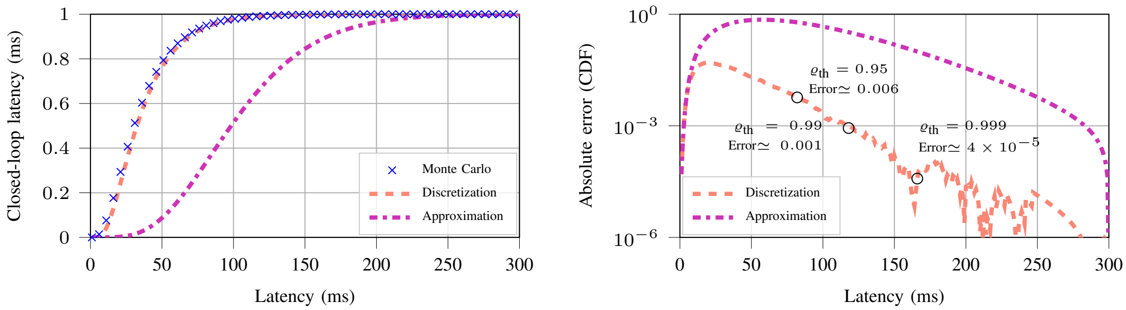
Ignoring the latency to send the low-level commands to robot and the feedback to the HO (see (32)), the closed-loop latency T_2 is given as

$$\begin{aligned} T_2 =& T_2^c + T_2^f \\ =& T_{\text{tx}}^c(N_c, \varepsilon) + T_c^c(\kappa_{\text{MEC}}, \beta_c, D_c, f_{\text{BS}}) + T_{\text{tx}}^f(N_f, \varepsilon) \\ &+ T_c^f(\kappa_{\text{MEC}}, \beta_c, D_s, f_{\text{BS}}) + T_c^c(\kappa_{\text{MEC}}, \beta_c, D_c, f_{\text{BS}}) \\ &+ T_c^f(\kappa_{\text{MEC}}, \beta_c, D_s, f_{\text{BS}}) + T_{\text{tx}}^c(N_c, \varepsilon) + T_{\text{tx}}^f(M_f, \varepsilon). \end{aligned} \quad (37)$$

The expected value of T_2, μ_{T_2}, is given as

$$\mu_{T_2} = \frac{(N_c + N_f)t_p}{1-\varepsilon} + \frac{D_c \kappa_{\text{MEC}} \beta_c}{f_{\text{BS}}} + \frac{D_s \kappa_{\text{MEC}} \beta_c}{f_{\text{BS}}}. \quad (38)$$

As the constituents of T_2 are all independent RVs, we can make some of the same inferences that we proved for T_1.



(a) CDF.

(b) Absolute error in the CDF approximation.

Figure 3: Variation of the CDF and the absolute error of the sum of four independent Gamma RVs with $D_s = 0.5$Mb, $f_{\text{BS}} = 15$GHz, $f_R = 5$GHz, $Q = 1.2$, and $\varepsilon_{\text{th}} = 10^{-4}$.

Remark 8. *The PDF of T_2 is neither a convex nor a concave function of time, because none of its constituent distributions are either convex or concave (see Lemma 1).*

Theorem 2. *The CDF of T_2 is neither a convex nor a concave function.*

Proof. See Appendix C. □

C. Optimization of the Closed-loop Teleoperation System

The closed-loop latency τ_i (where $i = 1$ for Case 1 and $i = 2$ for Case 2) of the MEC-enabled teleoperation system can be optimized by finding the optimal compression ratio used by the robot before transmitting its sensing data to the BS. For this purpose, an optimization problem is formulated as follows

$$\begin{aligned} \textbf{(P1)}: \quad & \min_{Q,\varepsilon} \quad \tau_i, \quad i \in \{1,2\} \\ \text{s. t.} \quad \textbf{(C1)}: \quad & F_{T_i}(\tau_i) \geq \varrho_{\text{th}}, \quad i \in \{1,2\} \\ \textbf{(C2)}: \quad & 0 \leq \varepsilon \leq \varepsilon_{\text{th}}, \\ \textbf{(C3)}: \quad & Q \geq Q_{\text{th}} = 1. \end{aligned}$$

Constraint **(C1)** ensures the closed-loop latency τ in probabilistic sense, where $F_T(\cdot)$ denotes the CDF of T. ϱ_{th} is an statistical parameter, which indicates the probability of the closed-loop latency be at most τ. Constraint **(C2)** limits the link outage probability, and constraint **(C3)** ensures that compression reduces the size of the data.

In order to solve the optimization problem **(P1)**, we need to obtain the distributions of T_1 and T_2 and verify that they meet the statistical guarantee on the closed-loop latency criterion **(C1)**. However, as noted above, the PDFs of T_1 and T_2 are hard to obtain, because the closed-form expression of the distribution of the sum of arbitrary Gamma RVs is not known. There are several approximation methods to estimate the distribution of sum of Gamma RVs reported in the literature [45]–[47], but they are only accurate when the number of RVs to be summed are very high. Here, the closed-loop latency T_1 is the sum of only four Gamma RVs (see (34)), whereas the closed-loop latency T_2 is the sum of only two Gamma RVs (see (37)). The approximation error may then be high, which is unacceptable for a teleoperation system in time-critical application scenarios. Therefore, these approximation methods are not viable options in the given context.

On the other hand, the PDF of the sum of independent RVs can be obtained by convolving the constituent PDFs. However, the continuous convolution is difficult to compute, since it requires the solution of a complicated multiple integral. Therefore, the convolution in the discrete time domain is adopted here as a simplifying approximation. Towards this, the constituent PDFs are discretized by sampling with the same sampling interval (here 1 millisecond is considered) in order to obtain the corresponding probability mass functions (PMFs). Then, the constituent PMFs are convolved to obtain the PMF of the closed-loop latency. Effectively, we use integration by rectangles as a numerical technique to solve the optimization problem **(P1)**, approximating the integral by its midpoint Riemann sum, which has negligible error due to the very small step size. Hence, the CDF obtained from the discretized convolution is very accurate, and the trade-off between accuracy and computational complexity can be tuned depending on the reliability criterion.

This can also be observed from the CDF of the sum of four independent Gamma RVs, as shown in Fig. 3a for three different methods. The first method is Monte Carlo simulation, which is extremely accurate due to the huge number of samples. The distribution of the sum of independent Gamma RVs is also approximated using both the method from [45]–[47] and the discretized convolution. One can observe from Fig. 3b that the CDF obtained by discretization method varies very closely with the CDF obtained by Monte Carlo simulation method. On the other hand, the CDF obtained by approximation method is very far from that obtained from Monte Carlo simulation. This justifies the choice of discretization method followed by convolution. The value of the absolute error is around respectively $0.006, 0.001$, and 4×10^{-5} for $\rho_{\text{th}} = 0.95, 0.99$, and 0.999, which is at least an order of magnitude lower than $1 - \rho_{\text{th}}$. Thus, the effects of the approximation error can effectively be neglected when using the discretized convolution.

In the next step, the optimal values of the latency, link outage ε, and compression ratio Q are found by exhaustive search. A computationally efficient way to simplify the task by reducing the search space of ε and Q is to observe that, as the value of ε_{th} increases, the transmission time decreases (see Remark 3), and hence $\varepsilon = \varepsilon_{\text{th}}$ will be the optimal value. The problem is then reduced to finding the optimal value of the compression ratio Q. Finding lower and upper bounds on the values of T_1 and T_2 will be helpful to solve the optimization problem **(P1)** efficiently.

Theorem 3. *The closed-loop latency T_1 for a given ϱ_{th} with $F_{T_1}(\tau_1) = \varrho_{th}$ is bounded as follows:*

$$\tau_{1,L}(Q, \varepsilon, \varrho_{th}) \leq \tau_1 \leq \tau_{1,U}(Q, \varepsilon, \varrho_{th}).$$

The lower bound is given by:

$$\begin{aligned}\tau_{1,L}(Q, \varepsilon, \varrho_{th}) = \max \big(&F^{-1}_{T^c_{tx}}(\varrho_{th}), F^{-1}_{T^f_{tx}}(\varrho_{th}), F^{-1}_{T^f_{cp}}(\varrho_{th}),\\ &F^{-1}_{T^c_c}(\varrho_{th}), F^{-1}_{T^f_d}(\varrho_{th}), F^{-1}_{T^f_c}(\varrho_{th})\big),\end{aligned}$$

where $F^{-1}_{\mathcal{Z}}(\cdot)$ denotes the inverse of CDF of RV $\mathcal{Z}$. The upper bound is given by:

$$\begin{aligned}\tau_{1,U}(Q, \varepsilon, \varrho_{th}) = \min \Bigg(&F^{-1}_{T^c_{tx}}(\varrho_{th}) + F^{-1}_{T^f_{tx}}(\varrho_{th}) + F^{-1}_{T^f_c}(\varrho_{th})\\ &+F^{-1}_{T^c_c}(\varrho_{th}) + F^{-1}_{T^f_d}(\varrho_{th}) + F^{-1}_{T^f_{cp}}(\varrho_{th}), \frac{\mu_{T_1}}{1-\varrho_{th}}\Bigg).\end{aligned}$$

Proof. See Appendix D. □

Theorem 4. *The closed-loop latency T_2 for a given ϱ_{th} with $F_{T_2}(\tau_2) = \varrho_{th}$ is bounded as follows:*

$$\tau_{2,L}(\varepsilon, \varrho_{th}) \leq \tau_2 \leq \tau_{2,U}(\varepsilon, \varrho_{th}),$$

where the two bounds are given by

$$\begin{aligned}\tau_{2,L}(\varepsilon, \varrho_{th}) &= \max\Big(F^{-1}_{T^c_{tx}}(\varrho_{th}), F^{-1}_{T^f_{tx}}(\varrho_{th}), F^{-1}_{T^c_c}(\varrho_{th}), F^{-1}_{T^f_c}(\varrho_{th})\Big)\\ \tau_{2,U}(\varepsilon, \varrho_{th}) &= \min\Bigg(F^{-1}_{T^c_{tx}}(\varrho_{th}) + F^{-1}_{T^f_{tx}}(\varrho_{th}) + F^{-1}_{T^c_c}(\varrho_{th})\\ &\qquad + F^{-1}_{T^f_c}(\varrho_{th}), \frac{\mu_{T_2}}{1-\varrho_{th}}\Bigg).\end{aligned}$$

Proof. See Appendix E. □

Using the bounds on the closed-loop latency, the following can be written

$$\tau^{\text{opt}}_{i,L} \leq \tau^{\text{opt}}_{i} \leq \tau^{\text{opt}}_{i,U}, \quad i \in \{1, 2\}, \tag{39}$$

where $\tau^{\text{opt}}_{i,L} = \underset{Q,\varepsilon_{\text{th}},\varrho_{\text{th}}}{\operatorname{argmin}} \ \tau_{i,L}$ and $\tau^{\text{opt}}_{i,U} = \underset{Q,\varepsilon_{\text{th}},\varrho_{\text{th}}}{\operatorname{argmin}} \ \tau_{i,U}$. It is important to determine $\tau^{\text{opt}}_{i,U}$, as it will be used subsequently. Denote the inverse of the error function $(\text{erf}^{-1}(\cdot))$ and inverse of lower incomplete Gamma function $(\Gamma^{-1}(\cdot))$ at ϱ_{th} as:

$$\begin{aligned}&\text{erf}^{-1}(2\varrho_{\text{th}} - 1) = \phi_0 > 0 \text{ (as } \varrho_{\text{th}} \to 1),\\ &\Gamma^{-1}(\varrho_{\text{th}}, \kappa_r) = \phi_1 > 0, \Gamma^{-1}(\varrho_{\text{th}}, \kappa_{\text{MEC}}) = \phi_2 > 0.\end{aligned} \tag{40}$$

Now, the inverse of the CDF of the RVs used in the expressions for lower and upper bound of T_1 is obtained as follows

$$\begin{aligned}F^{-1}_{T^c_{\text{tx}}}(\varrho_{\text{th}}) &= N_c \frac{1}{1-\varepsilon} t_p + \phi_0 \sqrt{N_c \frac{\varepsilon}{(1-\varepsilon)^2} t_p^2},\\ F^{-1}_{T^c_{\text{tx}}}(\varrho_{\text{th}}) &= N_f \frac{1}{1-\varepsilon} t_p + \phi_0 \sqrt{N_f \frac{\varepsilon}{(1-\varepsilon)^2} t_p^2}\\ F^{-1}_{T^f_{\text{cp}}}(\varrho_{\text{th}}) &= \frac{D_s C(Q) \phi_1}{\kappa_r f_R}, F^{-1}_{T^c_c}(\varrho_{\text{th}}) = \frac{D_c \beta_c \phi_2}{f_{\text{BS}}},\\ F^{-1}_{T^f_d}(\varrho_{\text{th}}) &= \frac{D_s \zeta C(Q) \phi_2}{Q \kappa_{\text{MEC}} f_{\text{BS}}}, F^{-1}_{T^f_c}(\varrho_{\text{th}}) = \frac{D_s \beta_c \phi_2}{f_{\text{BS}}}.\end{aligned} \tag{41}$$

The upper bound on the closed-loop latency can be written as follows:

$$\tau_{1,U}(Q, \varepsilon_{\text{th}}, \varrho_{\text{th}}) = \min(\mathcal{J}_1(Q), \mathcal{J}_2(Q)), \tag{42}$$

where we have:

$$\begin{aligned}\mathcal{J}_1(Q) &= \frac{t_p \left(N_c + N_f + \phi_0 \sqrt{(N_c + N_f)\varepsilon_{\text{th}}}\right)}{1 - \varepsilon_{\text{th}}} + \frac{D_s C(Q) \phi_1}{\kappa_r f_R}\\ &\quad + \frac{(D_s + D_c)\beta_c \phi_2}{f_{\text{BS}}} + \frac{D_s \zeta C(Q) \phi_2}{Q \kappa_{\text{MEC}} f_{\text{BS}}}\\ \mathcal{J}_2(Q) &= \frac{\frac{(N_c+N_f)t_p}{1-\varepsilon_{\text{th}}} + \frac{(D_s+D_c)\kappa_{\text{MEC}}\beta_c}{f_{\text{BS}}} + \frac{D_s C(Q)}{f_R} + \frac{\zeta D_s C(Q)}{Q f_{\text{BS}}}}{1 - \varrho_{\text{th}}}.\end{aligned} \tag{43}$$

Remark 9. *$\mathcal{J}_1(Q)$ and $\mathcal{J}_2(Q)$ are convex functions of Q. The proof is not included for brevity.*

Thus, the optimal value of $\tau_{1,U}(Q, \varepsilon_{\text{th}}, \varrho_{\text{th}})$ can be obtained as follows

$$\tau^{\text{opt}}_{1,U} = \min \left(\min_Q \mathcal{J}_1(Q), \min_Q \mathcal{J}_2(Q) \right).$$

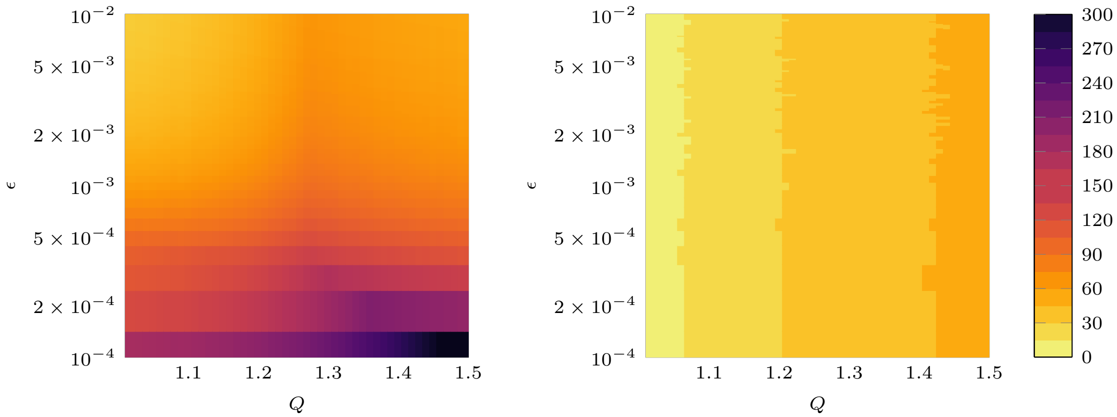
where $\min_Q \mathcal{J}_1(Q)$ and $\min_Q \mathcal{J}_2(Q)$ are straightforward to obtain due to their nature. The reduced search interval of compression ratio $\mathcal{Q}_i$ can be obtained as follows

$$\mathcal{Q}_i = \{Q \mid \tau_{i,L} \leq \tau^{\text{opt}}_{i,U}\}. \tag{44}$$

$\mathcal{Q}_i$ is segmented into equidistant intervals (specifically, we used the interval length of 0.01), and then the PMF of the closed-loop latency is computed for each of the points by convolving constituent PMFs. Finally, the optimal value of Q is the one which offers the minimum closed-loop latency by satisfying constraint **(C1)** of optimization problem **(P1)**.

VI. Simulation Results

We illustrate the analysis presented in the previous sections through numerical evaluations. If not stated otherwise, the values of the parameters considered are: $\ell = 2$, $D_c = 0.15$ Mb, $D_s = 0.5$ Mb, $f_{\text{BS}} = 15$ GHz, $\kappa_{\text{MEC}} = 1.25$, $\kappa_r = 1.5$, $\zeta = 0.1$, $\Psi = 3.5$, $B = 10$ MHz, $T_0 = 0.5$ μs, $K_0 = -27$ dB, $N_0 = -110$ dB, $d_{r-bs} = d_{bs-ho} = 2$ km, $P^{ho}_{\text{tx}} = P^{r}_{\text{tx}} = 0.5$ W. The computational capabilities of the MEC-enabled BS are consistent with the Nvidia Jetson TX1, a common embedded processor for edge computing applications [48].



(a) Lower bound ($\tau_1^{opt} - \tau_{1,L}$ (ms)).

(b) Upper bound ($\tau_{1,U} - \tau_1^{opt}$ (ms)).

Figure 4: Validation of the bounds on closed-loop latency for Case 1 with $f_R = 1$ GHz, $\rho_{\text{th}} = 0.95$.

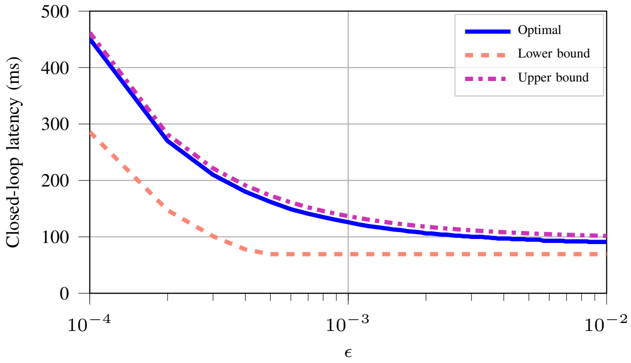


Figure 5: Validation of the bounds on closed-loop latency for Case 2 with $\rho_{\text{th}} = 0.95$.

A. Validation of Bounds

The validation of the bounds on the closed-loop latency obtained for Case 1 in Theorem 3 is shown in Fig. 4. Fig. 4a validates the lower bound as the difference between the optimal closed-loop latency and its lower bound, i.e., $\tau_1^{opt} - \tau_{1,L}$, is always positive. Similarly, the difference between the upper bound on the closed-loop latency and its optimal value, i.e., $\tau_{1,U} - \tau_1^{opt}$, is also always positive, as shown in Fig. 4b.[4] The bounds on the closed-loop latency obtained for Case 2 in Theorem 4 are shown in Fig. 5, demonstrating that the optimal value of closed-loop latency lies well within the bounds (the compression ratio here is 1, because sensing data is not compressed at the robot located in the remote environment). From Fig. 4 and Fig. 5, it can be noted that the optimal closed-loop latency in both cases is very close to the upper bound.

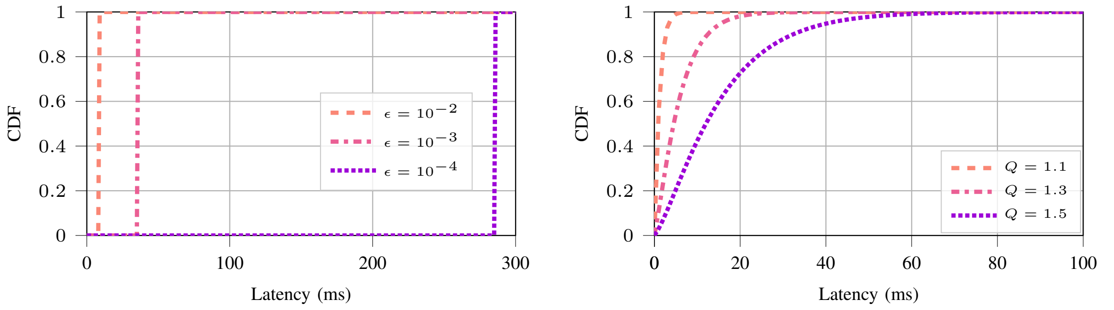
B. Optimal System Design

The CDF of the latency incurred in data transmission and compression is shown in Fig. 6. The CDF of the latency incurred in transmitting 0.5 Mb of sensing data is shown in Fig. 6a for different levels of link outage. It may be noted that the transmission time increases significantly as the target link outage level becomes stringent. This indicates that the data transmission with high level of accuracy demands relatively higher transmission times. The CDF of the latency incurred in compressing 0.5 Mb of sensing data is shown in Fig. 6b for different compression ratios Q with $f_R = 5$ GHz. The compression time increases significantly with the increase in the compression ratio. However, a higher compression ratio reduces the volume of sensing data, also reducing the transmission latency, and vice versa. Thus, there is a trade-off between compression and transmission times.

The optimal closed-loop latency as a function of the outage ε is shown in Fig. 7a for both cases. The optimal latency is high for very low values of the outage requirement and it decreases as the outage probability increases: higher outage probabilities will increase the data rate, which compensates the penalty for the increased number of retransmissions. The computational capability of the robot also has a significant impact on the latency, and the optimal value of the closed-loop latency decreases as the computational capability of the robot increases. It may be noted that the optimal latency for Case 1 is much lower than that compared to Case 2 for low target outage requirements. However, the optimal latency converges towards Case 2 as the outage increases even for the higher computational capability of the robot: as the data rate increases, compression rather than transmission becomes the most expensive task, and it becomes convenient to transmit the raw data. The optimal compression ratio against outage probability is shown in Fig. 7b, which depends upon the computational capability of the robot as well as the outage probability. The optimal compression ratio decreases as the outage probability increases, and increases with the processor speed of the robot, as the task becomes less time-consuming with respect to data transmission. As the outage probability increases, the optimal compression ratio converges towards 1, i.e., no compression as in Case 2.

Remark 10. *Data compression is not always beneficial. The decision about whether to compress the sensing data or not*

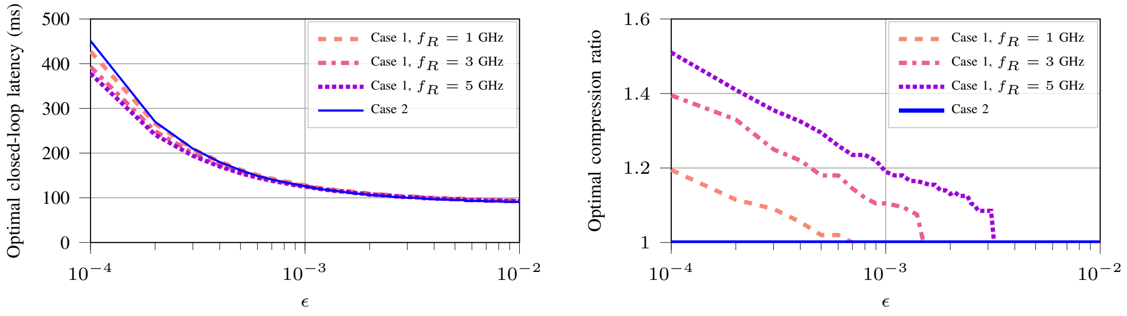
[4] We note that same behavior is observed for any arbitrary range of Q and ε.



(a) Transmission latency.

(b) Compression latency.

Figure 6: CDF of the latency incurred in data transmission and compression.



(a) Closed-loop latency.

(b) Optimal compression ratio.

Figure 7: Variation of the optimal closed-loop latency and compression ratio for different cases with $\rho_{\text{th}} = 0.95$.

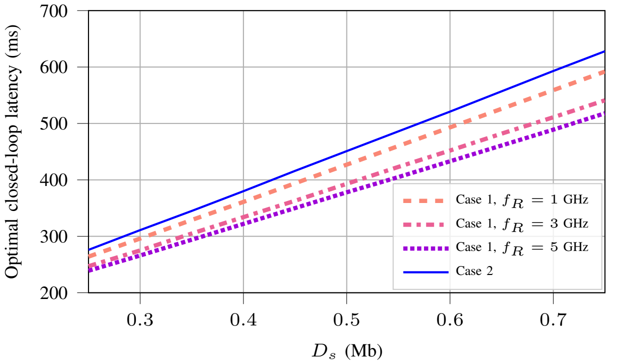


Figure 8: Variation of the optimal closed-loop latency as a function of the data volume D_s for different cases with $\rho_{\text{th}} = 0.95$ and $\varepsilon_{\text{th}} = 10^{-4}$.

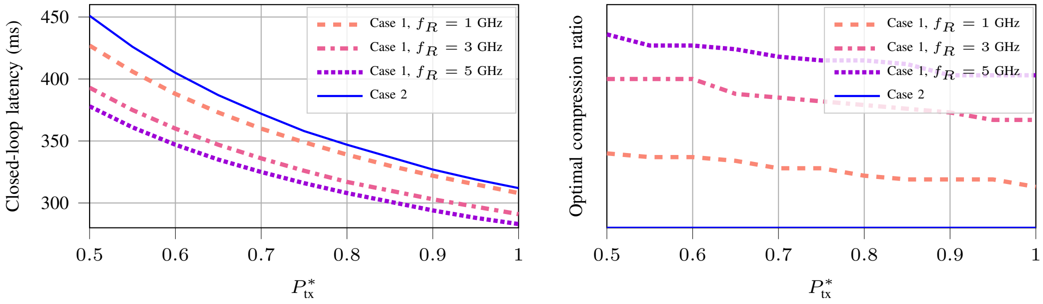
depends on the required outage constraint, as well as the computational capabilities of the robot.

Fig. 8 shows the optimal closed-loop latency as a function of the volume of sensing data for both cases, with $\rho_{\text{th}} = 0.95$ and $\varepsilon_{\text{th}} = 10^{-4}$. The optimal latency increases linearly with the volume of sensing data, as it has a linear effect on both the transmission and compression time. As noted above, the computational capabilities of the robot have a strong impact on the optimal latency, which it decreases significantly as the computational capability of the robot increases. Due to the nature of the optimization problem considered in the paper, and to the linearity of both transmission and compression time with respect to the data volume, the optimal compression ratio depends only on the computational capability of the robot and does not depend on the volume of sensing data. The optimal compression ratio is 1.19, 1.39, and 1.48 for $f_R = 1$GHz, $f_R = 3$GHz, and $f_R = 5$GHz, respectively. It may be noted that the optimal latency for Case 2 is always higher than for Case 1, and the difference increases as the computational capability of robot increases.

We can also consider the robot's transmission power as a parameter: Fig. 9a shows the closed-loop latency as a function of P_{tx} with $\rho_{\text{th}} = 0.95$ and $\varepsilon_{\text{th}} = 10^{-4}$. Increasing the transmission power leads to a higher transmission rate, reducing the transmission latency without increasing other components of the overall closed-loop latency. Fig. 9b shows that the optimal compression ratio also decreases when the robot transmits at a higher power: this is because the higher data rate changes the trade-off between the compression and transmission latency, making it more convenient to transmit more data rather than spend time compressing them.

C. Statistical vs Expected Sense System Design

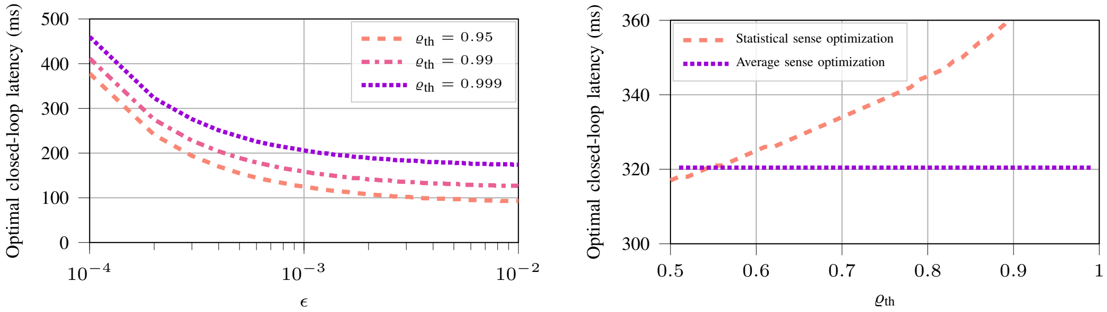
The works reported in [12]–[14], [25]–[27] consider a similar system design in expected or average sense rather than in a stochastic sense. Here, we perform a comparative analysis of the optimal closed-loop latency, considering the reliability criterion in the stochastic sense and the one in the average sense for Case 1. Similar inferences can be made also for Case 2, which is omitted due to space constraints.



(a) Closed-loop latency.

(b) Optimal compression ratio.

Figure 9: Variation of the optimal closed-loop latency and compression ratio against different transmit power level by robot for different cases with $\rho_{\text{th}} = 0.95$ and $\varepsilon_{\text{th}} = 10^{-4}$.



(a) Latency as a function of ε.

(b) Latency as a function of ϱ_{th}.

Figure 10: Performance comparison with average sense design for Case 1 with $f_R = 5$ GHz.

The closed-loop latency in the average sense is optimized using the average latency obtained in (35) from the following optimization problem:

$$\textbf{(P2)}: \min_{Q,\varepsilon} \quad \mu_{T_1}, \quad \text{s. t. } \textbf{(C2)} \text{ and } \textbf{(C3)}.$$

The average sense design in **(P2)** does not take into account the statistical guarantee, as the average closed-loop latency is optimized. **(P2)** is a convex optimization problem that can be solved using traditional solvers. The proof of convexity of **(P2)** is omitted for brevity.

The variation of the optimal closed-loop latency as a function of link outage is shown in Fig. 10a for the different stochastic reliability levels of $\varrho_{\text{th}} = 0.95, 0.99, 0.999$. It may be noted that the optimal latency increases significantly with the increase in ϱ_{th}. The comparative view of statistical and average sense design is depicted in Fig. 10b for $\varepsilon = 10^{-4}$ and $f_R = 5$ GHz. It may be noted that the latency obtained by the average sense design will satisfy the statistical guarantee on the closed-loop latency around $\varrho_{\text{th}} = 0.54$ only (i.e., this value of the latency will be exceeded 46% of the time), which may not acceptable for low-latency and high reliability applications in real-life deployment scenarios.

Remark 11. *The system design in average sense leads to under-provisioning and a potential performance degradation, which may have severe consequences in ultra-low latency and high-reliability applications.*

VII. Concluding Remarks

In this paper, we introduced a framework to analyze the closed-loop latency of a teleoperation system, where the command data from the HO and sensing data from the robot are exchanged over a wireless connection through a BS with MEC capabilities. The high-level command from the HO and the sensing data from the robot are processed by the BS to extract the low-level command for the robot and the feedback signal, respectively, which are then sent to the robot and HO. We have analyzed the closed-loop latency, which is found to be a sum of several RVs, obtained upper and lower bounds to its distribution, and formulated an optimization problem to control the transmission rate and compression ratio and provide statistical closed-loop latency guarantees. We then investigated different trade-offs in the achievable performance in terms of latency, link outage, and transmission reliability: the decision on whether and how much to compress mostly depends on the computational capability of the robot and link outage probability. The optimal latency increases with the volume of sensing data, but the optimal compression ratio depends only on the computational capability of the robot and does not depend on the volume of sensing data. We have also observed the shortcomings of the design approaches that only consider the expected value of the latency.

Future directions for further work include investigations of the closed-loop teleoperation system from an information freshness perspective. Another potential direction is the consideration of heterogeneous sensing data having different compression profiles and their impact on the QoE. Finally, the design of adaptive power control mechanisms at the robot and HO sides is another interesting extension of our work, as the devices may be energy-constrained, which calls for an energy-aware optimization of the closed-loop latency.

Appendix

A. Proof of Lemma 1

Let $\mathcal{W}$ be the distribution of the sum of two independent RVs $\mathcal{U}$ and $\mathcal{V}$, so that $f_{\mathcal{W}}(t) = \int_{-\infty}^{\infty} f_{\mathcal{V}}(\tau) f_{\mathcal{U}}(t-\tau) d\tau$. Now, let $f_{\mathcal{U}}(t)$ be a convex function of t. We then have $f_{\mathcal{U}}((1-\theta)t_1 + \theta t_2) \leq (1-\theta) f_{\mathcal{U}}(t_1) + \theta f_{\mathcal{U}}(t_2), 0 \leq \theta \leq 1$. Now, $f_{\mathcal{W}}((1-\theta)t_1 + \theta t_2)$ is given as

$$f_{\mathcal{W}}((1-\theta)t_1 + \theta t_2) = \int\limits_{-\infty}^{\infty} f_{\mathcal{V}}(\tau) f_{\mathcal{U}}((1-\theta)t_1 + \theta t_2 - \tau) d\tau.$$

Using the convexity of $f_{\mathcal{U}}(t)$, $f_{\mathcal{W}}((1-\theta)t_1 + \theta t_2)$ can be written as:

$$\begin{aligned} f_{\mathcal{W}}((1-\theta)t_1 + \theta t_2) \leq (1-\theta) &\int_{-\infty}^{\infty} f_{\mathcal{V}}(\tau) f_{\mathcal{U}}(t_1 - \tau) d\tau \\ + \theta &\int_{-\infty}^{\infty} f_{\mathcal{V}}(\tau) f_{\mathcal{U}}(t_2 - \tau) d\tau \\ = (1-\theta) & f_{\mathcal{W}}(t_1) + \theta f_{\mathcal{W}}(t_2). \end{aligned}$$

Hence, $f_{\mathcal{W}}(t)$ is a convex function of t. The concavity property can be proven in the same way.

B. Proof of Theorem 1

Let $F_{T_1}(t)$ denote the CDF of T_1, which is given as: $F_{T_1}(t) = \int_{-\infty}^{t} f_{T_1}(x) dx$, where $f_{T_1}(\cdot)$ denotes the PDF of T_1. The second derivative of $F_{T_1}(t)$ is $\frac{d^2}{dt^2} F_{T_1}(t) = \frac{d}{dt} f_{T_1}(t) = f'_{T_1}(t)$. For a function to be convex (concave), its second derivative should be non-negative (non-positive). Thus, the convexity or concavity of $F_{T_1}(t)$ depends on the sign of $f'_{T_1}(t)$. We know that $f_{T_1}(t)$ is neither a convex nor a concave function of t (see Remark 7), but this does not imply anything on the sign of $f'_{T_1}(t)$.

The RV T_1 given in (34) is the sum of six RVs, whose PDFs' properties are listed in Table I. The PDF of the sum multiple of RVs is the convolution of their PDFs. The works in [49], [50] investigated the nature of the convolution of two functions with following key observations: firstly, the convolution of two log-concave functions is also a log-concave function; secondly, the convolution of a log-concave function and a unimodal function is also a unimodal function; thirdly, the convolution of two asymmetric unimodal functions is a multi-modal function. Based on these findings and the nature of the constituent PDFs of T_1 as listed in Table I, one can deduce that $f_{T_1}(t)$ is a multi-modal function of t. Thus, the first derivative of $f_{T_1}(t) = f'_{T_1}(t)$ changes its sign multiple times in the domain of definition. Hence, $F_{T_1}(t)$ is neither a convex nor a concave function of t.

Table I: Nature of different constituent distributions of closed-loop latency T_1 (see (34))

	Unimodal	**Symmetric**	**Log-concave**
$T_{\text{tx}}^{\text{c}}, T_{\text{tx}}^{f}$ (cf. Remark 5)	Yes	Yes	Yes
T_{c}^{c} (cf. Remark 6)	Yes	No	Yes
T_{d}^{f} (cf. Remark 6)	Yes	No	Yes
T_{c}^{f} (cf. Remark 6)	Yes	No	Yes
T_{cp}^{f} (cf. Remark 6)	Yes	No	Yes

C. Proof of Theorem 2

This can also be proven in the same way as Theorem 1, following the steps in Appendix B.

D. Proof of Theorem 3

In order to prove the theorem, we first show that $\tau_{1,L}$ is a lower bound of τ_1. We define two RVs X and Y with differentiable strictly monotonic CDFs $F_X(x)$ and $F_Y(y)$, respectively, and their quantiles $x_0 = F_X^{-1}(\varrho_{\text{th}})$ and $y_0 = F_Y^{-1}(\varrho_{\text{th}})$. We then define RV $Z = X + Y$, and its quantile $z_0 = F_Z^{-1}(\varrho_{\text{th}})$. In order for the bound to hold, we need to prove the following inequality:

$$z_0 \geq \max(x_0, y_0). \tag{45}$$

The inequality can be proven by contradiction. Let, without loss of generality, $y_0 \geq x_0$, such that $\max(x_0, y_0) = y_0$. If we assume $z_0 < y_0$, we have:

$$\begin{aligned} P[z_0 < Z \leq y_0] =& F_Z(y_0) - F_Z(z_0) \\ =& P[Z \leq y_0 | Y \leq y_0] P[Y \leq y_0] \\ &+ P[Z \leq y_0 | Y > y_0] P[Y > y_0] - F_Z(z_0). \end{aligned} \tag{46}$$

By definition, we know that $F_Z(z_0) = F_Y(y_0) = \varrho_{\text{th}}$. Due to the non-negativity of X, we also know that $P[Z \leq y_0 | Y > y_0] = 0$. We can then solve the expression:

$$P[z_0 < Z \leq y_0] = \varrho_{\text{th}}(P[Z \leq y_0 | Y \leq y_0] - 1). \tag{47}$$

As no probability can be larger than 1, $P[z_0 < Z \leq y_0] \leq 0$. Naturally, a negative probability is a contradiction, while the case with probability 0 contradicts the strict monotonicity of the CDFs. As Gamma and left-truncated Gaussian RVs have strictly monotonic CDFs (i.e., $f_X(x) > 0 \, \forall x > 0$), we can extend the result to a summation of N elements and prove that $\tau_{1,L}$ is indeed a lower bound.

We can now prove that $\tau_{1,U}$ is an upper bound to τ_1. The higher quantiles of the sum of RVs are well-known in the statistical literature, and in the quantitative finance in the literature, by the name Value at Risk (VaR) [51]. The second term in the upper bound is equivalent to the *subadditivity* property, i.e., the guarantee that the quantile of the sum is lower than or equal to the sum of the individual RVs' quantiles, which holds for two generic RVs X and Y with CDFs $F_X(x)$ and $F_Y(y)$ if the following is true:

$$S_{X+Y}^{-1}(\nu) \leq S_X^{-1}(\nu) + S_Y^{-1}(\nu), \nu \in [0,1]. \tag{48}$$

where $S_X(x) = 1 - F_X(x)$ is the survival function or complementary CDF. Ibragimov [52] proved that subadditivity holds for the class of log-concave distributions, which includes both the Gamma distribution (see Remark 6) and the Gaussian distribution (see Remark 5). As the RVs in our sum are all log-concave (see Table I), subadditivity holds, and we have:

$$S^{-1}_{\sum_{i=1}^{N} X_i}(1-\varrho_{\text{th}}) \leq \sum_{i=1}^{N} S^{-1}_{X_i}(1-\varrho_{\text{th}}). \tag{49}$$

As $S^{-1}_X(1-\varrho_{\text{th}}) = F^{-1}_X(\varrho_{\text{th}})$ by definition, this proves that the second term inside the minimum is an upper bound.

In order to prove the theorem, the other term inside the minimum should also be an upper bound:

$$\tau_1 \leq \frac{\mu_{T_1}}{1-\varrho_{\text{th}}}. \tag{50}$$

As $\tau_1 > 0$ and $(1-\varrho_{\text{th}}) > 0$ are strictly positive, this is equivalent to the following:

$$\varrho_{\text{th}} \geq 1 - \frac{\mu_{T_1}}{\tau_1}. \tag{51}$$

We can then prove that (50) is true by contradiction. First, let us assume that the given term is not an upper bound, directly negating (51):

$$\varrho_{\text{th}} < 1 - \frac{\mu_{T_1}}{\tau_1}. \tag{52}$$

As the theorem hypothesis states that $F_{T_1}(\tau_1) = \varrho_{\text{th}}$, we can substitute it in the inequality:

$$F_{T_1}(\tau_1) < 1 - \frac{\mu_{T_1}}{\tau_1}. \tag{53}$$

We can express this using the survival function $S_{T_1}(\tau_1) = 1 - F_{T_1}(\tau_1) = P(T_1 > \tau_1)$:

$$S_{T_1}(\tau_1) > \frac{\mu_{T_1}}{\tau_1}. \tag{54}$$

We now state Markov's inequality, knowing that T_1 is non-negative, as it represents a latency, and that $\mathbb{E}[T_1] = \mu_{T_1}$:

$$P(T_1 > \tau_1) = S_{T_1}(\tau_1) \leq \frac{\mathbb{E}[T_1]}{\tau_1} = \frac{\mu_{T_1}}{\tau_1}. \tag{55}$$

The derived expression in (54) is then in direct contradiction with Markov's inequality, proving that (50) is true and the term is indeed an upper bound. As both conditions are always true, we can apply the one that gives the tightest upper bound on the value of τ_1, which is the minimum in Theorem 3.

E. Proof of Theorem 4

This can also be proven in the same way as Theorem 3, following the steps in Appendix D.

References

[1] G. P. Fettweis, "The tactile internet: Applications and challenges," *IEEE Veh. Technol. Mag.*, vol. 9, no. 1, pp. 64–70, 2014.

[2] J. Luo *et al.*, "A teleoperation framework for mobile robots based on shared control," *IEEE Robot. Autom. Lett.*, vol. 5, no. 2, pp. 377–384, 2020.

[3] K. Antonakoglou *et al.*, "Toward haptic communications over the 5G Tactile Internet," *IEEE Comm. Surv. Tut.*, vol. 20, no. 4, pp. 3034–3059, 2018.

[4] J. Zhao *et al.*, "Estimating the motion-to-photon latency in head mounted displays," in *IEEE Virtual Reality (VR)*. IEEE, 2017, pp. 313–314.

[5] S. Hirche and M. Buss, "Human-oriented control for haptic teleoperation," *Proc. IEEE*, vol. 100, no. 3, pp. 623–647, 2012.

[6] F. Müller *et al.*, "Stability of nonlinear time-delay systems describing human–robot interaction," *IEEE/ASME Trans. Mechatronics*, vol. 24, no. 6, pp. 2696–2705, 2019.

[7] P. Barba *et al.*, "Remote telesurgery in humans: a systematic review," *Surgical Endoscopy*, pp. 1–7, 2022.

[8] D. Valenzuela-Urrutia *et al.*, "Virtual reality-based time-delayed haptic teleoperation using point cloud data," *J. Intell. Robot. Syst.*, vol. 96, no. 3, pp. 387–400, 2019.

[9] Z. Shi *et al.*, "Effects of packet loss and latency on the temporal discrimination of visual-haptic events," *IEEE Trans. Haptics*, vol. 3, no. 1, pp. 28–36, 2010.

[10] L. Ruan, M. P. I. Dias, and E. Wong, "Achieving low-latency human-to-machine (h2m) applications: An understanding of h2m traffic for ai-facilitated bandwidth allocation," *IEEE Internet Things J.*, vol. 8, no. 1, pp. 626–635, 2021.

[11] R. Balachandran *et al.*, "Closing the force loop to enhance transparency in time-delayed teleoperation," in *2020 IEEE Int. Conf. Robot. Autom. (ICRA)*, 2020, pp. 10 198–10 204.

[12] X. Ge, R. Zhou, and Q. Li, "5G NFV-based Tactile internet for mission-critical IoT services," *IEEE Internet Things J.*, vol. 7, no. 7, pp. 6150–6163, 2020.

[13] N. Gholipoor *et al.*, "E2E QoS guarantee for the Tactile internet via joint NFV and radio resource allocation," *IEEE Trans. Netw. Service Manag.*, vol. 17, no. 3, pp. 1788–1804, 2020.

[14] M. Maier and A. Ebrahimzadeh, "Towards immersive tactile internet experiences: Low-latency FiWi enhanced mobile networks with edge intelligence [invited]," *IEEE J. Opt. Commun. Netw.*, vol. 11, no. 4, pp. B10–B25, 2019.

[15] Y. Xiao and M. Krunz, "Distributed optimization for energy-efficient fog computing in the tactile internet," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 11, pp. 2390–2400, 2018.

[16] M. Aazam, K. A. Harras, and S. Zeadally, "Fog computing for 5G tactile industrial Internet of Things: QoE-aware resource allocation model," *IEEE Trans. Ind. Informat.*, vol. 15, no. 5, pp. 3085–3092, 2019.

[17] J. Xu, K. Ota, and M. Dong, "Energy efficient hybrid edge caching scheme for tactile internet in 5G," *IEEE Trans. Green Commun. Netw.*, vol. 3, no. 2, pp. 483–493, 2019.

[18] S. Sedaghat and A. H. Jahangir, "RT-TelSurg: Real time telesurgery using SDN, fog, and cloud as infrastructures," *IEEE Access*, vol. 9, pp. 52 238–52 251, 2021.

[19] Y. Mao *et al.*, "A survey on mobile edge computing: The communication perspective," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 4, pp. 2322–2358, 2017.

[20] N. Abbas *et al.*, "Mobile edge computing: A survey," *IEEE Internet Things J.*, vol. 5, no. 1, pp. 450–465, 2018.

[21] S. Suman *et al.*, "Analysis and optimization of the latency budget in wireless systems with mobile edge computing," in *IEEE Int. Conf. Commun.*, 2022, pp. 1–6 (accepted).

[22] "O-RAN near-real-time RAN intelligent controller architecture & E2 general aspects and principles – v1.01," Tech. Spec., 2020.

[23] Cyberglove Systems Inc. (2017). [Online]. Available: http://www.cyberglovesystems.com/cybergrasp/

[24] J. L. Dideriksen, I. U. Mercader, and S. Dosen, "Closed-loop control using electrotactile feedback encoded in frequency and pulse width," *IEEE Trans. Haptics*, vol. 13, no. 4, pp. 818–824, 2020.

[25] M. Chowdhury and M. Maier, "Local and nonlocal human-to-robot task allocation in fiber-wireless multi-robot networks," *IEEE Syst. J.*, vol. 12, no. 3, pp. 2250–2260, 2018.

[26] A. Ebrahimzadeh and M. Maier, "Delay-constrained teleoperation task scheduling and assignment for human+machine hybrid activities over FiWi enhanced networks," *IEEE Trans. Netw. Service Manag.*, vol. 16, no. 4, pp. 1840–1854, 2019.

[27] M. Chowdhury and M. Maier, "Collaborative computing for advanced tactile internet human-to-robot (H2R) communications in integrated FiWi multirobot infrastructures," *IEEE Internet Things J.*, vol. 4, no. 6, pp. 2142–2158, 2017.

[28] C. She, C. Yang, and T. Q. S. Quek, "Cross-layer transmission design for tactile internet," in *IEEE GLOBECOM*, 2016, pp. 1–6.

[29] Y. Feng *et al.*, "Hybrid coordination function controlled channel access for latency-sensitive tactile applications," in *IEEE GLOBECOM*, 2017, pp. 1–6.

[30] N. Gholipoor, H. Saeedi, and N. Mokari, "Cross-layer resource allocation for mixed tactile internet and traditional data in scma based wireless networks," in *IEEE Wireless Commun. Netw. Conf. Wksp. (WCNCW)*, 2018, pp. 356–361.

[31] Z. Xiang *et al.*, "Reducing latency in virtual machines: Enabling tactile internet for human-machine co-working," *IEEE J. Sel. Areas Commun.*, vol. 37, no. 5, pp. 1098–1116, 2019.

[32] A. Aijaz *et al.*, "Shaping 5G for the Tactile Internet," in *5G Mobile Commun.* Springer, 2017, pp. 677–691.

[33] A. Goldsmith, *Wireless communications*. Cambridge university press, 2005.

[34] G. Hu, W. P. Tay, and Y. Wen, "Cloud robotics: Architecture, challenges and applications," *IEEE Netw.*, vol. 26, no. 3, pp. 21–28, 2012.

[35] W. Yuan and K. Nahrstedt, "Energy-efficient soft real-time CPU scheduling for mobile multimedia systems," *SIGOPS Oper. Syst. Rev.*, vol. 37, no. 5, p. 149–163, Oct. 2003.

[36] D. Han *et al.*, "Offloading optimization and bottleneck analysis for mobile cloud computing," *IEEE Trans. Commun.*, vol. 67, no. 9, pp. 6153–6167, 2019.

[37] S. Jošilo and G. Dán, "Selfish decentralized computation offloading for mobile cloud computing in dense wireless networks," *IEEE Trans. Mobile Comput.*, vol. 18, no. 1, pp. 207–220, 2019.

[38] X. Li *et al.*, "Wirelessly powered crowd sensing: Joint power transfer, sensing, compression, and transmission," *IEEE J. Sel. Areas Commun.*, vol. 37, no. 2, pp. 391–406, 2019.

[39] R. Kothiyal *et al.*, "Energy and performance evaluation of lossless file data compression on server systems," in *Proc. of SYSTOR*, ser. SYSTOR '09. New York, USA: Association for Computing Machinery, 2009.

[40] J. Ren, Y. Ruan, and G. Yu, "Data transmission in mobile edge networks: Whether and where to compress?" *IEEE Commun. Lett.*, vol. 23, no. 3, pp. 490–493, 2019.

[41] J.-B. Wang *et al.*, "Joint optimization of transmission bandwidth allocation and data compression for mobile-edge computing systems," *IEEE Commun. Lett.*, vol. 24, no. 10, pp. 2245–2249, 2020.

[42] M. Burrows *et al.*, "On-line data compression in a log-structured file system," in *Proc. Int. Conf. Architectural Support Programming Languages Operating Syst.*, ser. ASPLOS V. New York, NY, USA: Association for Computing Machinery, 1992, p. 2–9.

[43] Z. Zhang *et al.*, "Efficient I/O for neural network training with compressed data," in *IEEE Int. Parallel Distrib. Process. Symp. (IPDPS)*, 2020, pp. 409–418.

[44] H. C. Thom, "A note on the gamma distribution," *Monthly Weather Review*, vol. 86, no. 4, pp. 117–122, 1958.

[45] P. G. Moschopoulos, "The distribution of the sum of independent gamma random variables," *Annals of the Institute of Statistical Mathematics*, vol. 37, no. 3, pp. 541–544, 1985.

[46] H. Murakami, "Approximations to the distribution of sum of independent non-identically gamma random variables," *Mathematical Sciences*, vol. 9, no. 4, pp. 205–213, 2015.

[47] L. Tlebaldiyeva, B. Maham, and T. A. Tsiftsis, "Capacity analysis of device-to-device mmwave networks under transceiver distortion noise and imperfect csi," *IEEE Trans. Veh. Technol.*, vol. 69, no. 5, pp. 5707–5712, 2020.

[48] H. Halawa *et al.*, "NVIDIA Jetson platform characterization," in *Proc. European Conf. Parallel Process.* Springer, 2017, pp. 92–105.

[49] M. Bagnoli and T. Bergstrom, "Log-concave probability and its applications," *Economic Theory*, vol. 26, no. 2, pp. 445–469, 2005.

[50] S. Dharmadhikari and K. Joag-Dev, *Unimodality, Convexity, and Applications*. Elsevier, 1988.

[51] D. C. Joaquin, "Value at risk: Is a theoretically consistent axiomatic formulation possible?" *The Quarterly Review of Economics and Finance*, vol. 49, no. 2, pp. 725–729, 2009.

[52] R. Ibragimov, "Portfolio diversification and value at risk under thick-tailedness," *Quantitative Finance*, vol. 9, no. 5, pp. 565–580, 2009.

Suraj Suman (Member, IEEE) received the B.Tech. degree from IIITDM Jabalpur, India in 2013, the M.Tech. degree from IIT Patna, India in 2015, and the Ph.D. degree from IIT Delhi, India in 2020. He was a postdoctoral researcher with the Department of Electronics System, Aalborg University, Aalborg, Denmark from January 2021 to June 2022. He is currently an Assistant Professor with the Department of Electrical Engineering, IIT Patna, India. His research interests include tactile internet, green communications, and aerial communication systems.

Federico Chiariotti (Member, 2019) received his Ph.D. in information engineering from the University of Padova, Italy in 2019, where he is currently an assistant professor. From 2020 to 2022, he worked as a postdoc and assistant professor at the Department of Electronic Systems, Aalborg University, Denmark. He has authored over 70 published papers on wireless networks and the use of artificial intelligence techniques to improve their performance. His current research interests include semantic communications, transport layer protocols, Age of Information, bike sharing system optimization, and adaptive video streaming. He was a recipient of the Best Paper Award at several conferences, including the IEEE INFOCOM 2020 WCNEE Workshop.

Čedomir Stefanović (Senior Member, IEEE) received the Diploma Ing., Mr.-Ing., and Ph.D. degrees from the University of Novi Sad, Serbia. He is currently a Professor with the Department of Electronic Systems, Aalborg University, where he leads Edge Computing and Networking Group. He is a Principal Researcher on a number of European projects related to IoT, 5G, and mission critical communications. He has coauthored more than 100 peer-reviewed publications. His research interests include communication theory and wireless communications. He serves as an Editor for the IEEE INTERNET OF THINGS JOURNAL.

Strahinja Došen (Member, IEEE, EMBS) received the Diploma of Engineering in electrical engineering and the M.Sc. degree in biomedical engineering in 2000 and 2004, respectively, from the Faculty of Technical Sciences, University of Novi Sad, Serbia, and the Ph.D. degree in biomedical engineering from the Center for Sensory-Motor Interaction, Aalborg University, Aalborg, Denmark, in 2008. From 2011 to 2017, he was working as a Research Scientist at the Institute for Neurorehabilitation Systems, University Medical Center Gottingen, Germany, and then as an Associate Professor at the Department of Health Science and Technology, Aalborg University, Denmark. Currently, he is a Full Professor in Rehab Robotics at the same Department where he leads a research group on Neurorehabilitation Systems. Prof. Dosen is a principal investigator for AAU and HST in several EU (Tactility, Wearplex, Sixthsense, and SimBionics) and nationally (Robin, Remap, and Climb) funded projects. He has published more than 90 manuscripts in peer-reviewed journals. His main research interest is in the closed-loop control of movements and assistive systems, including human-machine interfacing, control of bionic limbs and rehabilitation robotics, artificial sensory feedback, and functional electrical stimulation.

Petar Popovski (Fellow, 2016) is a Professor at Aalborg University, where he heads the section on Connectivity and a Visiting Excellence Chair at the University of Bremen. He received his Dipl.-Ing and M. Sc. degrees in communication engineering from the University of Sts. Cyril and Methodius in Skopje and the Ph.D. degree from Aalborg University in 2005. He received an ERC Consolidator Grant (2015), the Danish Elite Researcher award (2016), IEEE Fred W. Ellersick prize (2016), IEEE Stephen O. Rice prize (2018), Technical Achievement Award from the IEEE Technical Committee on Smart Grid Communications (2019), the Danish Telecommunication Prize (2020) and Villum Investigator Grant (2021). He was a Member at Large at the Board of Governors in IEEE Communication Society 2019-2021. He is currently an Editor-in-Chief of IEEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS and a Chair of the IEEE Communication Theory Technical Committee. His research interests are in the area of wireless communication and communication theory. He authored the book “Wireless Connectivity: An Intuitive and Fundamental Guide.”