ViTaLS – A Novel Link-Layer Scheduling Framework for Tactile Internet over Wi-Fi

Vineet Gokhale[1], *IEEE Member* , Kees Kroep[1], *IEEE Member* , R. Venkatesha Prasad[1], *IEEE Senior Member* , Boris Bellalta[2], *IEEE Senior Member* , Falko Dressler[3], *IEEE Fellow*

[1]Delft University of Technology, The Netherlands
[2]Pompeu Fabra University, Spain
[3]TU Berlin, Germany

***Abstract*—The pioneering field of *Tactile Internet (TI)* will enable the transfer of human skills over long distances through haptic feedback. Realizing this demands a roundtrip latency of sub-5 ms. In this work, we investigate the capability of Wi-Fi 6 and existing TI scheduling/multiplexing schemes in meeting this stringent latency constraint. Taking the concrete example of the state-of-the-art Video-Haptic multiplexer (VH-multiplexer), we highlight the pitfalls of relying on the existing Wi-Fi 6 systems for TI communication. To circumvent this, we propose *Video-Tactile Latency Scheduler (ViTaLS)* – a novel link layer framework for tuning the video-tactile frame transmissions to suit their heterogeneous QoS requirements. We present a mathematical model to characterize the packet transmission duration of ViTaLS. Using a custom simulator, we validate our model and measure the objective performance improvement of ViTaLS over VH-multiplexer. We also present *ViTaLS-optimal* – a variant of ViTaLS, for further reducing the tactile latency. Objectively, we show that ViTaLS-optimal yields a latency improvement of up to $82\,\%$. Based on experiments conducted on a real TI testbed, we subjectively demonstrate that ViTaLS-optimal outperforms the VH-multiplexer.**

***Index Terms*—Tactile Internet, Wi-Fi 6, 802.11ax, ViTaLS, ultra-low latency.**

I. Introduction

The field of real-time applications is poised to achieve a whole new level of immersion due to the upcoming field of *Tactile Internet (TI)* [1]. TI's ability to transport haptic (touch) feedback will enable high-precision interaction with remote environments in an unprecedented manner. The most celebrated applications of TI are telesurgery, telemaintenance, and remote repairs in Industry 4.0.

A. Background

Taking the Industry 4.0 use case, a human operator equipped with a tactile wearable (glove, exoskeleton, or a full-body suit) in the *master domain* controls a robot arm inside a manufacturing plant (*controlled domain*) for performing a physical task. This setup is depicted in Fig. 1.

Sense-communicate-actuate. A typical TI application encompasses ① sensors for kinematic (position, velocity, orientation), haptic, and visual signals, ② communication modules for transporting the sensed signals, and ③ actuators on the robot arm and tactile wearable. For sensing the kinematic signal, the tactile wearable consists of sensors for capturing the operator's actions. The robot arm is equipped with actuators for reproducing the operators' actions as well as haptic and vision sensors for actuation in the master domain.

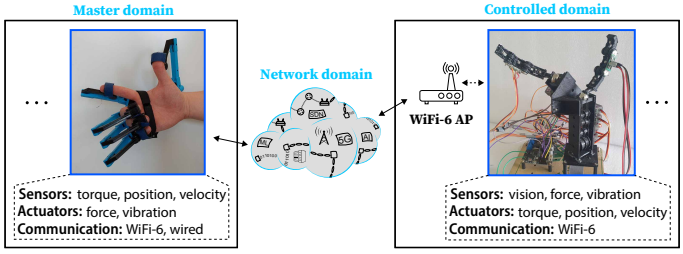


Fig. 1: Our in-house TI testbed showing the human operator with the tactile glove and the custom robotic arm with tactile sensors connected through a Wi-Fi 6 first/last mile communication.

Unprecedented network challenges. To enable effective teleoperation, worst-case roundtrip latency of sub-$5\,\mathrm{ms}$ is crucial for tactile traffic [2].[1] Not satisfying such ultra-low latency (ULL) guarantees may lead to catastrophic consequences, especially in mission-critical applications. The TI standards [2] also recommend packet-level reliability of 99.9999% as a key requirement for seamless user interaction. However, this is only a speculation, and there is no evidence-based substantiation of this reliability requirement. On the contrary, several independent studies reveal that the user experience decreases only marginally even up to 30% packet loss for many applications [3], [4], [5], [6]. Leveraging these insights is key to delivering high-quality performance for TI applications.

Under a tight ULL budget, it is crucial to perform optimizations at every segment of the network. Tremendous advancements, such as Time Sensitive Networking (TSN) and Deterministic Networking (DetNet), are underway to achieve ULL guarantees in the core network (*network domain*) [7]. However, the core network is operated by ISPs and telecom providers. This limits the ability to deploy and test new algorithms in the core network. On the contrary, much of the research in the first/last mile link has been on improving the network throughput. While this is important, it does not

[1]As both kinematic and haptic traffic are key for generating tactile feedback, we refer to them jointly as *tactile sensor traffic* or simply *tactile traffic*.

necessarily yield ULL performance. In this work, we focus on the first/last mile communications where Wi-Fi emerges as the forefront runner due to its large-scale deployments. IEEE 802.11be working group is already aiming to provide specific solutions to offer ULL guarantees [8], [9], although they are not tailored for TI applications. Wi-Fi operates in ISM bands and there exist open-source Wi-Fi stacks. This provides us with a great platform for the deployment and evaluation of custom link-layer designs for TI applications.

Heterogeneous QoS needs specialized scheduling. It is important to understand the Quality of Service (QoS) requirements of tactile and video modalities in TI. While the tactile modality requires ULL, a loss of even up to 30% is manageable, as stated earlier. On the other hand, the video feedback has a much higher latency budget ($\sim$30 ms) but an extremely low loss-tolerance of 2% [10], [11]. Satisfying these heterogeneous requirements demands efficient transmission scheduling policies. This is even more crucial in Wi-Fi networks where channel access uncertainty and collisions are significant. This results in large and unpredictable delays that can hamper the QoS performance of TI.

B. Relevant Literature

We now briefly review the related literature in the areas of Wi-Fi and TI to position this work properly.

The classical 802.11e amendment, a.k.a Enhanced Distributed Channel Access (EDCA) [12], provides different levels of QoS support by defining Access Categories (ACs) for prioritizing realtime traffic. EDCA is based on channel contention using a random backoff mechanism. Several enhancements have been proposed to improve the efficiency of EDCA further [13], [14], [15]. Additionally, to enhance Wi-Fi's overall performance, the state-of-the-art Wi-Fi 6 provisions Orthogonal Frequency-Division Multiple Access (OFDMA) which can significantly improve the latency performance [16]. However, the question of "*how to leverage the ACs and OFDMA optimally for TI communication?*" has not been addressed to date.

Independent of Wi-Fi, a few application layer strategies exist for multiplexing video and tactile traffic. The proposals Video-Haptic (VH) multiplexer [17] and Dynamic Packetization Module [18] augment the two traffic types to generate a single data stream, thereby inevitably requiring the use of a single AC. Such strategies fail to leverage Wi-Fi's ACs for differentiated service between video and tactile traffic. On the other hand, the works in [19], [20] propose to separate tactile and video frames for leveraging different ACs. This is a significant development. However, relying on standard Wi-Fi scheduling for multiple streams can be worse than single-stream solutions due to the increase in the amount of contending ACs. Without designing specialized scheduling policies for TI traffic, this is counterproductive.

Only a handful of works have studied TI communication over Wi-Fi. The work in [21] studies tactile latency using Hybrid Coordination Function Channel Access (HCCA), where the AP orchestrates the channel access. However, HCCA is not embraced by Wi-Fi vendors due to its protocol complexity, and hence is not a realistic Wi-Fi solution [22]. The work in [23] studies the latency-loss tradeoff in tactile traffic by sending only the latest buffered tactile frame at the time of channel access and dropping all previous ones. The work in [24] proposes a scheduling algorithm for TI over the FiWi (Fiber Wireless) network. None of the above works consider the video feedback, which forms the bulk of TI traffic. Hence, the above works are not comprehensive and do not consider realistic TI application scenarios.

Using standard Wi-Fi design and even state-of-the-art proposals leads to a severely under-optimized system for supporting TI communication. Contrary to this, one needs to jointly consider both TI application requirements as well as network characteristics for designing an effective TI framework. Motivated by this, we propose our solution in this work.

C. Our contributions

In this work, we demonstrate the major issues related to TI communication over Wi-Fi by taking state-of-the-art 802.11ax as well as TI scheduling/multiplexing protocols. Identifying link layer as the bottleneck, we propose a novel link layer framework, called *Video-Tactile Latency Scheduler (ViTaLS)*. The novelty of our work is the media-aware scheduling tuned for heterogeneous QoS requirements of video and tactile modalities, while fully conforming to the existing Wi-Fi standards. Our specific contributions in this work are the following: ❶ By taking as the reference VH-multiplexer, we provide a detailed overview of TI communication over Wi-Fi 6 and highlight their pitfalls (Sec. II). ❷ We propose ViTaLS as a way toward TI communication over Wi-Fi 6/7 networks. We describe the various ingredients of ViTaLS and present its design (Sec. III-A). ❸ We develop a mathematical model for characterizing packet duration of ViTaLS (Sec. III-C). Apart from providing a formal description of ViTaLS, our model validates the custom simulator used. ❹ We also present *ViTaLS-optimal* – an efficient variant of ViTaLS for optimizing the MAC queue sizes. Through extensive objective and subjective evaluations, we demonstrate that ViTaLS-optimal outperforms the VH-multiplexer (Sec. IV). ❺ We provide implementation notes to serve as guidelines for vendors/implementers to deploy ViTaLS-optimal on Wi-Fi 6/7 devices (Sec. III-E).

II. TI over Wi-Fi 6: An overview

Consider the Industry 4.0 use case of a connected factory (depicted in Fig. 1) with the communication inside the plant enabled by Wi-Fi 6.[2] Human operators control wireless, mobile robot arms inside the factory. Given the mission-critical nature of TI applications, it is reasonable to assume a tightly controlled Wi-Fi 6 network serving only TI traffic. We focus on a single Basic Service Set (BSS) where a single Wi-Fi 6 AP serves a set of Wi-Fi 6 STAs. This helps to completely remove the effects of overlapping BSS, thereby enabling us to isolate the performance of ViTaLS. We take the VH-multiplexer [17] as the reference TI multiplexing scheme.

[2]Although this work is built on top of Wi-Fi 6 specifications, we expect that it can also contribute to building a TI profile for upcoming Wi-Fi 7.

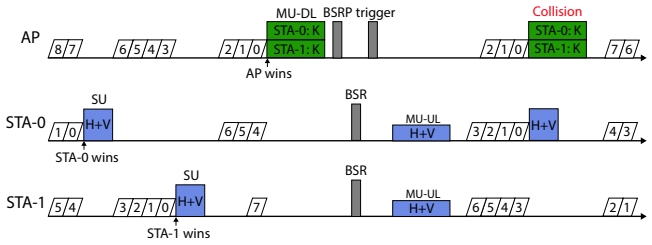


Fig. 2: Timing diagram showing DL and UL transmission within the Wi-Fi 6 framework when using VH-multiplexer.

A. Video-Haptic (VH) multiplexer

The VH-multiplexer is designed to operate in the controlled domain where video and haptic traffic are generated. Let us assume standard frame rates for video and haptic streams of 60 Hz and 1 kHz, respectively. The VH-multiplexer splits each video frame into multiple fragments at the application layer. An application layer message consists of an augmented haptic frame and a video fragment (*H+V* in Fig. 2) forming a MAC Protocol Data Unit (MPDU). This prevents the transmission of large video frames from holding up haptic frames while also meeting the video latency budget. The link-layer scheduling is managed by Wi-Fi 6.

B. Wi-Fi 6 communication

Channel access: We show EDCA of Wi-Fi 6 in Fig. 2. Compared to a slower AC, a faster AC has a smaller (i) contention window (CW) range and (ii) a smaller pre-defined interval known as arbitration interframe spacing (AIFS). When the channel is busy, a device backs off by selecting a random backoff counter (BO) uniformly from [0, CW-1], where $\text{CW} \in [\text{CW}_{\text{min}}, \text{CW}_{\text{max}}]$. When the channel becomes idle for AIFS, the BO countdown starts. BO is counted down every time the channel is idle for a pre-defined interval of *slot size* denoted as T_s. When BO reaches 0, the device transmits a packet. Upon collision, the CW follows a binary exponential increase until CW_{max} is reached.

Video-tactile transmission: When a STA wins contention, it transmits *H+V* frames in single-user (SU) mode on uplink (UL) occupying the entire bandwidth. On the other hand, when the AP wins contention, it could employ OFDMA if there are kinematic (*K*) frames for multiple STAs in its queue. This is multiuser downlink (MU-DL) transmission. Further, the AP can also provision MU-UL transmissions.[3] In MU-UL, the scheduled STAs transmit *H+V* frames in allocated portions of the channel.

To summarize, the UL transmissions happen when either a STA or the AP wins the contention, whereas DL transmissions only occur when the AP wins. Therefore, when the AP wins, it is important to first transmit the *K* frames as they would be queued up since the previous AP channel access. For TI applications, the AP must perform a MU-DL first and then provision a MU-UL. We adopt this strategy throughout this paper.

[3]Note that we consider only OFDMA-based multiuser transmission in this work and not MIMO-based multiuser transmission.

For MU transmissions, the channel is divided into blocks of subcarriers (tones), known as Resource Units (RUs). For example, an 80 MHz channel is made up of 996 tones and can be split into two 484-tone RUs, four 242-tone RUs, eight 106-tone RUs, and so on. We adopt a simple RU-allocation scheme proposed in [25] for maximizing the number of STAs scheduled during a MU-UL access. Table I summarizes the RU allocation and number of scheduled STAs. For the MU-DL, the AP looks up its queue to schedule DL transmissions. In the case of MU-UL, the information regarding the number of STAs with UL data and the scheduled ones are exchanged using Buffer Status Report Poll (BSRP), Buffer Status Report (BSR), and trigger frames [26].

# STAs with data	1	2	3	4	5	6	7	≥8
# scheduled STAs	1	2	3	4	5	6	7	8
RU size (tones)	996	484	242	242	106	106	106	106

TABLE I: Number of available and scheduled STAs in MU-UL for 80 MHz channel for the RU-allocation scheme in [25].

C. Shortcomings of VH-multiplexer with Wi-Fi 6

- Video-haptic augmentation leaves no scope to selectively transmit or drop (during congestion) frames belonging to a particular modality (haptic or video). Hence, VH-multiplexer fails to enable prioritized frame transmissions, which severely hampers QoS performance.
- Collision between augmented video-haptic frames results in a larger collision duration than when only haptic frames collide. In a collision-prone Wi-Fi network, this results in a considerable amount of wasted bandwidth.

We quantify the above claims objectively in Sec. IV-B1. In order to overcome the above limitations, we propose the *Visual-Tactile Latency Scheduler* (ViTaLS) framework.

III. The Proposed ViTaLS Framework

We present the architecture and design of the proposed ViTaLS framework depicted in Fig. 3. For concreteness in exposition, we have also depicted its workflow in Fig. 4. Note that we have not explicitly shown the first/last mile communication in the master domain for brevity and also to map the workflow to the considered setup of Wi-Fi in the controlled domain.

A. Design

1) Leveraging Wi-Fi ACs: To enable transmission prioritization between tactile and video frames, we propose to leverage the different ACs of Wi-Fi. At the STAs, the haptic and video frames are assigned to AC_VO (fastest AC) and AC_VI (slower AC), respectively. This allows us to tune the scheduling mechanism and other transmission parameters, such as retry limit and CW range, to suit the heterogeneous QoS requirements. However, this also poses a challenge. Each STA now has two independently contending ACs, potentially leading to higher collisions than single AC solutions [17], [18]. Although Wi-Fi offers virtual collision management between ACs within a device, AC_VI and AC_VO packets belonging

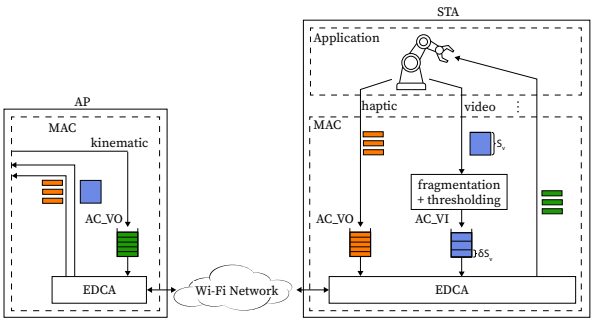


Fig. 3: Schematic representation of ViTaLS framework depicting the different steps involved at MAC layer.

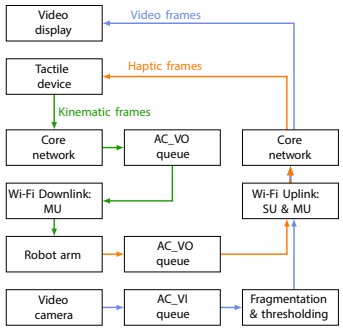


Fig. 4: Detailed workflow showing the data flow between the master and controlled domains encompassing the ViTaLS functionalities for Wi-Fi network in the controlled domain.

to different devices can still collide. To mitigate this issue, we propose to increase the CW range of AC_VI significantly compared to that of AC_VO so that the video frames reduce their SU transmissions. Essentially, the idea is to reduce tactile-video frame collisions in a Wi-Fi 6 standards-compliant manner. At the AP, kinematic frames are enqueued in AC_VO as they also require ULL guarantees. Haptic transmissions in SU mode are marked at $t_2, t_3,$ and t_4 in Fig. 5). We use the notation t_x to refer to different points in time where events of interest for us take place.

2) Scheduling video frames in MU-UL: While increasing the CW range of AC_VI favors tactile frames, it can potentially starve the video frames of channel resources, leading to video QoS violations. To address this, we leverage AP-initiated MU-UL transmissions for scheduling the video frames. The idea is to exploit this contention-free UL transmissions for transmitting high-reliability video frames. This implies that the video latency is predominantly dependent on AP channel accesses. Since the kinematic frames use AC_VO, one can expect the AP to get channel access quite often, thereby benefiting the video traffic. Further, collision-free video transmission also meets the high reliability requirement of the video stream. Moving video transmissions predominantly to MU-UL is an important feature of ViTaLS since the collisions occur only between the tactile frames, which are typically small.

3) Video fragmentation and thresholding: Transmitting a video frame as a whole results in a large MU-UL duration. As an example, consider 8 STAs, each employing modulation

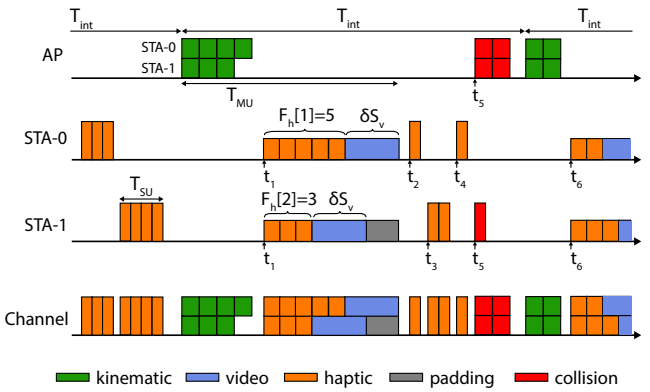


Fig. 5: Timing diagram showing UL and DL transmissions as defined in ViTaLS framework from the standpoint of Wi-Fi devices and the channel.

and coding scheme MCS-9 and generating video traffic at 15 Mbps. A channel data rate of 400 Mbps results in a MU-UL duration of 6 ms. This causes significant hold-up of tactile frames, increasing their worst-case latency. To prevent this, we adopt the idea of video fragmentation from the VH-multiplexer and optimize it further. Each video frame (of size S_v) is split into multiple fragments of size δS_v. Here, $\delta \leq 1$ denotes a parameter called *fragment threshold*. If video frames are available at the time of the MU-UL, the STA transmits a maximum of one fragment. Going back to our numerical example, $\delta = 0.33$ implies three fragments per frame, lowering the MU-UL duration to 2 ms.

A small δ is favorable for containing the tactile latency, but it requires more MU-UL accesses per video frame. In the case of a small number of STAs (denoted by N), a small δ suffices for meeting the video QoS requirements. However, higher N results in significant video latency. On the other hand, a large δ is favorable for video streams but is problematic for tactile streams. Therefore, an optimal choice of δ is important for seamless TI interaction. We elaborate on the impact of δ on TI latency in Sec. IV-B.

4) Tactile queue sizing: As explained in Sec. I, the perceptual experience degrades marginally even up to 30% tactile losses [3], [4], [5], [6]. This insight provides us with an opportunity to maintain a good user experience even during high load conditions. An efficient way to achieve this is by limiting the tactile queue size (denoted by Q) at the MAC layer. When the queue is full, the older tactile frames are considered outdated and dropped to make room for the newer ones. This induces an upper bound on the tactile queuing latency, although at the expense of loss. It is important to state the difference in Q at a STA and the AP, denoted by Q_{sta} and Q_{ap}, respectively. Q_{sta} is the maximum permissible haptic frames in the queue. Q_{ap} is the maximum permissible kinematic frames for each STA. Using Q_{sta} and Q_{ap} as design parameters, we demonstrate their impact on the overall performance in Sec. IV-B.

5) Heterogeneous payload: Since MU-UL provides collision-free channel access to the STAs, it is beneficial to leverage MU-UL access for tactile frame transmissions,

Algorithm 1 ViTaLS algorithm at STA

```
if haptic queue is full then
    Drop oldest frame upon new frame arrival
end if
if STA-l wins contention then
    Transmit F_h[l] haptic data or F_v[l] video data
end if
if STA-l is scheduled in MU-UL then
    if both queues are non-empty then
        Transmit multi-TID AMPDU with F_h[l] haptic
        data and F_v[l] video data
    else
        Transmit F_h[l] haptic data or F_v[l] video data
    end if
end if
```

Algorithm 2 ViTaLS algorithm at AP

```
if queue has Q_ap kinematic frames for STA-l then
    Drop oldest frame for STA-l upon new frame arrival
end if
if AP wins contention then
    Schedule STAs with most DL data
    Send kinematic AMPDUs on allocated RUs
    if STAs have UL data then
        Compute MU-UL duration using UL data and RUs
        Schedule STAs with highest UL data for MU-UL
    end if
end if
```

when possible, without necessitating any control overhead. Wi-Fi 6 introduces *Multi-Traffic Identifier Aggregated MPDU (multi-TID AMPDU)* where heterogeneous MPDUs can be included in the same packet. According to multi-TID AMPDU, when a particular AC is scheduled, even MPDUs belonging to higher priority ACs can be transmitted along with the scheduled AC MPDUs. We leverage this feature in ViTaLS to piggyback haptic frames when video frames are scheduled in MU-UL (t_1 and t_6 in Fig. 5). In MU-UL transmissions, the haptic frames are packetized prior to video frames as shown in Fig. 5. Note that this does not affect the video latency given that it has a less stringent QoS requirement ($30\,\mathrm{ms}$) as well as the transmission time of a haptic frame is small (around $40\,\mu\mathrm{s}$ for our setup explained later in Sec. IV-A1). This greatly benefits the tactile latency. Further, as per Wi-Fi 6 standards, padding bits are added to synchronize MU-UL transmission across the STAs. When the video buffer is empty, only haptic frames are transmitted in MU-UL.

B. ViTaLS algorithm

We will now describe the ViTaLS scheduling algorithm at both STA (Algo. 1) and AP (Algo. 2). Let $F_h[l]$ and $F_k[l]$ denote the amount of queued haptic and kinematic data (in bytes) belonging to STA-l, respectively.

SU transmissions: When STA-l wins the channel contention, it sends a packet comprising of either $F_h[l]$ haptic data or a video fragment depending on the winning AC. We expect negligible video transmission in SU mode due to higher CW for AC_VI (explained in Sec. III-A).

MU transmissions: Since we are employing the RU-allocation proposed in [25], an MU-DL transmission can accommodate up to 8 STAs based on the amount of DL data per STA. After MU-DL transmission, the AP seeks the *maximum permissible UL data* from each STA using BSRP. Let the maximum permissible video data of STA-l be denoted by $F_v[l]$ (in bytes). This is the minimum between δS_v and the video queue occupancy. With this information, the AP schedules up to 8 STAs based on $F_h[l]$+$F_v[l]$. The MU-UL duration is computed as the transmission time for the STA with the highest $F_h[l]$+$F_v[l]$ and is dependent on the RU allocated for that STA and MCS used. MU-UL duration (in the form of PHY layer field *L-SIG length*) along with the RU allocation are then communicated to all STAs using the trigger frames. This is followed by the MU-UL transmission of multi-TID AMPDUs or haptic AMPDUs.

C. Mathematical Model

We present an analytical model for estimating the packet durations of ViTaLS. For the ease of analysis, we make the following reasonable assumptions: ① The probability of AC_VI winning the channel contention is negligible. ② The CW range and the number of backoff stages of AC_VO at AP and STAs are identical. ③ There is no random access during MU-UL transmissions resulting in no collisions during this period as the AP broadcasts the MU-UL schedule to all STAs. ④ All the tactile frames in the queue are transmitted when a device gets a channel access. ⑤ There are no legacy (pre Wi-Fi 6) devices connected to the AP as TI communication necessitates a tightly controlled network.

The seminal work of Bianchi [27] provides an accurate model for the throughput performance of Wi-Fi. Many later works followed up on Bianchi's work to model the latency performance of Wi-Fi [28], [29]. The work in [26] estimates throughput for OFDMA-based Wi-Fi 6 systems. Based on a per-slot analysis, the above works show that the packet transmission probability, denoted by τ, of a device in a slot is a constant that is dependent only on the CW parameters. As per these works, if the CW parameters of the AP and STAs are identical, they have equal τ. It is important to note that this holds good only if any of the following conditions are satisfied. ⓐ Like legacy Wi-Fi systems, there is no AP-initiated MU-UL transmission [27], [28], [29], ⓑ the STAs do not reset their backoff counters after MU-UL transmissions for further channel contention, as is implicitly assumed in [26], [25]. In ViTaLS, although the CW parameters of AP and STA AC_VO are identical (assumption ②), none of the above conditions is satisfied. While condition ⓐ does not hold as ViTaLS relies heavily on MU-UL transmissions, condition ⓑ fails since the haptic queue of a STA is completely emptied during MU-UL transmissions (assumption ④) leading to resetting the backoff counters. Hence, these models are not fully applicable in our case. Therefore, capturing the above intricacies of ViTaLS

requires a departure from existing works. We take up this non-trivial exercise in the following part.

Characterizing transmissions: As explained previously, in ViTaLS the AC_VO backoff counters at the devices are reset every time the AP gains channel access. To make the analysis concrete, we view the temporal axis as a continuous series of time durations between the start of consecutive, successful AP transmissions, which we call "*intervening time*". This is denoted as T_{int} in Fig. 5. Note that within T_{int}, there can be $x \geq 0$ SU transmissions. This includes collided as well as successful ones. For the ease of analysis, we ignore the binary exponential nature of the backoff process. To begin with, let us consider the number of transmissions from a given STA in T_{int}. x can be interpreted as the number of independent backoff choices such that their cumulative sum is smaller than AP's backoff choice. We use the fact that the PDF of the sum of independent random variables is the convolution of their individual PDFs. Ignoring collisions, we can think of the devices as picking a real backoff value uniformly with the distribution $f(x) = 1$, if $x \in [0, 1]$, and 0 otherwise. Note that we handle collisions separately later in our model. This gives us the probability of at least n transmissions by the STA as

$$P(x \geq n) = \int_0^1 \Big(f(x) * \overset{n \text{ convolutions}}{\cdots} * f(x) \Big)\, dx = 1/(n+1)!$$

The limits of the integral denote the range of AP backoff values on the new scale. From first principles, the probability of exactly n transmissions by the STA can then be derived as

$$P(n) = P(x \geq n) - P(x \geq n+1) = (n+1)/(n+2)!$$

We can now calculate the expected number of SU transmissions per STA in T_{int} as

$$E[n] = \sum_{n=0}^{\infty} nP(n) = e - 2 \approx 0.72. \tag{1}$$

This means that for every AP transmission, each STA transmits a mean of approximately 0.72 SU packets. This reveals that τ of the AP and STAs are non-identical in ViTaLS. This important finding is a significant departure from the existing works and forms the basis of our mathematical model.

Collision model: The work in [26] derives τ as

$$\tau = \left[\frac{(1 - P_c - P_c(2P_c)^m)(\text{CW}_{\min} + 1)}{2(1 - 2P_c)} + \frac{1}{2} \right]^{-1} \tag{2}$$

where m is the retry limit, P_c is the collision probability of a transmitted packet. We will append the notations used so far with subscripts 'ap' and 'sta' to denote the specific parameters of AP and STA, respectively. Due to the asymmetric nature of transmissions between AP and STA derived in Eq. (1), we can obtain the respective transmission probabilities as

$$\tau_{\text{ap}} = \tau, \text{ and } \tau_{\text{sta}} = \alpha\tau, \tag{3}$$

where $\alpha = E[n]$. Based on τ_{sta} and τ_{ap}, the collision probabilities of AP and STA can be expressed as

$$P_{\text{c,ap}} = 1 - (1 - \tau_{\text{sta}})^N, \tag{4}$$

$$P_{\text{c,sta}} = 1 - (1 - \tau_{\text{ap}})(1 - \tau_{\text{sta}})^{N-1}. \tag{5}$$

The closed form expressions for $P_{\text{c,ap}}$ and $P_{\text{c,sta}}$ can be obtained by solving Eqs. (2)-(5).

Due to packet retransmissions, one can think of collisions as resulting in additional data to transmit from the standpoint of the network. Further, the collisions result in bigger packets due to MPDU aggregation. Therefore, the overall data rate scales by a factor of $1/(1 - P_c)$. Due to assumption ③, the AP collisions result only in kinematic frame retransmissions. Further, MU-UL transmissions also involve padding due to the unequal haptic frames at the STAs. Essentially, the amount of padding is determined by the STA with the highest haptic queue occupancy at the time of MU-UL transmission. As an upper bound, every successful SU transmission by a STA would create a padding frame equal to the amount of transmitted haptic data. Hence, the data rates of MU and per-STA SU transmissions can be respectively expressed as

$$D_{\text{MU}} = N\Big[\Big(\delta S_v + H\Big) f_v + (S_h + H) f_h + \frac{(S_k + H) f_k}{1 - P_{\text{c,ap}}}\Big]$$

$$D_{\text{SU}} = \Big(\frac{\alpha}{1+\alpha}\Big) \frac{(S_h + H) f_h}{(1 - P_{\text{c,sta}})}$$

where H denotes the header overhead per MPDU, $\alpha/(1+\alpha)$ is the ratio of haptic data transmitted by a STA to that by the AP. $f_h, f_k,$ and f_v denote the frame rates of haptic, kinematic, and video traffic.

Fine-grained durations: Denoting channel bandwidth as B, the mean duration of each MU and SU transmission can be respectively expressed as

$$T_{\text{MU}} = T^e_{\text{MU}} + \frac{D_{\text{MU}} T_{\text{int}}}{B}, \quad T_{\text{SU}} = T^e_{\text{SU}} + \frac{D_{\text{SU}} T_{\text{int}}}{\alpha B}, \tag{6}$$

where T^e_{MU} and T^e_{SU} denote the "*extra time*" per MU and SU transmission, respectively, due to control signals (TF, BSR, BSRP, etc.), PHY layer header, and other overheads (SIFS, AIFS, etc.). The expected backoff duration can be given as

$$T_b = \frac{\text{CW}_{\min} T_s}{2} \left[\frac{1 - (2P_{\text{c,ap}})^m}{1 - 2P_{\text{c,ap}}} \right] \tag{7}$$

We can now express T_{int} as

$$T_{\text{int}} = T_b + \alpha N T_{\text{SU}} + T_{\text{MU}}. \tag{8}$$

Substituting Eqs. (7) and (8) in Eq. (6), we obtain the simultaneous equations

$$-\alpha N T_{\text{SU}} + \Big(\frac{B}{D_{\text{MU}}} - 1\Big) T_{\text{MU}} = \frac{B}{D_{\text{MU}}} T^e_{\text{MU}} + T_b,$$

$$\Big(\frac{B}{D_{\text{SU}}} - N\Big) T_{\text{SU}} - \frac{T_{\text{MU}}}{\alpha} = \frac{B}{D_{\text{SU}}} T^e_{\text{SU}} + \frac{T_b}{\alpha}. \tag{9}$$

The above equations can be solved to obtain closed form expressions for T_{MU} and T_{SU} in terms of parameters of Wi-Fi and video-tactile traffic.

The parameters T_{MU} and T_{SU} significantly affect the latency performance of ViTaLS. Modeling video-tactile as a function of the above parameters is non-trivial and requires further analysis. This forms a part of our future work. Hence, in Sec. IV we validate the estimated T_{MU} and T_{SU} through simulations.

D. Complexity Overhead

In ViTaLS, we primarily leverage the MU-UL and multi-TID AMPDU features of 802.11ax protocol without modifying either the channel contention or the frame exchange mechanisms. For MU-UL transmissions, the scheduling and allocation of RUs are done by the AP as discussed earlier in Sec. III-B. This requires neither sorting of STAs according to their queue sizes nor proportional allocation of RUs. Hence, the MU-UL transmissions of ViTaLS adds no complexity overhead with respect to 802.11ax framework. Further, the STAs use standard 802.11ax headers to indicate the multi-TID AMPDU. Hence, there is no additional header complexity incurred as well. For composing the heterogeneous payload, the STAs packetize the haptic frames prior to the video fragments. Note that a haptic frame is typically much smaller in size compared to a video fragment. Further, a STA packetizes all haptic frames in the queue without requiring to perform any sorting or searching. To summarize, by design, the implementation changes required for ViTaLS are within the scope of 802.11ax framework and do not introduce any additional complexities in terms of communication as well as coding.

E. Implementation notes

Deployment of ViTaLS requires only minor modifications at the link layer. Firstly, the CW range of AC_VI should be configured to a much larger value than that of AC_VO. Under tightly controlled Wi-Fi networks allowing only TI traffic, this will not increase the video latency proportionately as the MU-UL access will satisfy the necessary video QoS requirements. Secondly, video frames at STAs should be fragmented as per the pre-defined δ before forwarding to the physical layer. Within the BSRs, the STAs must communicate to AP the amount of queued haptic frames and the permissible video data (based on δ) instead of the entire video queue occupancy. Lastly, the MAC queue of AC_VO should adopt a *head-drop* scheme for dropping earlier haptic frames when new ones arrive. The proposed updates to link layer can also serve as a basis for developing the TI operation profile for Wi-Fi 7.

IV. Performance Evaluation

In this section, we describe the experimental setup for the performance evaluation of ViTaLS and present our important findings.

A. Experimental Setup

1) Objective evaluation: Since deploying custom MAC algorithms at the kernel level on real Wi-Fi devices is challenging, we developed a custom Wi-Fi MAC simulator written in C++ for objective evaluation of ViTaLS. To facilitate rapid developments in the field of TI over Wi-Fi, we have open-sourced our simulator.[4] In our simulations, we use MCS-9 in a static fashion. This removes the impact of rate adaptation, enabling us to measure the performance improvement solely due to ViTaLS. This is a common approach in literature [9]. For our work, we choose a channel bandwidth of $80\,\mathrm{MHz}$ in the $5\,\mathrm{GHz}$ spectrum. The typical 802.11ax parameters are set as shown in Table II.

[4]Wi-Fi 6 MAC simulator - https://github.com/VinGok/Tactile-WiFi

Parameter	Value	Parameter	Value
AC_VO $[\mathrm{CW_{min}}, \mathrm{CW_{max}}]$	[32, 64]	MCS	9
AC_VI $[\mathrm{CW_{min}}, \mathrm{CW_{max}}]$	[512, 2048]	AIFS	$34\,\mu s$
AC_VO Retry limit	4	SIFS	$16\,\mu s$
AC_VI Retry limit	10	RTS, CTS	$44\,\mu s$
Max. PPDU duration	$5.4\,\mathrm{ms}$	Guard Interval	$0.8\,\mu s$
Block ACK	$44\,\mu s$	Slot size	$9\,\mu s$
BSRP, BSR, trigger	$44\,\mu s$	Aggregation	MPDU

TABLE II: 802.11ax configuration parameters used in our simulations.

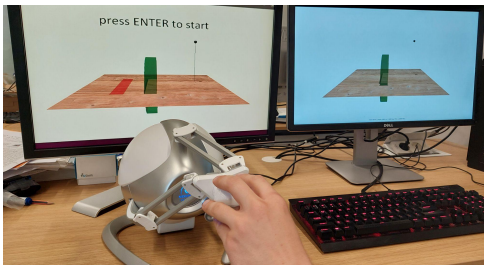


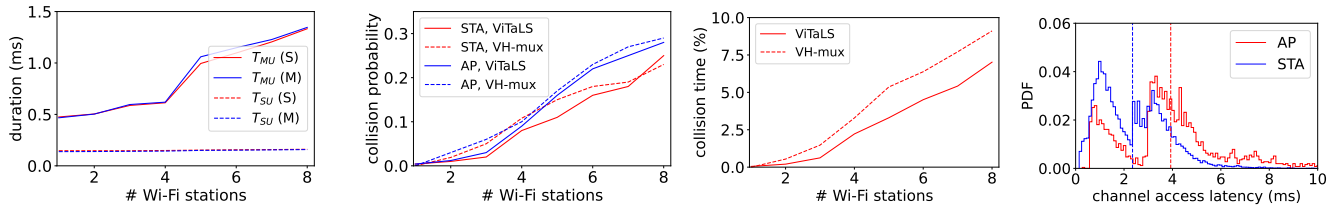
Fig. 6: Virtual environment setup showing the haptic device with the video feedback in master domain (left) and actual scene in the controlled domain (right).

The tactile traffic is generated at the standard rate of $1\,\mathrm{kHz}$. Each kinematic and haptic frame is $480\,\mathrm{B}$ and $240\,\mathrm{B}$, respectively, amounting to $5.8\,\mathrm{Mbps}$ of tactile traffic per STA. The video frames, each of size $30\,\mathrm{kB}$, are generated at $60\,\mathrm{Hz}$. This corresponds to realtime 4K video or VR traffic. Accounting for the packet header overheads, we obtain an overall traffic of roughly $25\,\mathrm{Mbps}$ per operator-teleoperator pair.

2) Subjective evaluation: For our subjective experiments, we employ TIXT – Tactile Internet eXtensible Testbed, that we developed in an earlier work [30]. The testbed encompasses real networks and tactile devices. We leverage NetEm – a standard network emulator for network latency and packet losses. We incorporate the latency and loss characteristics obtained from our simulations in the emulator. This setup provides an easy way to assess the subjective quality of ViTaLS and the VH-multiplexer.

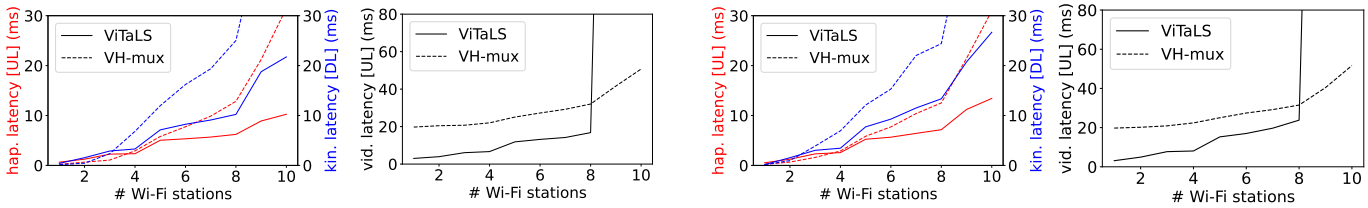
We perform TI interactions with a virtual environment (VE) setup with a standard Novint Falcon haptic device on a real network. Compared to our real teleoperation testbed (shown in Fig. 1), this approach provides consistent and easily reproducible results. We leverage a VE game that we developed in-house. The task for the participant is to interact with a VE object and move it to pre-determined target locations, as shown in Fig 6. The VE runs on a remote workstation (right-side display shown for illustration) and supplies haptic and visual feedback to the operator (left-side display).

The subjective study involved 20 participants in the age group between 17 and 53 years, with an average of 25 years. Roughly half of the participants were novice users of the haptic device. Each participant interacts with the VE under different network settings. The participants grade their TI experience



(a) (b) (c) (d)

Fig. 7: (a) Comparison between transmission durations (T_{MU} and T_{SU}) of ViTaLS as per our model (M) and simulations (S). (b) Collision probability of ViTaLS and VH-multiplexer, (c) Overall time spent in collision for the case of ViTaLS and VH-multiplexer. (d) Asymmetric channel access latency between AP and STAs along with their mean values.



(a) (b) (c) (d)

Fig. 8: Comparison of latency profiles between ViTaLS and VH-multiplexer under different communication modes. (a) tactile in basic mode, (b) video in basic mode, (c) tactile in RTS/CTS mode, and (d) video in RTS/CTS mode.

on a scale of 10 as follows:
10: no perceivable impairment; **8-9**: slight impairment but no disturbance; **6-7**: perceivable impairment, slight disturbance; **3-5**: significant impairment, disturbing; **1-2**: extremely disturbing.

B. Results

1) Objective evaluation: Owing to the ULL requirements, TI applications can only be supported in scenarios where the network load is significantly lower than the saturation throughput. In such scenarios, both ViTaLS and VH-multiplexer will yield the same network throughput. Hence, we do not show network throughput as a performance metric. However, there is a significant difference in their latency performance, which we will elaborate on in the rest of this paper. In our simulations, we empirically choose $\delta = 0.33$ unless mentioned otherwise.

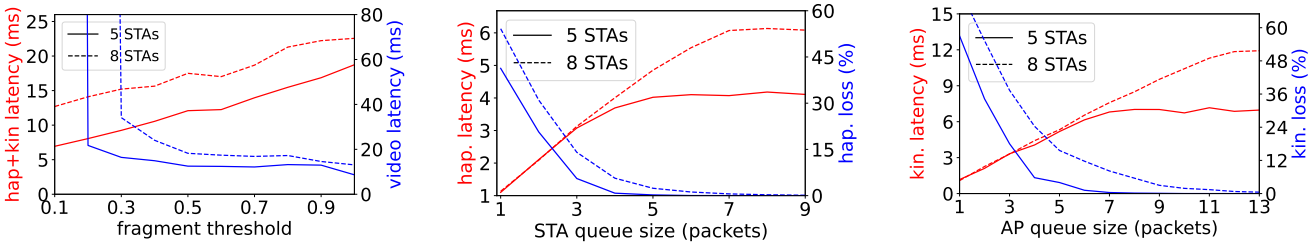
Model validation: We begin by validating our mathematical model. To match the fluid nature of traffic assumed in the model, we reduce the impulses created by video frames by smoothing the data generation process. Also, we use a single backoff stage with the retry limits specified in Table II. For a traffic setting of $15\,\text{Mbps}$ per operator-teleoperator pair, it can be seen in Fig. 7(a) that the estimations given by our model (M) for both T_{MU} and T_{SU} corroborate very well with the simulations (S). T_{MU} increases monotonically with the amount of STAs as the network load and channel contentions increase proportionately. On the other hand, T_{SU} increases rather slowly with N. This is because the SU transmissions occupy the entire channel bandwidth, which is significantly high in Wi-Fi 6 and 7 networks.

Packet collisions: We now compare the amount of packet-level collisions between ViTaLS and VH-multiplexer. In Fig. 7(b), we present the collision probabilities of AP and STAs. ViTaLS improves the collision probabilities albeit marginally as it does not modify the channel contention mechanism. The improvement, however, comes from the fact that there are more successful packets transmitted due to a reduction in the wastage of airtime. On the other hand, ViTaLS prevents the collision of video fragments to a large extent as described in Sec. III-A2. To assess this effect, we measure *collision time* – the percentage of total time that is wasted due to collisions, shown in Fig. 7(c). Taking VH-multiplexer as the baseline, ViTaLS reduces the collision time by up to 30%. Apart from prioritizing tactile frames, this significant reduction in the collision time is a major reason for the expected latency improvement of ViTaLS.

Latency measurements: We plot the PDF of the channel interaccess latency for AP and STAs along with their mean values in Fig. 7(d) for $N = 8$ and $Q_{\text{ap}} = Q_{\text{sta}} = 50$. Note that the STA interaccess latency includes both SU and MU channel accesses. As expected, each STA gets channel access much more frequently than the AP. This is because of the AP-initiated MU-UL transmissions. This implies that the haptic frames encounter significantly lower latency than the kinematic frames.[5] This is an important observation and will be utilized later in this section for optimal tactile queue sizing.

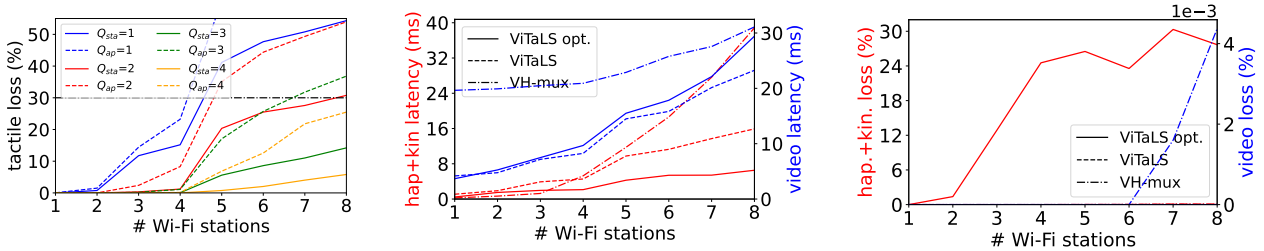
We now present the worst-case latency performance of ViTaLS and VH-multiplexer over a range of N. We take the 95$^{\text{th}}$ percentile value as the worst-case latency. In Figs. 8(a) and 8(b), we show the tactile and video latency profiles, respectively, for basic mode of transmission where data packets are transmitted without any control frames. As can be

[5]To reiterate, the haptic and kinematic frames are transmitted on UL and DL, respectively.



(a) (b) (c)

Fig. 9: (a) Evaluation of ViTaLS depending on the system parameters. Impact of (a) fragment threshold (δ) on video-tactile latency, (b) STA queue size (Q_{sta}) on haptic latency and loss, and (c) AP queue size (Q_{ap}) on kinematic latency and loss.



(a) (b) (c)

Fig. 10: (a) Tactile loss for different AP and STA queue sizes along with the loss threshold of 30%, (b) latency characteristics and (c) loss characteristics of ViTaLS, ViTaLS-optimal, and VH-multiplexer.

seen, ViTaLS comprehensively results in significantly lower latency overall with a peak reduction of up to 47% in the *two-way latency*, which is the sum of haptic and kinematic latency. Up to $N = 3$, where the amount of collisions is negligible, the tactile latency of ViTaLS and VH-multiplexer are comparable. However, the video latency of ViTaLS is significantly lower. For $\delta = 0.33$, three MU-UL channel accesses are required to transmit a video frame. On the other hand, the VH-multiplexer transmits a video frame over $\sim$17 haptic frames (video frames are generated at 17 ms intervals), and thereby takes much longer. On the other hand, it can be seen that beyond $N = 8$, the video latency of ViTaLS increases drastically. This is primarily because the chosen δ cannot match the video generation and transmission rates. A higher δ is favorable at higher network loads. On the contrary, the video latency of the VH-multiplexer is still contained, as every SU transmission also carries video fragments.

As explained in Sec. II-C, one of the reasons for the high latency of VH-multiplexer is the significant collision time. A standard method to reduce collision time is to use Request-To-Send (RTS) and Clear-To-Send (CTS) frames. To understand if this results in performance improvement of the VH-multiplexer, we enable RTS/CTS with an RTS threshold of 1 kB. The latency profiles are presented in Figs. 8(c) and 8(d). VH-multiplexer sees no performance improvement in both tactile and video latency, as RTS/CTS is known to be effective only when N is substantially higher [31]. In the remainder of the paper, we focus on the performance of ViTaLS up to $N = 8$ in basic communication mode (without RTS/CTS).

Impact of δ: In Fig. 9(a), we present the impact of δ on the latency characteristics of ViTaLS by varying δ in the range [0.1,1]. As can be seen, the two-way latency is an increasing function of δ since larger video fragments negatively impact the worst-case tactile latency. On the other hand, the video latency is a decreasing function of δ. Further, the minimum feasible value of δ for meeting the video QoS latency increases with N due to less frequent MU-UL channel access. It is important to note the trade-off between tactile and video latency, as explained in Sec. III-A. Further, given N, the two-way latency varies significantly over the sweep of δ. This suggests that choosing the optimal δ is crucial for a smooth TI experience. Although determining the optimal δ is out of the scope of this paper, we empirically choose $\delta = 0.33$ for the rest of our experiments. This guarantees minimum tactile latency while also supporting video QoS even for 8 STAs.

Impact of Q_{sta} **and** Q_{ap}: Naturally, higher Q_{sta} and Q_{ap} result in higher tactile latency and lower loss. This can be seen from Figs. 9(b) and 9(c), respectively. For $N = 5$, the loss reaches 0 % at $Q_{\text{sta}} = 4$, at which point the haptic latency saturates at around $\sim$4 ms. This implies that at most four haptic frames are queued up at the STAs, and higher Q_{sta} would overprovision the queue. Hence, $Q_{\text{sta}} = 4$ is sufficient to transmit all haptic frames without dropping any. Due to the higher channel inter-access latency at AP, Q_{ap} needs to be higher to achieve 0% loss for the same value of N. Note the difference in scale of the latency axis in Figs. 9(b) and 9(c). Further, the minimum Q_{sta} and Q_{ap} for achieving 0 % loss increase with N due to higher collisions and more queueing. These insights suggest that there is potential for further improving the latency performance of ViTaLS by trading off loss through controlling the queue sizes.

We present the tactile loss as a function of queue sizes in

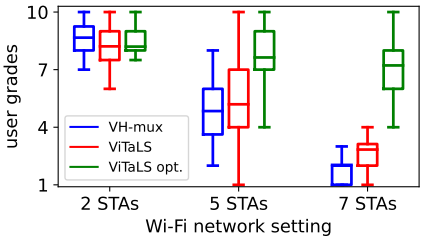


Fig. 11: User grades showing that ViTaLS-optimal outperforms VH-multiplexer and ViTaLS over different network settings.

Fig. 10(a). For $N = 1$ and 2, even a small queue size of 1 frame results in no loss. For higher N, it can be seen that for the same queue size, the AP drops more frames than STAs. This is due to the asymmetric AP and STA channel interaccess behaviors (explained earlier in Fig. 7(b)). Therefore, to achieve a target tactile loss one needs to employ different values for Q_{sta} and Q_{ap}, and further tune it depending on N for optimal performance. For instance, to utilize the 30% tactile loss target, one can set $Q_{\text{sta}} = Q_{\text{ap}} = 1$ for $N = 4$, whereas $Q_{\text{sta}} = 2$ and $Q_{\text{ap}} = 4$ is the optimal setting for $N = 8$.

ViTaLS-optimal: With the above insights, we empirically tune Q_{sta} and Q_{ap} for each setting of N to fully utilize the 30% loss target. As explained in Sec. I, this level of tactile loss is known to be imperceivable to human operators in many TI interactions [3], [4], [5], [6]. We call this version of ViTaLS as "*ViTaLS-optimal*". We now compare the performances of ViTaLS, ViTaLS-optimal, and VH-multiplexer. In Fig. 10(b), it can be seen that ViTaLS-optimal yields a reduction of up to 82% in two-way latency compared to VH-multiplexer. The advantage of exploiting the loss threshold is clearly reflected in the latency improvement. Further, the video latency of ViTaLS-optimal also improves since dropping tactile frames reduces the network load. As seen in Fig. 10(c), the tactile loss in case of ViTaLS-optimal reaches up to $30\,\%$. The non-monotonic loss behavior is because the optimal Q is fine-tuned depending on N. The video loss for all schemes is negligible.

2) Subjective evaluation: In Fig. 11, we present the user grades for the cases of VH-multiplexer, ViTaLS, and ViTaLS-optimal under three network conditions: $N = 2, 5,$ and 7. At $N = 2$, the performances of the three methods are comparable due to similar latency profiles. At higher N, the users experience a significant disturbance with both VH-multiplexer and ViTaLS, although ViTaLS provides much better objective performance – a two-way latency of $27.5\,\text{ms}$ with VH-multiplexer versus $17.2\,\text{ms}$ with ViTaLS for $N = 7$. The reason for similar subjective performance between VH-multiplexer and ViTaLS despite the objective improvement is that the above latency numbers exceed the ULL budget by a significant margin. On the other hand, ViTaLS-optimal provides a significantly higher subjective performance despite the high network load due to its ability to dynamically drop frames without causing any perceptual degradation. This further substantiates the efficacy of ViTaLS-optimal.

V. Conclusions and Future Work

In this work, we investigated the less explored problem of TI communication over Wi-Fi 6 networks. We showed conceptually and experimentally that the state-of-the-art scheduling schemes in TI fall short of satisfying the ULL requirement. To bridge this gap, we designed ViTaLS – a novel latency scheduling framework for TI. Taking VH-multiplexer as the baseline, we showed that ViTaLS reduces the tactile latency by about $47\,\%$. Further, we present ViTaLS-optimal that employs optimal queue sizes, leading to $82\,\%$ latency improvement over VH-multiplexer. Using a realistic TI testbed encompassing haptic devices and a network, we demonstrated that ViTaLS-optimal maintains a high quality user experience even under high load conditions, while the performance of VH-multiplexer deteriorates significantly. The proposed framework can be a strong candidate for making Wi-Fi 6 fit for TI communication. Further, ViTaLS-optimal can also be used to create a TI operation profile for Wi-Fi 7 systems.

In future, we would like to investigate the interplay between the parameters of ViTaLS, such as δ and Q, as well as their dependence on the network load and TI application characteristics. Additionally, we would like to focus on tuning these parameters in realtime using AI-based solutions for reaping the full benefits of ViTaLS [32]. Further, testing ViTaLS-optimal on real Wi-Fi 6 devices is an interesting research avenue.

Acknowledgment

This work has been undertaken in the *Internet of Touch* project sponsored by Cognizant Technology Solutions and Rijksdienst voor Ondernemend Nederland under PPS O&I.

References

[1] G. P. Fettweis, "The Tactile Internet: Applications and challenges," *IEEE Vehicular Technology Magazine*, vol. 9, no. 1, pp. 64–70, 2014.

[2] O. Holland, E. Steinbach, R. V. Prasad, Q. Liu, Z. Dawy, A. Aijaz, N. Pappas, K. Chandra, V. S. Rao, S. Oteafy *et al.*, "The IEEE 1918.1 "Tactile Internet" Standards Working Group and its Standards," *Proceedings of the IEEE*, vol. 107, no. 2, pp. 256–279, 2019.

[3] M. Rank, Z. Shi, H. J. Müller, and S. Hirche, "Predictive communication quality control in haptic teleoperation with time delay and packet loss," *IEEE Transactions on Human-Machine Systems*, vol. 46, no. 4, pp. 581–592, 2016.

[4] J.-y. Lee and S. Payandeh, "Forward error correction for reliable teleoperation systems based on haptic data digitization," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2013, pp. 5871–5877.

[5] J. Qin, K.-S. Choi, R. Xu, W.-M. Pang, and P.-A. Heng, "Effect of packet loss on collaborative haptic interactions in networked virtual environments: An experimental study," *Presence*, vol. 22, no. 1, pp. 36–53, 2013.

[6] H. Kroep, V. Gokhale, J. Verburg, and R. V. Prasad, "ETVO: Effectively Measuring Tactile Internet With Experimental Validation," *submitted to IEEE Transactions on Mobile Computing*, 2021.

[7] A. Nasrallah, A. S. Thyagaturu, Z. Alharbi, C. Wang, X. Shao, M. Reisslein, and H. ElBakoury, "Ultra-low latency (ull) networks: The ieee tsn and ietf detnet standards and related 5g ull research," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 1, pp. 88–145, 2018.

[8] T. Adame, M. Carrascosa-Zamacois, and B. Bellalta, "Time-sensitive networking in IEEE 802.11 be: On the way to low-latency WiFi 7," *Sensors*, vol. 21, no. 15, p. 4954, 2021.

[9] G. Naik *et al.*, "Can Wi-Fi 7 Support Real-Time Applications? On the Impact of Multi Link Aggregation on Latency," in *IEEE International Conference on Communications (ICC)*, 2021.

[10] J. Nightingale, Q. Wang, C. Grecos, and S. Goma, "The impact of network impairment on quality of experience (QoE) in H. 265/HEVC video streaming," *IEEE Transactions on Consumer Electronics*, vol. 60, no. 2, pp. 242–250, 2014.

[11] T. N. Minhas, O. G. Lagunas, P. Arlos, and M. Fiedler, "Mobile video sensitivity to packet loss and packet delay variation in terms of QoE," in *2012 19th International Packet Video Workshop (PV)*. IEEE, 2012, pp. 83–88.

[12] S. Mangold, S. Choi, P. May, O. Klein, G. Hiertz, and L. Stibor, "IEEE 802.11 e Wireless LAN for Quality of Service," in *Proc. European Wireless*, vol. 2, 2002, pp. 32–39.

[13] C.-Y. Chang, H.-C. Yen, C.-C. Lin, and D.-J. Deng, "QoS/QoE support for H. 264/AVC video stream in IEEE 802.11 ac WLANs," *IEEE Systems Journal*, vol. 11, no. 4, pp. 2546–2555, 2015.

[14] K. Kosek-Szott, M. Natkaniec, and L. Prasnal, "IEEE 802.11 aa intra-AC prioritization-A new method of increasing the granularity of traffic prioritization in WLANs," in *2014 IEEE Symposium on Computers and Communications (ISCC)*. IEEE, 2014, pp. 1–6.

[15] G. Tian, S. Camtepe, and Y.-C. Tian, "A deadline-constrained 802.11 MAC protocol with QoS differentiation for soft real-time control," *IEEE Transactions on Industrial Informatics*, vol. 12, no. 2, pp. 544–554, 2016.

[16] L. Lanante, C. Ghosh, and S. Roy, "Hybrid OFDMA random access with resource unit sensing for next-gen 802.11 ax WLANs," *IEEE Transactions on Mobile Computing*, vol. 20, no. 12, pp. 3338–3350, 2020.

[17] B. Cizmeci, X. Xu, R. Chaudhari, C. Bachhuber, N. Alt, and E. Steinbach, "A multiplexing scheme for multimodal teleoperation," *ACM Trans. Multimedia Comput. Commun. Appl.*, vol. 13, no. 2, pp. 21:1–21:28, Apr. 2017.

[18] V. Gokhale, J. Nair, and S. Chaudhuri, "Congestion control for network-aware telehaptic communication," *ACM Trans. Multimedia Comput. Commun. Appl.*, vol. 13, no. 2, pp. 17:1–17:26, Mar. 2017.

[19] H. Al Osman, M. Eid, R. Iglesias, and A. El Saddik, "Alphan: Application layer protocol for haptic networking," in *2007 IEEE International Workshop on Haptic, Audio and Visual Environments and Games*. IEEE, 2007, pp. 96–101.

[20] M. Eid, J. Cha, and A. El Saddik, "Admux: An adaptive multiplexer for haptic–audio–visual data communication," *IEEE Transactions on Instrumentation and Measurement*, vol. 60, no. 1, pp. 21–31, 2010.

[21] Y. Feng, C. Jayasundara, A. Nirmalathas, and E. Wong, "A feasibility study of IEEE 802.11 HCCA for low-latency applications," *IEEE Transactions on Communications*, vol. 67, no. 7, pp. 4928–4938, 2019.

[22] D. Cavalcanti, J. Perez-Ramirez, M. M. Rashid, J. Fang, M. Galeev, and K. B. Stanton, "Extending accurate time distribution and timeliness capabilities over the air to enable future wireless industrial automation systems," *Proceedings of the IEEE*, vol. 107, no. 6, pp. 1132–1152, 2019.

[23] V. Gokhale, M. Eid, K. Kroep, R. V. Prasad, and V. S. Rao, "Toward Enabling High-Five Over WiFi: A Tactile Internet Paradigm," *IEEE Communications Magazine*, vol. 59, no. 12, pp. 90–96, 2021.

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

[25] D. Magrin, S. Avallone, S. Roy, and M. Zorzi, "Validation of the ns-3 802.11 ax OFDMA Implementation," in *Proceedings of the Workshop on ns-3*, 2021, pp. 1–8.

[26] B. Bellalta and K. Kosek-Szott, "Ap-initiated multi-user transmissions in ieee 802.11 ax wlans," *Ad Hoc Networks*, vol. 85, pp. 145–159, 2019.

[27] G. Bianchi, "Performance analysis of the ieee 802.11 distributed coordination function," *IEEE Journal on selected areas in communications*, vol. 18, no. 3, pp. 535–547, 2000.

[28] I. Tinnirello and G. Bianchi, "Rethinking the ieee 802.11 e edca performance modeling methodology," *IEEE/ACM transactions on networking*, vol. 18, no. 2, pp. 540–553, 2009.

[29] S. Youm and E.-J. Kim, "Latency and jitter analysis for ieee 802.11 e wireless lans," *Journal of Applied Mathematics*, vol. 2013, 2013.

[30] V. Gokhale, K. Kroep, V. S. Rao, J. Verburg, and R. Yechangunja, "TIXT: An extensible testbed for tactile internet communication," *IEEE Internet of Things Magazine*, vol. 3, no. 1, pp. 32–37, 2020.

[31] O. Sharon and Y. Alpert, "The combination of QoS, aggregation and RTS/CTS in very high throughput IEEE 802.11 ac networks," *Physical Communication*, vol. 15, pp. 25–45, 2015.

[32] S. Szott, K. Kosek-Szott, P. Gawłowicz, J. Torres Gómez, B. Bellalta, A. Zubow, and F. Dressler, "Wi-Fi Meets ML: A Survey on Improving IEEE 802.11 Performance with Machine Learning," *IEEE Communications Surveys & Tutorials*, 6 2022.