2022 ACM/IEEE 13th International Conference on Cyber-Physical Systems (ICCPS)

Blind Spots of Objective Measures: Exploiting Imperceivable Errors for Immersive Tactile Internet

H. J. C. Kroep
Delft University of Technology
H.J.C.Kroep@tudelft.nl

V. Gokhale
Delft University of Technology
V.Gokhale@tudelft.nl

R. Venkatesha Prasad
Delft University of Technology
R.R.VenkateshaPrasad@tudelft.nl

***Abstract*—*Tactile Internet (TI)* enables the transfer of human skills over the Internet, enabling teleoperation with force feedback. Advancements are being made rapidly at several fronts to realize a functional TI soon. Generally, TI is expected to faithfully reproduce operator's actions at the other end, where a robotic arm emulates it while providing force feedback to the operator. Performance of TI is usually characterized using objective metrics such as network delay, packet losses, and RMSE. *Pari passu*, subjective evaluations are used as additional validation, and performance evaluation itself is not primarily based on user experience. Hence objective evaluation, which generally minimizes *error* (signal mismatch), is oblivious to subjective experience.**

In this paper, we argue that user-centric designs of TI solutions are necessary. We first consider a few common TI errors and examine their *perceivability*. The idea is to reduce the impact of perceivable errors and exploit the imperceivable errors to our advantage, while the objective metrics may indicate that the errors are high. To harness the imperceivable errors, we design *Adaptive Offset Framework (AOF)* to improve the TI signal reconstruction under realistic network settings. We use AOF to highlight the contradictory inferences drawn by objective and subjective evaluations while realizing that subjective evaluations are closer to ground truth. This strongly suggests the existence of 'blind spots of objective measures'. Further, we show that AOF significantly improves the user grade, up to 3 points (on a scale of 10) compared to the standard reconstruction method.

***Index Terms*—Tactile internet, user experience, QoS, teleoperation**

I. Introduction

Through seamless audio-visual communication, today's Internet has transcended the physical barriers between remotely located humans. The pioneering idea of *Tactile Internet (TI)* envisions to further diminish this physical separation by enabling *transfer of skills* [1]. The cornerstone of TI is the fast and accurate replication of human operator's actions by a robotic teleoperator in the remote environment and the transmission of haptic (force) feedback generated thereof to the operator. Such an ability to teleoperate will undoubtedly empower us to perform complicated tasks, especially in hard-to-reach remote locations, as if we are present there.

Due to its vision of carrying human skills anywhere, TI has the potential to completely revolutionize several domains such as healthcare, disaster response, edutainment, and e-commerce. The most celebrated applications are telesurgery, where a surgeon conducts medical operations on a remote patient, and remote disaster management, such as cleaning up the Fukushima nuclear disaster site. In such human-in-the-loop applications, the *master domain* comprising of the operator transmits kinematic signals (position and velocity) for teleoperating the robot in the *controlled domain*. Haptic sensation generated due to the physical interaction is transmitted back to the operator along with audio-visual feedback. A simple depiction of TI is in Fig. 1, where both the operator and the teleoperator (robotic arm) are shown.

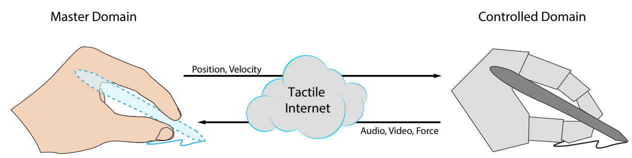


Fig. 1: Depiction of a typical Tactile Internet system highlighting the master and controlled domains and data communication between them. In light blue the pen is indicated in the master domain to be present only through haptic and visual feedback, while the real pen is only present in the controlled domain.

Challenges: Realizing a functional TI necessitates addressing a plethora of challenges. Firstly, for effective teleoperation, the transmission of kinematics and haptic feedback requires ultra-reliable low latency communication (URLLC); the widely prescribed requirements being sub-10 ms latency and 99.999 % reliability [2]. Achieving URLLC is extremely challenging. Secondly, the human-in-the-loop nature of TI applications poses additional complexities. Due to the safety-critical nature of TI applications, accuracy in task execution and user (operator) experience are of paramount importance. The operator's perception and response dictate these factors, which are increasingly complicated to model due to the subjectivity involved. Hence, quantification of TI performance is challenging, and we now list some of the issues.

In a realistic teleoperation scenario, signal errors (deviations from the expected signal trajectory) are inevitable. Some example causes of errors include delay, information loss, and heterogeneous workspaces and capabilities between master and controlled domains. These could be added by sensing, actuation, compression, communication, and other processing modules. Existing TI solutions have leveraged several conventional objective metrics for directly or indirectly indicating/measuring/estimating these errors [3], [4], [5], [6]. Such metrics include network delay, packet losses, throughput,

978-1-6654-0967-4/22/$31.00 ©2022 IEEE
DOI 10.1109/ICCPS54341.2022.00011

and Root Mean Square Error (RMSE). Existing methods in the literature indirectly strive to minimize the *error* (mismatch between sensed and reconstructed signals) as a whole based on the above objective metrics. Going by the literature, there is a heavy reliance on objective metrics as the key performance indicators (KPIs), while subjective evaluations (user grades) are, most often, used only for additional validation of the objective results. The rationale for such a notion is reasonable and, to a large extent, justified as objective studies are controllable and repeatable, and they work well for humans in open-loop systems. However, some crucial limitations surface when we work with human-in-the-loop TI systems.

Issues with TI performance characterization. There are two major limitations of above performance characterization techniques.

1. Existing works implicitly assume that *users are equally sensitive to all kinds of errors* without distinguishing different types of errors. In other words, there is a lack of understanding of how humans perceive and respond to different types of error. This has led to a counterproductive effect – some TI solutions attempt to rapidly minimize error at the expense of introducing perceptual artifacts that further hampers performance (demonstrated in detail in Sec. V-A).
2. This may be catastrophic for safety-critical TI applications. There could exist several insensitivities of humans to specific errors. Not studying them may cause us to miss out on promising opportunities that could significantly boost TI performance and relax the stringent URLLC constraints, **which are mostly unnecessary**. Hence, substantial efforts are necessary to understand human perception, thereby designing user-centric TI solutions that leverage this knowledge rather than just treating the user as a black box.

The above observations lead us to the following crucial questions. 1 How impactful are different types of errors from the standpoint of user experience? 2 If there exist imperceivable errors, how can we uncover and exploit them to enhance TI communication? 3 Are standard objective metrics holistic and sufficient to capture these effects and characterize overall performance? We believe that answering these questions opens new directions in characterizing and building systems that support TI communication.

Our approach: We take a radically different approach from the existing works to seek answers to the above questions. We analyze the impact of certain types of errors through a user-centric study. To leverage these insights for enhancing TI performance, we come up with the design of *Adaptive Offset Framework (AOF)* – a novel TI signal reconstruction technique for intelligently handling errors introduced in a TI system.

Contributions. Our contributions in the process are listed below.

1. Motivated by the need of a user-centric approach, we examine a few common errors that occur in TI scenarios. We develop the concept of '*perceivability of errors* to indicate the perceptual significance of errors.
2. To leverage the insights gained, we propose *Adaptive Offset Framework (AOF)* that dynamically adjusts the position offset between master and controlled domains for producing a smooth reconstruction signal without introducing any perceptual impairments (Sec. III). By implementing AOF in a realistic TI setup involving human subjects, we present examples to illustrate the working of AOF.
3. We measure the TI performance both objectively and subjectively over a wide range of network settings. As per the objective measurements, AOF performs worse than standard reconstruction methods. However, the subjective measurements suggest otherwise, revealing that AOF significantly improves the user experience (Sec. V-C).
4. The glaring contrast between subjective and objective measures expose 'blind spots' of objective measures. Further, the subjective measurements indicate that AOF significantly improves the overall TI performance. (Sec. V-B).

II. Errors in TI and their perceivability

Several types of mismatch (*error*) can exist while reproducing a sensed signal in a TI system that could heavily influence the performance. To interpret these errors, we introduce the notion of *decomposition of errors* in a TI system. We consider the most common TI errors between master and controlled domains and examine their impact on user performance.

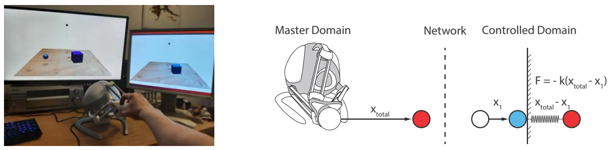
A. Typical TI Setup

We begin by first describing a typical TI setup. A haptic device resides in the master domain. A teleoperator, often a robotic arm, resides in the controlled domain in a remote physical environment. We use a Novint Falcon in the master domain and a virtual environment (VE) in the controlled domain for our experiments. The TI setup in our laboratory is shown in Fig. 2(a). When a point on the haptic device is moved (from white to red location), the corresponding part of the teleoperator in the VE, known as *haptic interaction point (HIP)*, moves accordingly [7]. If a rigid object in the VE is at a distance of x_1 from the HIP, then the HIP applies a force F proportional to penetration depth $(x_{total} - x_1)$, when the device displacement is x_{total}, i.e., $F - k(x_{total} - x_1)$ where k is the spring constant. This is illustrated in Fig. 2(b). The experienced force is measured with sensors and fed back to the operator through the haptic device. We opt to use a virtual physics environment to represent the controlled domain. The key advantages of using a virtual physics environment are the complete access of all information in the controlled domain and repeatable experimentation that provides all participants a consistent and reproducible experience. Note that physics interaction calculations in the VE reflect the general behavior in a physical environment.

Currently, a TI application cannot be facilitated by a network with ideal performance. We investigate these errors to better understand their impact on the overall performance.

B. Decomposition of Error

An ideal TI system must not generate any error, i.e., sensed signals in one domain are reproduced with neither temporal



(a) (b)

Fig. 2: (a) Tactile Internet setup in our lab showing the the user interacting with a virtual environment. The monitor on the left and right correspond to master and controlled domains, respectively. (b) Conceptual illustration of generation of haptic feedback. HIP is the white circle in the controlled domain.

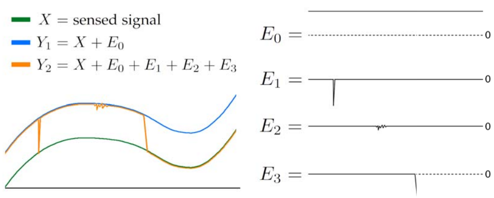


Fig. 3: Illustration of decomposition of errors on two estimated signals Y_1 and Y_2.

error (zero delay) nor spatial (position) error. Such a TI system can be described as,

$$Y[k] = X[k],$$

where $X[k]$ is the sensed sample in one domain at time k and $Y[k]$ is the estimated value at the other domain at time k. However, in practice, signal reproduction is prone to errors. Therefore, a practical TI system can be represented by,

$$Y[k] = X[k + l[k]] + E[k], \tag{1}$$

where $E[k]$ represents the position error at time k, and $l[k]$ is the temporal error at time k. Note that these errors are themselves dependent on k. Although in Eq. (1) we represented the error conceptually as a whole, we can improve the analysis by considering temporal and position errors as multiple components acting simultaneously. In other words, $E[k]$ and $l[k]$ can be decomposed as,

$$E[k] = \sum_m E_m[k] \text{ , and } l[k] = \sum_n l_n[k], \tag{2}$$

where $E_m[k]$ is the m^{th} component of $E[k]$ and $l_n[k]$ is the n^{th} component of $l[k]$. It should be noted that in Eq. (2) we do not define any correlation between errors for the sake of simplicity. However, error components can be correlated with the sensed signal or other error components. Therefore, there is an unlimited number of ways to decompose $E[k]$. We illustrate this concept in Fig. 3. Here, two estimations Y_1 and Y_2 of sensed signal X are shown. Y_1 appears to be identical to X apart from a stationary offset, while Y_2 has multiple deviations. An example of how the error in Y_2 can be decomposed into multiple components is shown with E_0, E_1, E_2, and E_3. Any decomposition can be considered as long as their sum matches the total.

This notion of error decomposition allows us to isolate errors and examine their impact on the operator separately. The objective of this work is not to extract error components from the signal but to consider some common errors that provide a scope for improving TI performance. We now consider some common errors that occur in TI communication and examine their impact on the user experience.

C. *Perceivability of errors*

We now consider a few types of errors that offer us the most interesting opportunities to improve the TI performance. We examine the *perceivability of errors* which is the impact of an error on the user experience. To this end, we consider three specific types of errors.

1) Stationary offset: A mapping between the master and controlled domains is defined for reproducing the operator's actions. This causes any unique location in the master domain to point to a unique location in the controlled domain. The choice of this mapping is heavily dependent on the application. There can be an application where the operator spans the entire workspace of the teleoperator and another application where the operator is interested in fine-grained movements in a limited portion of the workspace. In any case, the operator learns the deployed mapping by interacting with the TI setup. Let us consider the former scenario in which each point in the teleoperator's workspace is uniquely mapped to a point in the operator's workspace and vice-versa. Let us suppose there is a stationary offset of $2\,\mathrm{cms}$ in the mapping, and the operator intends to pick and place an object in the controlled domain. If the operator can perform all actions as intended, the offset does not pose any issues. On the other hand, if the teleoperator (HIP in case of VE) is a few centimeters away from the object while the operator has reached the workspace edge and can not move any further, then the offset starts to make a negative impact, and this is undesirable.

The operator realizes the offset only due to the presence of *reference points*. In the above example, the workspace edge acts as the reference point. This is illustrated in Fig. 4. Here, the blue cube indicates the workspace of the haptic device. An example of a good map from the operator to the teleoperator's workspace is the green cube in the controlled domain. However, a stationary offset results in the red cube being the teleoperator's workspace. If the operator intends to touch the remote object (vase), the teleoperator can never really allow that since it can access only a portion of the object. However, the interaction would have been satisfactory if the object resided in the green and red cubes' overlapping regions. The reference point, in this case, is a combination of the workspace and the desired area of operation.

From the above examples, it can be observed that reference points play a significant role in governing the perceivability of stationary offsets. If the offset is small with respect to the reference points, then the offset can be considered imperceivable to the operator. However, the same offset can be a significant error for objective metrics.

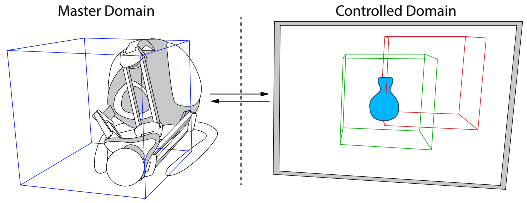


Fig. 4: Illustration of the notion of reference points. The blue cube indicates the operator's workspace, whereas the green and red cubes indicate the desired workspace and stationary offset workspace of the teleoperator, respectively. In the offset case, the object (blue vase) can never be fully accessed, leading to performance issues.

2) Velocity Scaling error: When the operator performs an action, if the teleoperator moves considerably faster or slower than the operator, then it becomes perceivable. For example, if the operator moves the hand, and the teleoperator barely moves, this will be highly perceivable. However, small variations in the scale of the velocity of the teleoperator's movements will be imperceivable.

3) Delay-induced position errors: Due to the inherent TI delay (network, processing, among others), there would be a lag in replicating the operator's actions, leading to position errors. The presence of haptic feedback strongly determines how perceivable these position errors are. When the HIP is distant from the objects in the VE, there is no haptic feedback. Hence, position errors corresponding to even a few milliseconds between master and controlled domains will not cause any disturbances in the operator's ability to teleoperate. On the other hand, if the HIP is in the vicinity of or in contact with VE objects, these position errors could create undesirable haptic feedback. For example, there is a sharp transition between free space and hard object since the force rises rapidly from zero (free space) to a considerable value (on the object's surface). Hence, even minor position errors can harm the user experience. Suppose the operator is transitioning from free space to hard object. If the force feedback is delayed, then the operator would have applied a large force to the object before force feedback is experienced. The large penetration generates a high force that could impair the operator's teleoperation ability. Hence, minor delay-induced position errors could be highly detrimental.

D. Blind spots in objective measures

So far, we have explored multiple errors and their perceivability for a human operator. To find opportunities that are not explored in the state-of-the-art in TI, we focus on errors that either have (i) *a large impact on objective measures but a small impact on the user experience* or (ii) *a small impact on objective measures but a large impact on the user experience*.

Stationary offset and velocity scaling error belong to the former category, and delay-induced position error belongs to the latter. Clearly, in these cases, contradictory inferences are drawn by objective measures and user experience. Since user experience is the KPI in TI systems, any measure that does not agree with it manifests severe shortcomings with respect to performance characterization. Hence, we argue that there exist *blind spots* in objective measures, which is their inability to characterize TI performance properly. As an example, we take the stationary offset discussed in Sec. II-C1. Objective measures based on network parameters are agnostic to the underlying data. Therefore, there is no way to identify any error term. However, that does not mean those network parameters are not useful. An increased delay and increased information loss will undoubtedly deteriorate the system's performance. However, it does have blind spots to pinpoint the performance more accurately.

A simple objective measure that is signal-aware is RMSE. A stationary offset will cause a significant increase in the RMSE, which can marginalize other deterioration like high-frequency noise. This is a blind spot within RMSE, causing it to significantly drop in effectiveness when any form of stationary offset is present. Another example is the delay-induced position errors described in Sec. II-C3. Here we identify that force feedback significantly impacts the consequence of an error. Objective measures based on network parameters or RMSE can identify a delay or an error but do not consider their effect on the physical environment. This concept is another blind spot in objective measures.

We now leverage the insights gained on perceivability of errors to improve user experience. To this end, we propose *Adaptive Offset Framework (AOF)* for reconstructing a smooth kinematics signal in the controlled domain.

III. Adaptive Offset Framework (AOF)

In Sec. II, a small stationary offset was deemed almost imperceivable, with the caveat that the offset should be sufficiently small with respect to potential reference points. This observation provides us with a range of stationary offsets that can be maintained indefinitely without affecting the user experience. This range can be deployed as an *adaptive offset*. The adaptive offset can be deployed just before the estimation is used. With this, we get

$$Z[k] = Y[k] + A[k], \tag{3}$$

where $Z[k]$ is the reconstructed value at time k and $A[k]$ the state of the adaptive offset at time k. $Y[k]$ is the estimation as defined in Eq. (1). Whenever errors are identified in the system, they can be absorbed into the adaptive offset instead of correcting for them directly. The error can then be handled at a later time. This enables us to address these errors at the opportune moment.

The adaptive offset by itself does not provide an improvement to the user experience. We introduce *shaping functions* that modify the adaptive offset to improve the user experience. Their intended purpose is to mask *errors with a small impact on objective measures but a large impact on the user experience* so that with minimal dependence on the adaptive offset, they improve the user experience. At the same time, we need to prevent the offset from ever-increasing. To this end, we introduce *decay functions* whose primary goal is to

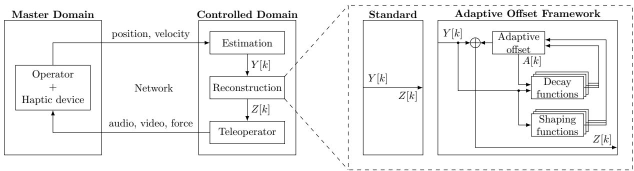


Fig. 5: Block diagram representation of the proposed AOF solution. Also shown is the contrast with standard reconstruction methods in TI literature.

shrink and contain the offset, by utilizing *errors with a large impact on objective measures but a small impact on the user experience*. Multiple shaping and decay functions can be active simultaneously. We define the adaptive offset as

$$A[k] = A[k-1] + \sum_p S_p[k] + \sum_q D_q[k], \tag{4}$$

where $S_p[k]$ is the contribution of the p^{th} shaping function at time k, and $D_q[k]$ the contribution of the q^{th} decay function at time k. The interconnection between the different modules of AOF is shown in Fig. 5.

A. Shaping function for one-shot correction error

An offset can be built up when new information is not received at the controlled domain. When new information eventually arrives, the standard method is to adjust to the new information as soon as possible, causing all offset to be removed in one shot. We define this change as a correction. The benefit of the correction is clear. It removes all of the position offset in the system by going to the intended position. However, the problem is that these corrections can result in short spikes in the velocity that were not present in the original signal. We call this spike in velocity a *one-shot correction error*. The one-shot correction error is experienced as an impulse by the user. If this happens in the vicinity of a physical object, a large spike in force feedback can be experienced. These effects are highly perceivable, especially when the spike in force feedback is sufficiently large. When the corrections are sufficiently large, these can be perceived even visually. The effects are even more pronounced in the presence of delay and packet losses.

The one-shot correction error can be removed entirely by subtracting an equal amount of the offset from the buffer when the correction is performed. Instead of a high spike in velocity, the adaptive offset is altered. To do this, the correction needs to be calculated before it can be committed to the buffer. When calculating a new reconstruction after a new packet was received, one should not calculate the correction for the upcoming step but the correction needed in the previous step. That way, the signal maintains its velocity correctly. $S_{\text{corr}}[k] = Y[k-1] - X[k-1]$, where $k-1$ indicates the time of the previous estimation and the arrival of the latest packet. S_{corr} is the shaping function that targets the correction error.

Depending on the quality of the network, removing the corrections can create a large amount of pressure on the adaptive offset. Therefore an option to tune the aggressiveness of the shaping function is useful. One way to use the same concept less aggressively is to introduce a threshold τ_{corr}. Any corrections smaller than a certain amount are deemed acceptable, but anything larger could hurt the user experience. This also potentially works well when packet loss is present, which can sometimes cause potentially harmful corrections. The resulting shaping function is defined as,

$$S_{\text{corr}}[k, \tau] = \begin{cases} 0 & \text{if } \|S_{\text{corr}}[k, 0]\| < \tau_{\text{corr}}, \\ S_{\text{corr}}[k] - \tau_{\text{corr}} \hat{S}_{\text{corr}}[k] & \text{otherwise,} \end{cases} \tag{5}$$

where τ_{corr} is the threshold and $\hat{S}_{\text{corr}}$ the unit vector of S_{corr}.

B. Shaping function for delay

Inevitably there exists a delay between the master and controlled domain. This delay can be caused by the Round Trip Time (RTT), information loss, or other methods. Multiple ways can be considered to decrease this problem. For example, future predictions can be considered. However, prediction does not come for free and runs the risk of instability. For this work, we will consider a different approach to suppress the effects of delay.

In order to decrease the adverse effects of delay on the user experience, a more specific error needs to be found. We can use a concept discussed previously, where position errors are more perceivable when interacting with physics objects. A delay allows the operator to move through solid objects like walls without feeling force feedback for a short period. When the force feedback does arrive, the operator has already moved deep into the object, causing the device in the controlled domain to apply a lot more force to the physical object than the operator intended.

There is a way for the controlled domain to recognize instances when the force feedback in the system and the force feedback experienced by the operator have a mismatch. When the force feedback changes, the controlled domain knows this before the operator in the master domain. The system can keep track of the information experienced by the operator, and with that information available, a shaping function can be crafted.

Let $F_{\text{controlled}}[k]$ be the sensed force in the controlled domain at time k. Then we take $F_{\text{master}}[k]$ as the force feedback experienced by the operator in the master domain at time k. The difference in force can be calculated as

$$F_{\text{difference}}[k] = F_{\text{controlled}}[k] - F_{\text{master}}[k]. \tag{6}$$

We now make the assumption that the effect of $F_{\text{difference}}$ would result in an amount of velocity, would the operator have experienced it. We can then proactively apply the effects of that velocity by modifying the buffer. The resulting shaping function becomes

$$S_{\text{delay}} = f_{\text{delay}}(F_{\text{difference}}), \tag{7}$$

where S_{delay} is the shaping function, and f_{delay} is a function indicating the amount of velocity as a result of the force difference.

With the shaping function stated above, the kinematic data is slowed down as it moves through a rigid object. Of course, this ”slowing down” only happens at the receiver. The operator is not affected. The goal is to suppress a potentially unintended spike in force. However, there is a high risk of a positive feedback loop: the pressure is lessened because of the sensed increase in force. Then the operator feels less force as a result and slows down less. Once again, the sensed force is increased. With this loop, the wall can appear very weak.

To make sure the effect of S_{delay} is as desired, f_{delay} should be chosen appropriately. Additionally there is a consideration in how the $F_{\text{controlled}}$ and F_{master} are obtained. We propose to deploy a shift register at the receiver that keeps track of recent force measurements. Through communication a Round Trip Time (RTT) can be obtained. Based on the RTT, F_{master} can be chosen from the shift register. We then choose $f_{\text{delay}}(F_{\text{difference}}) = C_{\text{delay}} \cdot F_{\text{difference}}$, where ${}_{\text{delay}}C$ is a constant that linearly correlates the difference in force with a velocity. For a fixed RTT, this naturally balances the offset created by S_{delay}. For an infinitely long session, if it ends with a period of zero force, the cumulative effect of S_{delay} will be zero. This property reduces the pressure of S_{delay} on the adaptive offset. Still C_{delay} needs to be chosen carefully so it has a noticeable effect, but not so high that it makes the walls feel flimsy.

C. Decay functions

Besides shaping functions, we need adequate decay functions to handle the pressure shaping functions apply to the adaptive offset. The combined effort of all deployed decay functions should handle the pressure provided by all deployed shaping functions.

1) Velocity scaling Decay function: As shown in Sec. II, a stationary scaling error is almost imperceivable. By extension, slightly scaling the velocity at run-time is also hard to perceive. This provides an excellent opportunity to shrink the adaptive offset.

Any time the pointer moves, the component in the direction of the adaptive offset can be slightly scaled, either increasing or decreasing the movement slightly. When there is a non-zero velocity in the system, this function provides a steady shrinking of the adaptive offset. We first need to project the movement and we can project the velocity onto the adaptive offset.

$$\dot{X}_{\text{projection}}[k] = \frac{\dot{X}[k] \cdot A[k]}{\|A[k]\|} \frac{A[k]}{\|A[k]\|} \tag{8}$$

where $\dot{X}_{\text{projection}}[k]$ is the projected velocity. Now we scale the projection depending on it matches or opposes the direction of the adaptive offset.

$$D_{\text{scaling}}[k] = \begin{cases} C \cdot \dot{X}_{\text{projection}}[k] & \text{if } \frac{\dot{X}[k] \cdot A[k]}{\|A[k]\|} \geq 0, \\ \frac{-C}{1+C} \cdot \dot{X}_{\text{projection}}[k] & \text{otherwise.} \end{cases} \tag{9}$$

Here D_{scaling} is the decay function based on scaling velocity, and C_{scaling} a scalar indicating the strength. To choose a good value for C_{scaling} we can use the same concept deployed in Perceptual Deadband. A concept called Just Noticeable Difference (JND) indicates how much a velocity can differ before the operator notices it. Typically this value is stated as 10%. Here the same value can be used for C_{scaling}.

An alternative method is to make C_{scaling} dependent on the size of the adaptive offset. The idea is that there is not a strong need for the decay function to act for a small adaptive offset, but when the size is relatively big, the offset should be suppressed more strongly.

$$C_{\text{scaling}}[k] = \frac{2\|A[k]\|}{B_{\max}}, \tag{10}$$

where $B_{\max}$ is the maximum size of the buffer. A $C_{\text{scaling}} = 2$ means that all velocity in the direction of the offset will be completely nullified. It is possible to choose different functions for C_{scaling}, but exploring other options remains future work.

With the above described shaping and decay functions, we arrive at a specific implementation of the Adaptive Offset Framework, which can be expressed as

$$A[k] = A[k-1] + S_{\text{corr}}[k, \tau] + S_{\text{delay}}[k] + D_{\text{scaling}}[k]. \tag{11}$$

There is sufficient scope to explore other functions to further improve the user experience.

IV. Experimentation

A. Experimental Setup

As explained in Sec. II-A, in this work, we use a VE setup for our experimentation. Note that AOF's working and performance evaluation also applies to remote physical environments. Virtual physics exist within conventional game engines. Conventional game engines calculate physical parameters linked to the frame rate, typically 60 Hz. This rate is insufficient as haptic signal requires 1 kHz update rate. The TI testbed proposed by Bhardwaj et al. [8] solves this problem. The game engine used is Chai3D, which has a physics engine detached from the frame rate, and runs physics calculations at the required 1 kHz.

As shown in Fig. 2(a), in our the master domain houses the haptic device and a monitor. The controlled domain houses the VE, and the two domains are connected via a real network. In the master domain, the force is fed to the haptic device. The kinematic data of the dynamic objects are used to update virtual copies of those dynamic objects stored locally. The rendering engine then produces frames locally.

We develop a VE game that we call *FollowMe* where the task is for the user to track a continuously moving target using the haptic device. A snapshot of the FollowMe game is shown in Fig. 6. The demo was designed with minimal VE objects to ensure a consistent experience across different participants. The only objects are a rigid, immovable surface (‘C’ in the figure) and a slider (‘B’ in the figure) that can be moved using the haptic device.

In order to test the efficacy of AOF, we perform experiments under a wide variety of different network behaviors. We use Netem – a standard network emulation tool to consistently and precisely emulate various network conditions. A consistent behavior is desirable as it helps reduce variance in performance

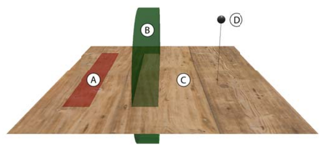


Fig. 6: A snapshot of the *FollowMe* game used for the performance evaluation of AFO. The target 'A' needs to be tracked by moving the green object indicated as 'B' using the haptic device. 'C' is a rigid surface like a table and 'D' is the HIP.

and isolate the effects of AOF. We consider network delay, uniform packet loss, and bursty packet loss settings. Bursty packet loss is induced using the Gilbert-Elliot model.

We also deploy Perceptual Deadband (PD) [5] – a state-of-the-art compression scheme for haptics signals. PD works by estimating the perceptually insignificant samples. The transmitter can avoid sending such samples leading to improvement in application bandwidth requirement. For example, a PD of $15\,\%$ implies that a sample is transmitted only if the percentage change in magnitude with respect to the previous transmitted sample is higher than $15\,\%$.

B. Experimental Procedure

The goal of the experiment is to investigate the effect of different forms of performance degradation on the user experience. In this regard, we consider three specific tasks: 1 *Pushing* the slider at a steady rate. This motion helps in recognizing subtle disturbances due to PD and packet loss. 2 *Dropping* the HIP on the surface from a height as there is a sharp transition in generated force. This motion is like resting a hand on a table. 3 *Rubbing* the rigid surface. There is both a steep transition in force and smooth motion. Participants are requested to experiment with all three actions to get a more inclusive idea of the user experience in a more realistic scenario. Participants are given time to familiarize themselves with the application, typically five minutes.

Participants are presented with ten sets of network settings in random order. Once a setting is chosen, it is given twice – once with AOF and the other with standard behavior in a random order, as explained in Fig. 5. Hence, there are 20 different scenarios altogether. For every setting, the target travels a predefined trajectory for 20 seconds. At the end of each setting, the participant grades the experience as per Table I. The subjective study involved fifteen participants in the age group between 20 and 64 years, with an average of 32 years. No participant suffered from known neurological disorders.

TABLE I: Description for subjective grading.

10	no perceivable impairment
8-9	slight impairment but no disturbance
6-7	perceivable impairment, slight disturbance
4-5	significant impairment, disturbing
1-3	extremely disturbing

V. Performance Evaluation

A. AOF behavior analysis

To illustrate the working of AOF, we first present some examples, shown in Fig. 7 . Each data set is created within the experimental setup. At $0\,\mathrm{cm}$ on the vertical axis, there is a rigid surface extending downwards. We plot the teleoperator position in the vertical axis while TI interaction is being conducted. We measure the generated force based on reconstructed position as explained in Sec. II-A. Figs. 7(a) and 7(b) correspond to a *dropping motion* where the operator attempts to put drop the device down on the rigid surface while being subjected to $10\,\mathrm{ms}$ of RTT. Figs. 7(c), 7(d), 7(e), and 7(f) correspond to a *rubbing motion* where the operator attempts to rub the device over the rigid surface. The difference between AOF and standard behavior is described in Fig. 5. We will explain some of the key performance benefits of AOF from these examples.

1. Suppression of oscillations. In Fig. 7(a), one can see that the dropping motion causes significant oscillations with the standard reconstruction method. This effect is indicated by marker ❶. Due to the delay, the user enters the surface without feeling the force feedback, thus penetrating deeper before the force feedback arrives. The force feedback is larger than desired as the penetration depth is larger. This causes the user to be pushed out of the surface quicker. When the operator continuously applies downward force, this causes oscillations. This is typical of TI because of the delay in the network, which otherwise would not be physically possible. Fig. 7(b) corresponds to AOF and the function $S_{\text{delay}}[k]$ is active. $F_{\text{difference}}$ is linearly proportional to a velocity added to the adaptive offset. The velocity slows down when the user moves into the surface, and the force feedback has not yet been experienced. Consequently, the surface is penetrated less deep than before, and a lower force is generated. Consequently, the operator is forced less aggressively out of the surface. When the operator applies constant downward pressure, the forces in opposite directions are identical. Therefore, the operator can lay on the surface comfortably, as seen at marker ❷.

2. Suppression of large one-shot corrections. In Fig. 7(c) and 7(d), there is 15% PD and 50% uniform packet loss. The standard behavior, to always apply corrections in one shot whenever new information is received, is demonstrated by the marker ❸. The operator is immediately forced out of the surface due to a one-shot correction. Since the delay is negligible, oscillations are absent. The amount of information loss is very high. Therefore the size of the corrections can be so large that when canceled out by $S_{\text{corr}}[k]$, the change in adaptive offset does not go unnoticed. An example is shown by the marker ❹. An upside of canceling the correction is that no significant undesired force is generated onto the surface. This means that both the surface and the operator do not experience a sudden spike in measured force. If there would be a TI application with a delicate object, this is undoubtedly an improvement over the standard method. A second example of a correction being suppressed is shown at marker ❺.

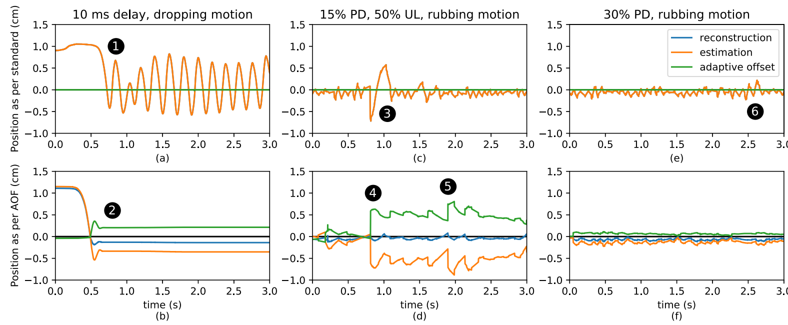
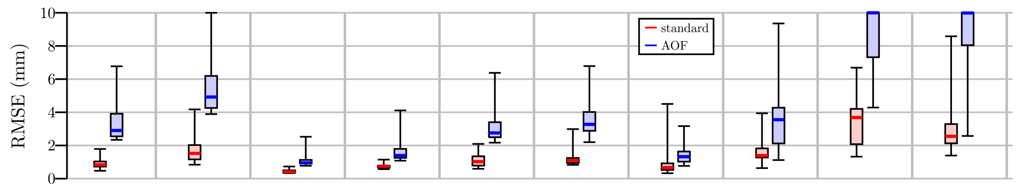


Fig. 7: Demonstration of the working of AOF. The rigid surface is placed at a height of 0 cm. The force feedback is inversely proportional to the penetration depth below 0 cm, and zero otherwise. In (a) and (b) a dropping motion is performed and in (c), (d), (e), and (f) a rubbing motion is performed.



	(a)	(b)	(c)	(d)	(e)	(f)	(g)	(h)	(i)	(j)
delay	5 ms	10 ms	0 ms	0 ms	5 ms	5 ms	0 ms	0 ms	0 ms	5 ms
uniform loss	0 %	0 %	50 %	0 %	50 %	0 %	0 %	0 %	50 %	50 %
bursty loss	0 %	0 %	0 %	50 %	0 %	50 %	0 %	0 %	0 %	0 %
perceptual deadband	0 %	0 %	0 %	0 %	0 %	0 %	15 %	30 %	15 %	15 %

Fig. 8: RMSE is based on logged data for each experiment. The RMSE values are capped at 10 mm, but the last two columns values go higher than the limit. One can see that the proposed AOF consistently scores worse than the usual method of immediate corrections as per the objective measure (RMSE).

3. Suppression of small one-shot corrections. In Fig. 7(e) and 7(f), 30% PD is used. This scenario results in smaller and more consistent corrections than in the previous scenario. In Fig. 7(e), one can see that the estimation appears consistent. However, there is a consistent high-frequency component. This directly results in a noticeable high-frequency component in the measured force and a distinctly recognizable deterioration for the operator. The high-frequency force can push the operator out of the surface completely, as is seen at the marker ❻. In Fig. 7(f), a combination of $S_{\text{delay}}[k]$ and $S_{\text{corr}}[k]$, suppress the high frequency signal and produce a more smooth experience. The reconstruction is consistently just below the surface with a smaller variance than seen in 7(e). Here, AOF helps the user move over the surface more smoothly without clear downsides. Only a marginal amount of adaptive offset is used to accomplish this feat.

B. Objective Analysis

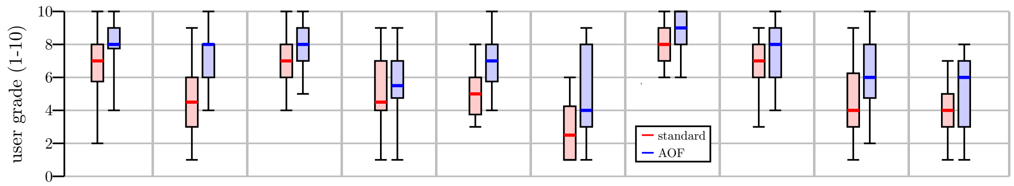
There are multiple objective measures to consider, and among these are multiple traditional network performance parameters like packet loss or transmission delay. However, in comparing AOF and the standard behavior, identical network behavior is used. Therefore network parameters will not provide an insight into the difference between AOF and the standard behavior.

Alternatively, some methods look at the underlying data. Multiple objective measures have been proposed over the years, but none of the proposed methods address the blind spots we highlight in this work. As a representative of these methods, we use RMSE. The data is plotted in Fig. 8. One can see that the reconstructions produced by AOF pose a significant increase in RMSE for every network scenario. Note that for the two rightmost columns, the displayed RMSE is capped at 10 mm, but some of the measured RMSE is well over that value. We use this data to make several observations.

1. AOF creates an overall deterioration in RMSE. Based on RMSE, AOF is outperformed by standard behavior for every scenario tested. This is expected, as AOF actively maintains an adaptive offset, which RMSE will take strong note of. We elaborate further in Sec. II-D.

2. Delay overshadows effects from information loss. Fig. 8(a), 8(e), and 8(f) have the same 5 ms delay, but with no loss, uniform loss and bursty loss, respectively. One can see that the observed RMSE is consistent between these methods. However, it is reasonable to expect that adding uniform and burst loss would deteriorate the system performance.

3. Significant information loss dominates RMSE for AOF. Fig. 8(i) and 8(j) both have a combination of PD and uniform loss. This combination significantly impacts RMSE, especially when AOF is included. Because PD removes most redundancy in the communication, all network loss drops affect packets of significant importance. Because of this, the number of significant corrections is numerous. With the presence of $S_{\text{corr}}[k]$ this causes a significant impact on the adaptive offset



delay	5 ms	10 ms	0 ms	0 ms	5 ms	5 ms	0 ms	0 ms	0 ms	5 ms
uniform loss	0 %	0 %	50 %	0 %	50 %	0 %	0 %	0 %	50 %	50 %
bursty loss	0 %	0 %	0 %	50 %	0 %	50 %	0 %	0 %	0 %	0 %
perceptual deadband	0 %	0 %	0 %	0 %	0 %	0 %	15 %	30 %	15 %	15 %
	(a)	(b)	(c)	(d)	(e)	(f)	(g)	(h)	(i)	(j)

Fig. 9: Subjective user grades showing that the proposed AOF provides a significantly higher user experience.

and thus the RMSE.

C. *Subjective Analysis*

1. AOF creates an across-the-board improvement. As per Fig. 9, AOF improves the user grade significantly for every network scenario compared to the standard method yielding an average of up to three points (on a scale of ten). Note that we expect more improvements by further tuning the shaping and decay functions. A wider variety of tasks and a significantly more extensive data set should give a more accurate view of AOF's improvements and limitations.

2. Significant improvement for delay. In Fig. 9(a) and 9(b), the network is only affected by delay. Because there is no information loss $S_{\text{corr}}[k]$ is inactive, leaving $S_{\text{delay}}[k]$ as the only active shaping function. One can see that the effect of the user grade is significant, where a $10\,\text{ms}$ delay with AOF scores better than $5\,\text{ms}$ delay with the standard method. This is a significant result, as TI demands extremely low delay. This suggests that AOF can potentially relax the stringent delay requirement. However, more research is needed to verify this conclusively.

3. Comparing shaping functions. We consider different scenarios in Fig. 9(c), 9(d), 9(g), 9(h), and 9(i) where no delay is added to the network. In these scenarios, $S_{\text{delay}}[k]$ is marginally active, leaving $S_{\text{corr}}[k]$ as the main shaping function. While for each scenario the inclusion of AOF poses an improvement, the benefits are smaller than any of the scenarios where delay is present. This suggests that, while $S_{\text{corr}}[k]$ is beneficial, $S_{\text{delay}}[k]$ is even more so. This can be partly explained, because the effect of $S_{\text{corr}}[k]$ has the same irregular and one-shot nature, as the correction errors it targets. This also puts pressure on the adaptive offset and the decay functions at play.

4. Uniform versus Bursty loss. Fig. 9(c), 9(d) have uniform loss and bursty loss respectively. The average packet loss is identical, which means that the number of packets dropped is identical between the methods. The only difference is the distribution. A bursty loss distribution causes an average difference in user grade of three points. This significant difference illustrates the disruptive effect of bursty loss on TI applications. The observation is in line with the expectations. A bursty loss model risks longer periods without transmissions, causing the performance to take a large dip at irregular intervals. Uniform loss is fundamentally more consistent and provides a more convincing user experience.

5. Uncovering blind spots of objective results. When considering RMSE as a measure for performance, it would seem that AOF is a deterioration of standard behavior. However, when considering the user grades shown in Fig. 9, AOF provides across-the-board improvement. There are several blind spots at play for this result to happen. First, the concept that a stationary offset is almost imperceivable is not being considered. Instead, the adaptive offset has a massive impact on RMSE. Secondly, the concept that velocity scaling is almost imperceivable is similarly not considered. Causing all efforts by $D_{\text{scaling}}[k]$ to increase RMSE. In both these cases, the blind spots are the lack of consideration for the imperceivability of these errors. Secondly, RMSE does not notice that the high-frequency corrections are mostly nullified. The intentional compensation in velocity, which leads to less unstable force feedback, is not considered either. RMSE does not consider force feedback or any additional information related to the environment. In both these cases, the blind spots are the lack of understanding that specific differences improve the user experience significantly. This is further explained in Sec. II-D. With this, we demonstrate the blind spots present in RMSE and currently available objective measures. Additionally, we demonstrate how we successfully exploit this underutilized potential from the perceivability of errors with AOF to improve the user experience.

VI. Related Works

A. *Standard reconstruction methods*

It is important to maintain a steady refresh rate for TI signals. However, due to network imperfections such as delay, jitter, and packet losses, irregularities in packet arrivals are inevitable despite having a steady transmission at the source. Further, signal compression techniques, such as PD (explained in Sec. IV-A), deliberately drop samples for reducing the data rate. Hence, existing works employ standard reconstruction methods at the receiver. The works in [6], [9], [4] employ zero-order-hold reconstruction, whereas the works in [10], [11], [12] employ linear extrapolation based on the last two received samples. In these works, while extrapolation is done between sample arrivals, the samples are played out with one-shot corrections when received. We showed in Sec. V-C that one-shot correction introduces perceivable impairments.

B. Performance metrics in TI

1) Objective Metrics: These include conventional metrics for network performance such as network delay, jitter, packet loss, and bandwidth. While the work in [10] uses application bandwidth, [4] uses bandwidth and delay, and [3] consider all the delay, packet loss, and bandwidth. The authors in [13] focus on delay and jitter. Another common indicator of teleoperation quality is the position error. The works in [6], [9], [14] use RMSE as the objective metric. Recently, another work [15] proposed using both temporal and position errors for characterizing the quality of TI interaction. We showed blind spots for RMSE (Sec. V-B). However, a similar analysis can be performed for all other objective metrics currently used for TI.

2) Subjective Metrics: Literature provides many works that have conducted a subjective evaluation of TI experiments. Collecting subjective grades by including human subjects in the is the most commonly used metric [5], [16], [9], [15]. However, this approach is often cumbersome and time-consuming. As a workaround, people have come up with several metrics for indicating subjective experience. Some examples of such metrics are Perceptual Mean Square Error (PMSE) [17], Haptic Structure SIMilarity (HSSIM) [18], Haptic Perceptually Weighted Peak Signal to Noise Ratio (HPW-PSNR) [19]. All of these works use variants of RMSE. Hence the inherent problems that exist with RMSE also apply here.

VII. Conclusions

Tactile Internet (TI) presents fresh challenges due to the presence of a human-in-the-loop with haptic feedback in teleoperation. Generally, stringent requirements in terms of latency and reliability are often stated. However, by curating the experience tailor-made for a human operator and exploiting the limited human perception, we can significantly relax the stringent requirements for TI while maintaining a satisfying performance. In this work, we examined how errors can be classified based on their perceivability and their impact on the user experience. We proposed the Adaptive Offset Framework (AOF) to exploit perceivable and imperceivable errors by modifying the adaptive offset to improve the user experience. Subjective experiments confirmed that AOF provides an improvement in user experience in every network configuration. Specifically, we show that AOF significantly improves the user grade, up to 3 points (on a scale of 10) in comparison with the standard reconstruction method. We compared these results with objective analysis and demonstrated multiple blind spots in objective measures that lead to an incorrect characterization of the performance of the TI application. We believe that the concepts explored in this work can provide numerous additional opportunities to improve the user experience, further relaxing the TI system requirements.

VIII. Acknowledgments

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. 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.

[4] 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. [Online]. Available: http://doi.acm.org/10.1145/3063594

[5] P. Hinterseer, E. Steinbach, S. Hirche, and M. Buss, "A novel, psychophysically motivated transmission approach for haptic data streams in telepresence and teleaction systems," in *Proceedings.(ICASSP'05). IEEE International Conference on Acoustics, Speech, and Signal Processing, 2005.*, vol. 2. IEEE, 2005, pp. ii–1097.

[6] A. Bhardwaj, S. Chaudhuri, and O. Dabeer, "Design and analysis of predictive sampling of haptic signals," *ACM Transactions on Applied Perception (TAP)*, vol. 11, no. 4, pp. 1–20, 2014.

[7] M. A. Srinivasan and C. Basdogan, "Haptics in virtual environments: Taxonomy, research status, and challenges," *Computers & Graphics*, vol. 21, no. 4, pp. 393–404, 1997.

[8] A. Bhardwaj, B. Cizmeci, E. Steinbach, Q. Liu, M. Eid, A. E. Saddik, R. Kundu, X. Liu, O. Holland, M. A. Luden, S. Oteafy, and V. Prasad, "A Candidate Hardware and Software Reference Setup for Kinesthetic Codec Standardization," in *2017 IEEE International Symposium on Haptic, Audio and Visual Environments and Games (HAVE)*, 2017, pp. 53–58.

[9] V. Gokhale, J. Nair, and S. Chaudhuri, "Opportunistic adaptive haptic sampling on forward channel in telehaptic communication," in *2016 IEEE Haptics Symposium (HAPTICS)*. IEEE, 2016, pp. 217–222.

[10] P. Hinterseer, S. Hirche, S. Chaudhuri, E. Steinbach, and M. Buss, "Perception-based data reduction and transmission of haptic data in telepresence and teleaction systems," *IEEE Transactions on Signal Processing*, vol. 56, no. 2, pp. 588–597, 2008.

[11] C. Schuwerk, R. Chaudhari, and E. Steinbach, "Perceptually robust traffic control in distributed haptic virtual environments," in *International Conference on Human Haptic Sensing and Touch Enabled Computer Applications*. Springer, 2012, pp. 469–480.

[12] K. M. Yap, T. H. Tee, A. Marshall, K. S. Eu, Y. K. Lee, T.-H. Lee, P. H. Lim, and Y. Chook, "A network-adaptive prediction algorithm for haptic data under network impairments," *IEEE Access*, vol. 9, pp. 52 672–52 683, 2021.

[13] 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. [Online]. Available: http://doi.acm.org/10.1145/3052821

[14] K. Polachan, B. Turkovic, T. Prabhakar, C. Singh, and F. A. Kuipers, "Dynamic network slicing for the tactile internet," in *2020 ACM/IEEE 11th International Conference on Cyber-Physical Systems (ICCPS)*. IEEE, 2020, pp. 129–140.

[15] J. P. Verburg, H. Kroep, V. Gokhale, R. V. Prasad, and V. Rao, "Setting the yardstick: A quantitative metric for effectively measuring tactile internet," in *IEEE INFOCOM 2020-IEEE Conference on Computer Communications*. IEEE, 2020, pp. 1937–1946.

[16] E. Muschter, A. Noll, J. Zhao, R. Hassen, M. Strese, B. Guelecyuez, S.-C. Li, and E. Steinbach, "Perceptual quality assessment of compressed vibrotactile signals through comparative judgment," *IEEE Transactions on Haptics*, 2021.

[17] R. Chaudhari, E. Steinbach, and S. Hirche, "Towards an objective quality evaluation framework for haptic data reduction." IEEE, 2011, pp. 539–544.

[18] R. Hassen and E. Steinbach, "Hssim: An objective haptic quality assessment measure for force-feedback signals," in *2018 Tenth International Conference on Quality of Multimedia Experience (QoMEX)*. IEEE, 2018, pp. 1–6.

[19] N. Sakr, N. Georganas, and J. Zhao, "A perceptual quality metric for haptic signals," in *2007 IEEE International Workshop on Haptic, Audio and Visual Environments and Games*. IEEE, 2007, pp. 27–32.