TIXT: An Extensible Testbed for Tactile Internet Communication

Vineet Gokhale[†], Kees Kroep[†], Vijay Rao[⋆], Joseph Verburg[†], and Ramesh Yechangunja[⋆]

[†]Delft University of Technology, The Netherlands
[⋆]Cognizant Technology Solutions, The Netherlands and India

***Abstract*—The field of teleoperation coupled with force-feedback will undergo a paradigm shift in the forthcoming years with the advent of *Tactile Internet* (TI). Through TI, humankind will enjoy the ability to control and manipulate remote environments in real-time by creating a perception of physical collocation for the human operator. While research and development in TI have seen a surge in the past few years, the overall progress is constrained by two major barriers. Firstly, lack of a TI testbed has made it difficult to establish a performance benchmark. Secondly, asynchronous efforts led by sub-groups belonging to different research disciplines have severely impeded the overall progress of TI. In this work, we take the first step towards addressing these open issues by developing a common testbed for TI applications – Tactile Internet eXtensible Testbed (TIXT). We begin by presenting a classification of the diverse range of TI applications. This helps in making TIXT generic, modular, and extensible. We then present the design principles of TIXT and shed light on its implementation guidelines. Finally, we present the proof-of-concept (PoC) of TIXT through demonstration of two realistic use-cases of TI.**

***Index Terms*—Tactile Internet, TI architecture, TIXT, classification.**

I. Introduction

Hitherto, human interactions with geographically distant environments are limited to visual and auditory modes. While audio and video communications have enabled humans to see and talk to remote entities, they alone are insufficient to reach the next frontier – *teleoperation*. Teleoperation, here, refers to the ability to control and manipulate remote environments in real-time such that physical collocation is perceived for the human operator. The sensation of touch plays a key role in such physical interactions. Therefore, augmenting the *haptic feedback* along with audio and visual communication will provide a sense of truly being present in the remote environment. This requires ultra-low latency (ULL) communication, typically in the order of a few milliseconds. The field of *Tactile Internet* (TI) aims to realize this grand vision of ULL communication [1]. Once deployed, TI will disrupt the fundamental way in which we perceive remote environments.

The TI wave has garnered massive attention recently due to its potential to unlock the gateway to a world of unprecedented applications impacting every aspect of our lives, such as health, education, and entertainment. Such a system enables humankind to benefit from the best of both worlds, wherein a skilled operator can be located in his/her comfort zone and still cater to emergencies in remote places which are otherwise inaccessible, for example, due to logistic or safety constraints. Promising use-cases of TI include telesurgery, remote disaster management including rescue and clean-up, tele-servicing/repair of machinery, cooperative automated driving, logistic robots in smart industries, and collaborative VR gaming. Note that TI also encompasses applications involving teleoperation between machines. Due to its potential to foster the industrial revolution, several industrial giants, such as Nokia, Vodafone, and Ericsson, are investing massively in TI [2]. Early estimates suggest that the market potential of TI could exceed $20 trillion [3].

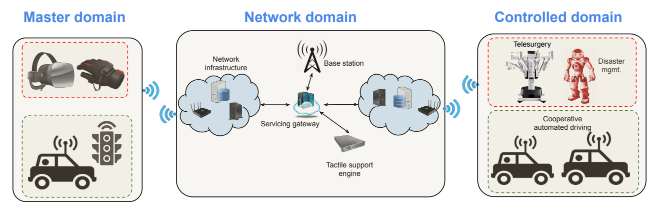

Fig. 1: A high-level representation of TI highlighting the operator, teleoperator, and the various components of the network domain.

A high-level representation of TI is presented in Figure 1. For concreteness in exposition, let us consider the mission-critical TI use-case of telesurgery. The doctor, referred to as *operator*, performs a medical procedure by controlling a remote robotic device, referred to as *teleoperator*, that operates physically on the patient. Haptic-audio-visual signals captured by the teleoperator are transmitted and displayed to the operator via wearable devices. The operator and the teleoperator with the necessary control and communication capabilities are jointly referred to as *master domain* and *controlled domain*, respectively. The *network domain* houses the core network infrastructure for facilitating communication between master and controlled domains.

Interestingly, the promising benefits of TI extend beyond what it has been initially envisioned for. TI's ability to enable teleoperation reduces the need for humans to travel physically to far off places to perform actions. Further, cooperative automated driving, such as convoy of autonomous vehicles, has

A few symbolic icons used in the paper have been reproduced from the following webpages: https://www.intuitive.com, http://www.enggbook.com, https://tracklab.com.au, https://www.softbankrobotics.com.

been proven to substantially cut down on the fuel consumption. Emissions arising from transportation are known to be one of the major contributors to the aggravating problem of *climate crisis* [4]. In this context, TI manifests enormous potential to reduce the global carbon footprint, and thereby take a giant leap towards environmental sustainability.

As the saying goes, *"there is no such thing as a free lunch"*, formidable barriers need to be overcome before we reap the full benefits of TI. Predominant ones include designing techniques for ULL communication (typically sub-$10\,\mathrm{ms}$), accurate sensing and actuation, control loop stability, data compression, among many others. As can be seen, TI is at the crossroads of several independent engineering and scientific disciplines, such as human psycho-physics, mechanical design, communication and control engineering. This has led to widespread research and development activities in both industrial and academic communities towards developing TI solutions. While this is desirable for a speedy realization of TI, two major barriers constrain it.

- Several independent research groups belonging to different organizations are developing TI solutions under heterogeneous experimental settings. For carrying out a reasonable performance comparison between them, it is imperative to bring them to a common ground. This requires a standard implementation of the TI framework that serves as a testbed. However, such a system does not exist in literature. This makes it challenging to create a performance benchmark. Due to the lack of such a testbed, the need and development for new hardware and software techniques, algorithms and modules at every layer of the TI network stack will be hindered. For instance, algorithms for adapting session quality based on network dynamics. Furthermore, when deploying TI, estimating the quality of a session apriori becomes important before deploying mission-critical applications. Again, a testbed is amiss for such requirements.
- It is crucial that the sub-groups belonging to different disciplines work in synergy. However, there currently exists an evident disconnect between them. For instance, mechanical and control researchers simulate network characteristics without employing any real network. This is due to the lack of expertise in setting up a real network. While this could suffice for a preliminary investigation, thorough analyses are meaningful only in the presence of real network behaviour. Similar issues exist in other domains as well.

Such asynchronous research efforts are not desirable for all-round progress of TI. To bridge this large gap and facilitate the large-scale roll-out of TI, there is a severe need for laying out implementation practises to be followed as well as establishing a common ground in the form of a testbed that conforms to a generic TI architecture. In this paper, we take the first step in this direction through the design of *TIXT* - Tactile Internet eXtensible Testbed. Our primary contributions in this work are as follows.

- We present a detailed classification of the broad range of TI applications along with example use-cases for each class. This enables us to understand the important modules of a generic TI application.

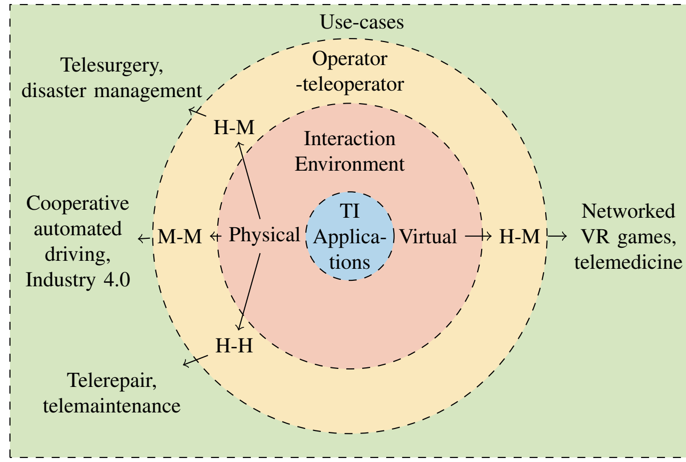

Fig. 2: Classification of TI applications based on the interaction environment and the operator-teleoperator combination. H-M refers to human operator-machine teleoperator. Other classes are abbreviated in a similar fashion.

- Based on this, we provide a generic TI architecture, which is also in line with the architecture being conceived in the IEEE P1918.1 standards working group on TI [5]. We, then proceed to the main proposal of our testbed, TIXT, and its architecture. We also outline the flexibility offered by TIXT and few implementation practices on how to experiment with a specific module in TI.
- We present the PoC of TIXT by demonstrating two typical TI use-cases belonging to different classes.

II. Classification of TI Applications

This section aims to provide a classification of the heterogeneous TI applications that are currently envisioned. This enables us to design TIXT in such a way that it is compatible with all TI applications. We present our two-level classification in Figure 2. While the first level classification is based on the nature of the controlled environment (physical or virtual), the second level takes into account the type of operator-teleoperator combination. Note that each of the operator and the teleoperator can be either a human or machine. We describe each of the TI classes using realistic TI use-cases.

1) Physical TI interaction: This class involves a teleoperation in a *physical* environment. A few use-cases belonging to this class include telesurgery, remote disaster management, and cooperative automated driving. Based on the nature of operator and teleoperator, we divide this class further as follows:

- **Human operator-machine teleoperator (H-M):** As the name suggests, this category of applications is characterized by a human operator manoeuvring a machine teleoperator through haptic feedback. This class is represented as H-M in Figure 2 under the physical interaction class. Let us take telesurgery as an example use-case for this class. The kinematics signal corresponding to the doctor's actions is captured and transmitted over the network domain. The received signal at the controlled domain is used to stimulate the actuators on the robotic device (for example, a servo motor) to replicate the doctor's hand movements. Any

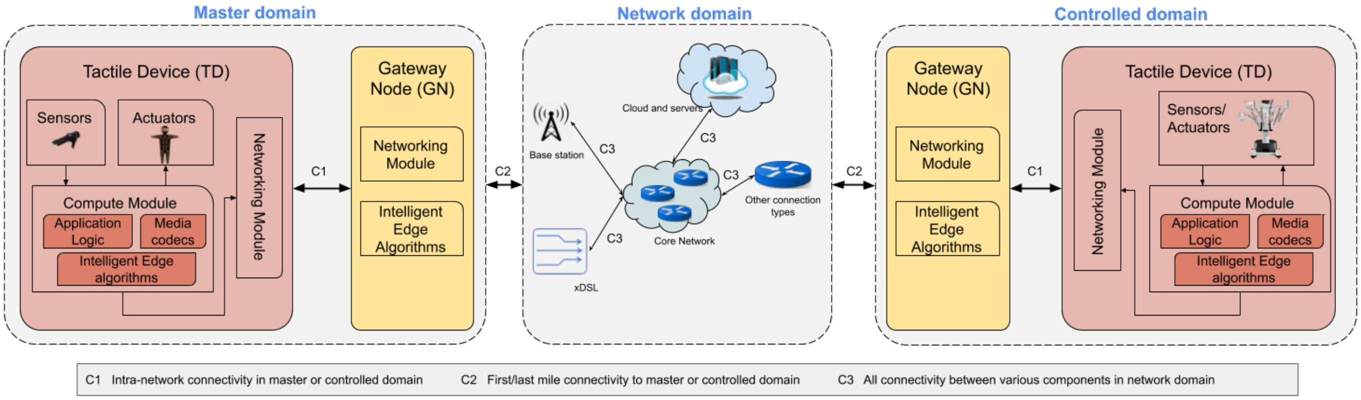

Fig. 3: Block diagram representation of the generic TI architecture that is based on the architecture proposed by IEEE P1918.1 TI standards working group in [5]. This architecture aids us in the design of the proposed generic TI testbed.

changes at the controlled domain, as a result of the actuation or other external factors, are captured by the sensors (for example, pressure sensor, camera, and microphone) and transmitted back to the master domain. This is then displayed to the doctor enabling him/her to make further educated decisions on the task. This class also subsumes applications involving multiple human operators controlling one or more machines to perform a collaborative task. For instance, rescue operations or radioactive cleaning up in a disaster-stricken zone.

- **Machine operator-machine teleoperator (M-M):** This category of TI applications involves a machine operator controlling another machine to performs a physical operation without or with minimal human intervention. This class is represented as M-M in Figure 2. Note that the haptic feedback is unnecessary for this class of applications. Let us take the example of convoy of autonomous trucks that have garnered attention recently due to their fuel-efficiency. Smaller distance between the trucks implies higher fuel savings. This can be potentially achieved with ULL guarantees of TI. The commander truck (master domain) dynamically changes its speed and direction based on the time-varying traffic conditions and the destination. These updates are sensed and communicated to all the follower trucks (controlled domain), which then actuate, for example, the accelerator and the brakes in order to synchronize with the commander. These changes are continuously sensed and transmitted back to the commander, which further guides the followers to the correct trajectory. Other example use-cases include smart factories that make use of logistic robots for fully automating industrial processes and Internet of autonomous drones.
- **Human operator-human teleoperator (H-H):** In this category, a human operator performs a physical, non life-critical operation through a human teleoperator. This class is represented as H-H in Figure 2. An example use-case is tele-repair of machinery, where a skilled human operator located in a call centre (master domain) repairs a broken vehicle by using an unskilled human present in the controlled domain as his/her avatar. The teleoperator could wear an exoskeleton that guides him/her to follow the operator's actions. The operator receives haptic-audio-video feedback from the controlled domain. Such a system not only relaxes the heavy demand for skilled operators that are limited and saves resources due to less/no transits, but also reduces the need for deploying a high-precision robotic device to enable teleoperation.

2) Virtual TI interaction: This category of TI applications involves manipulating remote objects in a *virtual environment* (VE). Since the operation is virtual in nature, the teleoperator is invariably a machine. Further, two machines interacting remotely in VE holds no significance in real-world. Hence, there exists a single category in this class of TI applications.

- **Human operator-machine teleoperator (H-M):** This category involves one or more human operators interacting with a virtual object via force feedback. A typical use-case is immersive, multi-player networked VR gaming. In such a scenario, physical actions of geographically-distributed gamers' (master domain) are sensed locally and sent to a centralized server (controlled domain) running a high-dynamic physics engine. These movements are then mapped to a shared VE, which then determines the visual, audio, and haptic feedback that need to be transmitted back to each gamer. Another use-case is telemedicine, for example, online physiotherapy, where a physiotherapist and a distantly-located patient can collaborate in VE for performing a task involving force feedback activity to help the patient regain muscular and motor skills.

III. A Generic TI Architecture

Based on the different application categories presented in the previous section, we can identify the following core components in any TI application:

- *Sensor*: This component is present in both master and controlled domains, wherein it captures the tele-manipulation instructions in the master domain (e.g., kinesthetic inputs or acceleration inputs) and the changes due to the results of tele-manipulation in the controlled domain (e.g., haptic feedback).

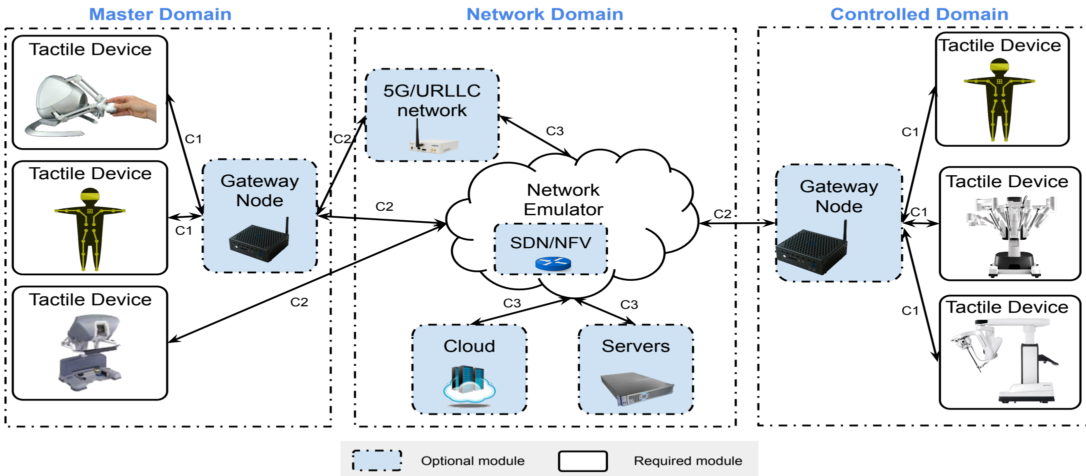

Fig. 4: The modular architecture of TIXT.

- *Actuator*: This component is again present in both master and controlled domains. In the controlled domain, these components execute the operator's instructions. In the master domain, an actuator is used to render haptic feedback or may be a display for visualization.
- *Compute*: This component provides the computation and control logic; it assimilates the sensor/actuator data and generates the control commands intended to accomplish the application goals in the master and the controlled domain.
- *Network*: This module bridges the master and the controlled domains and is responsible for ULL communication between the two domains.

With these necessary components, we present a generic TI architecture (see Figure 3) that can cater to the categories of applications presented in the previous section. This architecture is a simplistic version of the architecture proposed by IEEE P1918.1 TI standards working group that has been published in [5] that aids us in designing a generic testbed.

The master and controlled domains could consist of one or more *Tactile Devices* (TD), which refers to devices with sensing and actuation modules along with intelligent control and networking capabilities. Note that sensing and actuation deals not only with haptics, but also with audio and video signals. Each TD could be a stationary or mobile device. The compute module of TD can further be composed of the following elements.

- **Application logic** that houses the 'controller' functionality in the master domain along with the responsibility of interfacing with peripherals, multiplexing multimedia signals and communicating these with the controlled TDs. In the controlled domain, the application logic module is the same without the controller functionality. For its functions, it coordinates with the media codecs and intelligent edge algorithms modules to deliver a seamless TI application.
- **Media codecs** that digitize and provide encoding/decoding functionalities for haptic (both kinesthetic and tactile), audio, and video signals efficiently.
- **Intelligent edge algorithms** that monitor and adapt the application parameters to meet the required quality of experience (QoE) despite any network delays and losses.

TD also encompasses a networking module that enables connectivity (C1 in Figure 3) to a gateway node (GN). The connectivity could be through any wired/wireless/optical networking technology.

A *Gateway Node* (GN) is responsible for enabling the first/last mile connectivity to TDs. The networking module in GN switches packets between the network domain and one or more TDs that are connected to the GN. As networks can offer time-varying performance, the intelligent edge algorithms in a GN can choose the best network(s) or adapt networking parameters to maintain the quality of a TI session.

The *network domain* encompasses 5G networks and other ULL technologies that provide connectivity to the edge devices along with core network functionalities to route packets. The core network may employ software-defined networking (SDN) technologies to provide a higher quality of service (QoS) to TI applications. Furthermore, the network domain provides connectivity to cloud computing facilities, in case the TDs want to leverage, and servers for TI session establishment and tear-down.

IV. TIXT: TACTILE INTERNET EXTENSIBLE TESTBED

In this section, we present TIXT – Tactile Internet eXtensible Testbed – our vision towards building a generic, modular, and extensible TI testbed platform encompassing several standard TI functionalities that can be readily utilized off-the-shelf. This facilitates easy development and testing of a broad range of TI applications.

A. Architecture

TIXT's architecture, as shown in Figure 4, follows the lines of our generic architecture presented in Figure 3. Apart from TDs and GNs, there are a few more modules that are explicitly depicted, such as base station, SDN/NFV, cloud, and server modules. Below we explain the different modules in the architecture.

In the master domain, TD is a real device such as a Novint Falcon/haptic gloves for H-H/H-M applications or a software controller for M-M applications. In the controlled domain, a TD can be a surgical robot or VE. As we intend to develop a real-world testbed, this module must be implemented as it would be in a real TI application.

As indicated in the figure, GNs are optional as a TD may directly connect to the network domain, for instance, through 5G networks, Ethernet, or optical fibers. If 5G networks are to be experimented with, then a 5G base station module can be deployed. The packets from a TD are routed through this module to the network domain.

For the core network, we propose to employ a network emulator. We choose a network emulator over a simulator, as in [6], since we are interested in mimicking the behavior of a live network, while a simulator utilizes mathematical models of traffic, network models, channels and protocols and hence not close to a real-world testbed. This module can be configured with parameters to emulate the loss, delay, and jitter characteristics as in a real network; for instance, based on a measurement campaign on a real URLLC network and then employing these characteristics in this module. Due to this, the scope of this module can be varied depending on the experiments. In one case, the core network could be configured to emulate the end-to-end network characteristics i.e., encompassing all network characteristics between GNs in the master and the controlled domains. In another case, this module may be shrunk to represent a tiny part of the core network.

B. Implementation Guidelines

In this section, we explain the interfaces and provide important implementation guidelines to setup suitable experiments.

Typical tactile sensors and actuators can be interfaced to micro-controllers or a PC/laptop. While the micro-controllers can implement, at the very least, the basic versions of the necessary sub-modules, PCs can implement sophisticated algorithms and codecs. The micro-controllers can also be interfaced with a PC, if necessary.

GNs, as mentioned earlier, is an optional module. This module can be implemented on a hardware, such as PC, single-board computers and the likes. The connectivity between TDs and GNs, C1, can be wired, wireless, or any other type of communication technology.

The interface, C2, can be implemented through a network interface hardware or a simulator, such as NS3. NS3 has the ability to send and receive packets from a real network interface. We envision that the network emulation module transacts with standard Internet Protocol (IPv4/IPv6) packets. That is, the C2 interfaces terminating at the emulator and C3 use IP packets to communicate with other modules.

Since all the communication interfaces are based on standard technologies, TIXT becomes inherently extensible. For instance, consider that one would like to simulate 5G networks in TIXT. In such a case, the C2 interface between GN and the base station must employ the necessary 3GPP standards. Such an interface can be built in simulators such as NS3.

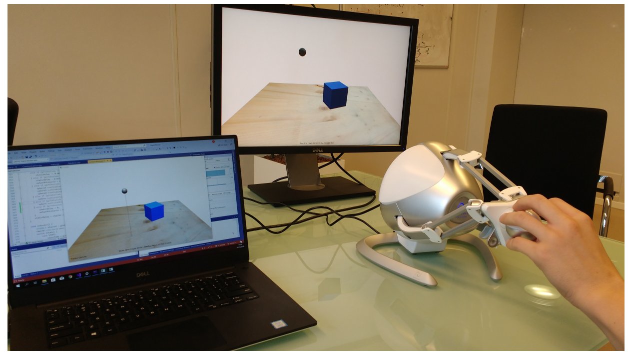

Fig. 5: Demonstration of TI use-case of H-M interaction in VE using TIXT. The VE is rendered on a remotely located computer (laptop in the figure) and the human operator looks at the desktop monitor while operating the haptic device.

A 5G topology would be created in NS3, which includes a device representing the GN (simulated GN) with an IP network interface. The GN will send all its packets to the simulated GN, and then a 5G communication network can be looped into the testbed. The 5G network gateways can send out IP packets to the network emulator interface.

Consider another case wherein a real base station has to be connected to TIXT. As long as the device outputs IP packets to the emulator, the base station can be interfaced. For emulating URLLC in 5G networks, we envision using a software-defined radio with the OpenAirInterface 5G implementation.

Similarly, in the core network, experiments on SDN can be done by either connecting a real SDN enabled switch or through a simulator for SDN enabled switch and talking IP packets to the emulator. Another case in point is interfacing external cloud servers or session management servers easily.

Therefore, TIXT is modular and extensible as it is loosely coupled using standard communication technology interfaces. Any TI scenario with a combination of simulators and real hardware can easily be incorporated in the same experiment. This makes TIXT also generic while allowing for a complex TI scenario to be evaluated reliably in an almost live network on a table.

C. Proof-of-Concept (PoC)

In this section, we present the PoC of TIXT through demonstration of two realistic use-cases belonging to different classes: (i) H-M in VE, and (ii) H-M in physical environment. We explain the implementation details of the different elements of TIXT pertaining to these two applications. It is important to remark that the other classes of applications can be seen as elementary extensions of the ones presented here, thanks to the modular and extensible design of TIXT.

1) H-M in VE: For this class, we designed a simulated environment with haptic feedback enabled by a physics engine shown in Figure 5.

- **TD:** The human operator interacts with a remotely rendered VE application using Novint Falcon haptic device [7]. The physics in VE is simulated using Chai3D – a cross-platform, C++-based haptic framework [8]. The VE comprises of

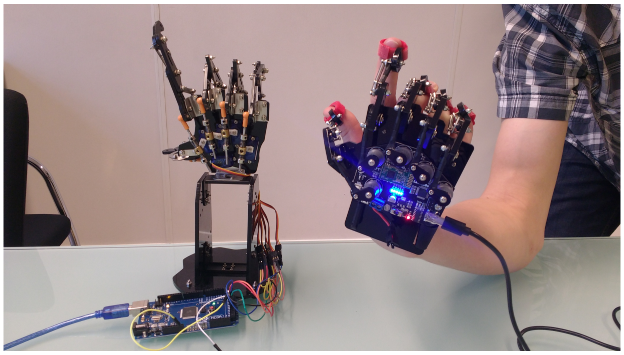

Fig. 6: Demonstration of TI use-case of H-M interaction in physical environment using TIXT where the human operator wears an exoskeleton and controls the remote robotic arm.

a solid cube placed on a rigid surface. The task for the operator is to move the cube towards a pre-determined target location in the VE. The desktop workstation (only monitor shown in the figure) along with the haptic device forms the TD in the master domain, whereas the laptop running the VE forms the TD in the controlled domain.

As a haptic (kinesthetic) codec, we implement perceptual deadband based on Weber's law of perception [9]. To reduce the high bandwidth requirement of video feedback, we take the following design approach: The master TD locally renders a copy of the VE which is communicated by the controlled TD at the beginning of the session. Only the updates to the VE as a result of the interaction are communicated at the standard haptic sampling rate of 1kHz. The master TD then applies these updates to the VE to generate the haptic and visual feedback locally. To achieve this optimal TI communication, we use the code base provided by Bhardwaj et al. [10] in which the authors implement haptic interaction on a non-networked setup.

- **Network domain:** The real network characteristics are generated using Netem, a standard network emulator, installed on a dedicated workstation running Ubuntu 18.04 OS. A GUI allows the user or network administrator to systematically configure network parameters such as latency, jitter, loss, and buffer size. For packet loss, we choose Gilbert-Elliott model since it is known to capture the losses on Internet at a reasonable level.

2) H-M in physical environment: We developed an application where the operator wears an exoskeleton device to grasp a remote, physical object through a robotic arm, as shown in Figure 6.

- **TD:** The exoskeleton and the arm, both manufactured by Lobot Robot [11], are embedded with kinematic sensors and servo motors, respectively. The arm has 15 degrees of freedom – 3 DoFs for each of the 5 fingers of the hand. We employed an Arduino Mega 2560 board in conjunction with the arm since the arm has no on-board computational unit. The kinematics signal generated due to the hand movements of the operator are transmitted to the robotic arm. The exoskeleton and the arm, along with their corresponding workstations, form the TDs in the master and the controlled domains, respectively. For the sake of simplicity, we did not mount any force sensors on the arm. We employed a commodity webcam in the controlled domain and displayed the the visual scene in the controlled domain to the operator using a computer monitor. Therefore the human operator utilizes only visual feedback for this task. However, it is worth remarking that TIXT encompasses necessary software implementation such that by merely embedding force sensors on the arm, the operator can be supplied with force feedback.
- **Network domain:** The implementation of the network domain is the same as in case of H-M interaction in VE setting that we described earlier in Section IV-C1.

In both of the above use-cases, several human subjects were invited to interact with the remote environments. We were able to conduct repeatable and reliable experiments across the user set, thereby substantiating the validity of TIXT. We also employed TIXT for conducting TI experiments, the results of which are presented in our recent work [12].

V. Lessons Learned

In the course of design and implementation of TIXT, we learned several important lessons for ensuring compliance to stringent requirements of TI applications. We outline them in this section.

- **Characterisation and optimisation:** Accurate characterization of each and every sub-module employed in the testbed, such as haptic/audio/video interfaces, wires/wireless communication technologies, network emulators/simulators and OS, is crucial for performance evaluation. For instance, we observed that while Linux OS guarantees haptic sampling rate of 1 kHz, the sampling rates on Windows show vigorous fluctuations between 300 Hz and 1 kHz. Further, configuration of the devices for optimal performance is necessary to avoid any undesired latency in the TI application. For example, haptic frames need to be processed invariably at 1 kHz, whereas video frames require only 30-60 Hz. We found that high priority sub-tasks should be executed by spawning separate threads, while multiple low priority ones could be grouped together depending on their frequency of execution.
- **Modelling:** When employing network emulators or simulators, models that closely represent the real-world networks should be employed. This ensures that the testbed is useful in characterising and generalising the performance of TI algorithms and other sub-modules.
- **Clock synchronization:** Intelligent orchestration between network domain components at fine-grained timescales is crucial for TI. This demands clock synchronization with an accuracy of sub-microseconds or higher. Several protocols are available for this purpose, such as Network Time Protocol (NTP), Precision Time Protocol (PTP). While clock synchronization is relatively straightforward on local or small networks, thorough investigation is necessary before choosing the synchronization protocol for large networks.

VI. Related Literature

Several testbed proposals for URLLC applications have emerged in recent times; see, for example, [13], [14], [15]. The work in [13] presents a testbed for autonomous driving with 5G network. This testbed is extremely tuned to a specific use-case which significantly limits its applicability for TI applications. The work in [14] presents a testbed for Cyber-Physical System (CPS). A generic CPS application differs widely from TI application, for example, in terms of the existence of tight control loop and ULL requirements, thereby making this testbed infeasible for TI. The work in [15] specifically proposes a 5G testbed framework for the physical layer. The authors in [6] present a framework designed specifically for TI. However, this framework is tailor-made only for physical remote environments. The authors do not discuss the extensibility with other classes of TI applications. Lack of generic testbeds motivated us to design a generic, modular, and extensible testbed for TI applications.

VII. Conclusions

Tactile Internet (TI) has gained significant research emphasis in the recent past due to its potential for providing perceived collocation of remote environments. While remarkable progress is being achieved in the direction of building TI sub-systems, two major barriers exist: (i) lack of common ground for establishing performance benchmark, and (ii) asynchronous research efforts across different TI disciplines. These are severely impeding the overall progress of TI. In this paper, we took the first step towards addressing these issues through the design of TIXT – a generic, modular, and extensible testbed for TI applications. First, we carried out a classification of the broad range of TI applications. Using a generic TI architecture, which aligns with the architecture proposed under P1918.1, we presented the design and implementation guidelines of TIXT. We demonstrate the PoC and general applicability of TIXT through two realistic TI use-cases belonging to different TI application classes.

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] "Tacnet 4.0," http://www.tacnet40.de/index_englisch.html, accessed: 2019-10-19.

[3] K. Moskvitch, "Tactile internet: 5g and the cloud on steroids," *Engineering & Technology*, vol. 10, no. 4, pp. 48–53, 2015.

[4] "Who: Health and sustainable development," https://www.who.int/sustainable-development/transport/en/, accessed: 2020-01-31.

[5] 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.

[6] K. Polachan, T. Prabhakar, C. Singh, and F. A. Kuipers, "Towards an open testbed for tactile cyber physical systems," in *International Conference on Communication Systems & Networks (COMSNETS)*. IEEE, 2019, pp. 375–382.

[7] "Haptics house," https://hapticshouse.com, accessed: 2019-10-19.

[8] F. Conti, F. Barbagli, D. Morris, and C. Sewell, "Chai 3d: An open-source library for the rapid development of haptic scenes," in *World Haptics*. IEEE, 2005, pp. 21–29.

[9] 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.

[10] A. Bhardwaj, B. Cizmeci, E. Steinbach, Q. Liu, M. Eid, J. AraUjo, A. El Saddik, R. Kundu, X. Liu, O. Holland *et al.*, "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)*. IEEE, 2017, pp. 1–6.

[11] "Lobot robot," https://www.lobot-robot.com, accessed: 2019-10-19.

[12] J. 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 - International Conference on Computer Communications*. IEEE, 2020.

[13] S. Pandi, F. H. Fitzek, C. Lehmann, D. Nophut, D. Kiss, V. Kovacs, A. Nagy, G. Csorvasi, M. Tóth, T. Rajacsis *et al.*, "Joint design of communication and control for connected cars in 5g communication systems," in *2016 IEEE Globecom Workshops (GC Wkshps)*. IEEE, 2016, pp. 1–7.

[14] M. Szczodrak, Y. Yang, D. Cavalcanti, and L. P. Carloni, "An open framework to deploy heterogeneous wireless testbeds for cyber-physical systems," in *2013 8th IEEE International Symposium on Industrial Embedded Systems (SIES)*. IEEE, 2013, pp. 215–224.

[15] H. Cao, S. Gangakhedkar, A. R. Ali, M. Gharba, and J. Eichinger, "A testbed for experimenting 5g-v2x requiring ultra reliability and low-latency," in *WSA 2017; 21th International ITG Workshop on Smart Antennas*. VDE, 2017, pp. 1–4.

Biographies

Vineet Gokhale (V.Gokhale@tudelft.nl) received a Ph.D degree in 2017 from Indian Institute of Technology Bombay, India. During 2017-2019, he was an Assistant Professor at University of South Bohemia, Czech Republic. He is currently a post-doctoral researcher in the Embedded and Networked Systems group at Delft University of Technology, The Netherlands. His research interests include tactile internet, haptics technology, and MAC protocols for wireless communication. He is a contributing member of IEEE standardization committee for Tactile Internet P1918.1.

Kees Kroep (H.J.C.Kroep@tudelft.nl) received an MSc degree in electrical engineering at Delft University of Technology in 2018. Currently he is working toward his Ph.D. degree in Embedded and Networked Systems group at the same faculty. His Ph.D. work focuses on edge computing for tactile internet. His research interests include tactile internet and haptics technology.

Vijay S. Rao (Vijay.Rao@Cognizant.com), part of the Cognizant IoT Center of Excellence, is currently focused on developing new technical capabilities and offerings in the IoT domain. Vijay is leading the R&D programs in the Netherlands in the areas of AI/ML at the edge, swarm robotics, 5G and tactile internet. Prior to Cognizant, he was a post-doctoral fellow in the ENSys group in TUDelft, during which he also contributed as the lead software developer to the TUDelft's lunar rover mission. He received his Ph.D. and MSc (*cum laude*) from Delft University of Technology, The Netherlands. He has several top peer-reviewed publications and holds two patents. He is a contributor to the IEEE standards on tactile internet (P1918.1) and nano-communications (P1906.1).

Joseph Verburg (J.P.Verburg@student.tudelft.nl) received his BSc (2017) and MSc *cum laude* (2019) in electrical engineering from Delft University of Technology. His MSc thesis focused on developing a quantitative metric for measuring Tactile Internet. He is currently working as a Software Engineer in KPN, The Netherlands. His research interests include Tactile Internet, LoRa, and NB-IoT.

Ramesh Yechangunja (Ramesh.Yechangunja@cognizant.com), Director, Cognizant IoT Center of Excellence, is currently focused on building market facing offerings in the area of IoT which is one of the key Digital Imperatives at Cognizant. He is also leading key initiatives in building cutting edge technology accelerators in the areas of 5G and IoT Edge. Ramesh is also driving technology R&D programs globally. In this role, he is instrumental in establishing the relationships and research programs with some of the leading universities across the world. Currently oversees research in the areas of Autonomous Robots, Connected Vehicles, Digital Twin for Humans, Immersive Communication, Tactile Internet, and Extended Reality. He has a Masters in EE from Penn State University.

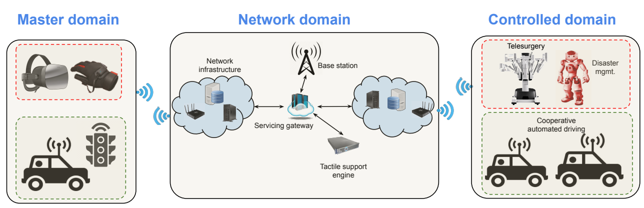

Fig. 1: A high-level representation of TI highlighting the operator, teleoperator, and the various components of the network domain.

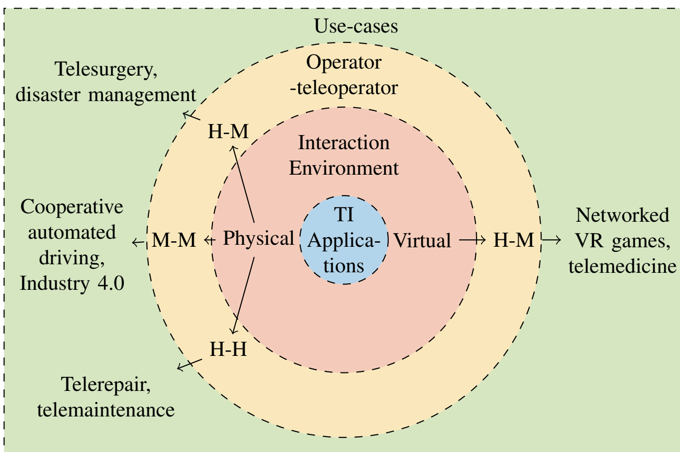

Fig. 7: Classification of TI applications based on the interaction environment and the operator-teleoperator combination. H-M refers to human operator-machine teleoperator. Other classes are abbreviated in a similar fashion.

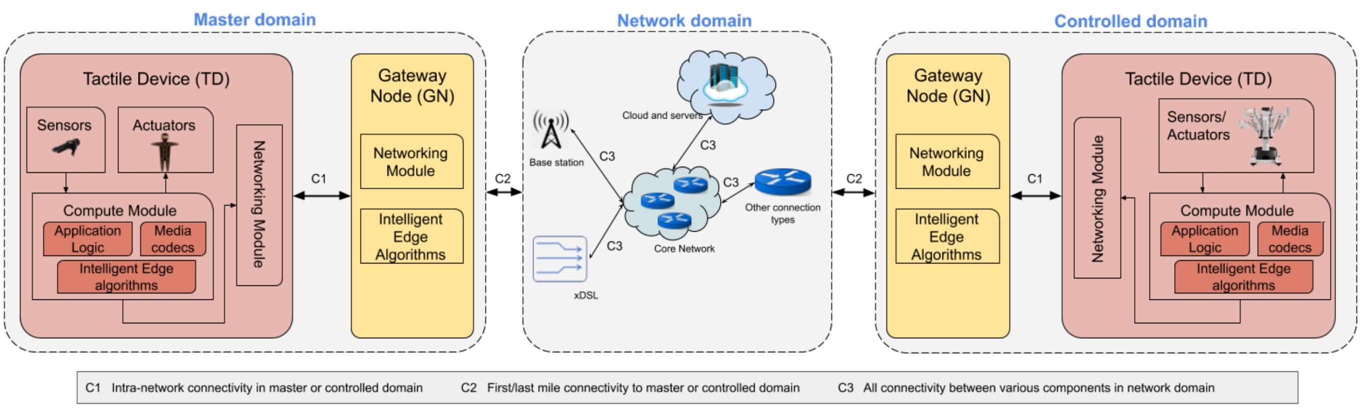

Fig. 3: Block diagram representation of the generic TI architecture that is based on the architecture proposed by IEEE P1918.1 TI standards working group in [5]. This architecture aids us in the design of the proposed generic TI testbed.

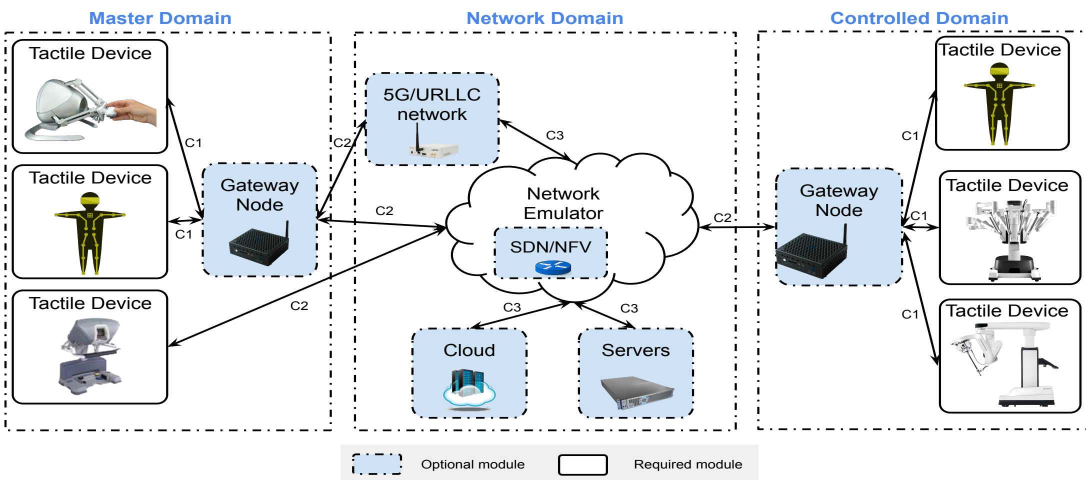

Fig. 4: The modular architecture of TIXT.

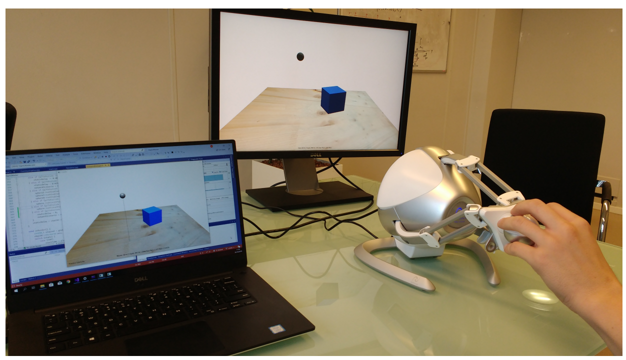

Fig. 5: Demonstration of TI use-case of H-M interaction in VE using TIXT. The VE is rendered on a remotely located computer (laptop in the figure) and the human operator looks at the desktop monitor while operating the haptic device.

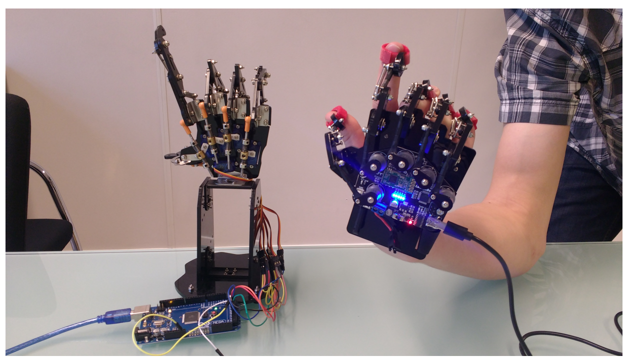

Fig. 6: Demonstration of TI use-case of H-M interaction in physical environment using TIXT where the human operator wears an exoskeleton and controls the remote robotic arm.