Deterministic Networking Empowered Robotic Teleoperation

Chao Yang, Hao Yu, Qize Guo, Tarik Taleb, Jose Costa Requena, and Kari Tammi

***Abstract*—Robotic teleoperation has seen widespread adoption across industries. Advances in technologies like the Internet of Things (IoT), Artificial Intelligence (AI), and Extended Reality (XR) are making teleoperation more flexible. The integration of visual, audio, tactile, and immersive interfaces enhances situational awareness, enabling teleoperators to interact effectively with complex remote environments and make informed decisions. However, challenges persist, particularly in environments characterized by constrained network resources. The limited bandwidth, delays, and intermittent communication can disrupt the teleoperator's interaction with the robot. This study aims to comprehensively understand the scenarios of industrial ground robotic teleoperation and the intricacies of its network to effectively enhance teleoperation performance. Initially, we introduce the multimodal perception-enhanced robotic teleoperation, accompanied by an analysis of the Quality of Service (QoS) of all involved streams. Subsequently, we introduce a Deterministic Robotic Teleoperation (Det-RT) system, along with a deterministic traffic flow scheduling framework designed for real-time remote environment perception. Finally, we evaluate the proposed scheduling solution in a simulated environment to assess its performance within the Det-RT system. The results obtained demonstrate the capability of our solution to deliver high-quality teleoperation performance.**

I. Introduction

Robotic teleoperation is crucial in industrial settings, such as logistics, healthcare, manufacturing, oil and gas, and nuclear power plants, allowing operators to control robots remotely for tasks that are within hazardous or inaccessible environments. By utilizing robotic teleoperation, industries can improve safety by reducing human exposure to dangerous conditions. Additionally, teleoperation enables the execution of tasks that require high flexibility and human supervision, extending beyond the capabilities of automated systems alone. In the robotic teleoperation system, a teleoperator directly controls and supervises a robot over a designated communication channel, collects information concerning the remote environment, issues commands related to task completion as a master, and the robot executes the task based on the control and feedback from the teleoperator. For teleoperators, understanding the robot's conditions and its surroundings helps them understand remote environments and achieve better performance in teleoperation.

The integration of multimodal perception has notably enhanced situational awareness, encompassing visual, tactile, and environmental awareness in robotics teleoperation, reshaping operators' interactions with remote robots. The major operational paradigm relies on line-of-sight control with visual contact or video feedback [1]. However, this approach frequently leads to coordination discrepancies between teleoperators and robots, resulting in disorientation. Integration of depth sensors addresses this challenge by furnishing operators with precise spatial data, thereby facilitating a comprehensive understanding of the environment's layout and potential obstacles. Haptic feedback systems further enrich the operator's understanding of the robot's interactions by providing tactile information through vibrotactile feedback of the robot's contact vibrations. Additionally, Extended Reality (XR) presents immersive environments to enable operators to perceive the robot's status and surroundings in real-time. Moreover, Artificial Intelligence (AI) enhances situational awareness by analyzing sensor data, detecting objects, predicting potential obstacles, and autonomously adjusting the robot's behavior accordingly. These advancements collectively empower operators with heightened perception and control in robotics teleoperation scenarios.

However, it is essential to consider that with the influx of new devices and services aimed at achieving a higher level of situational awareness, there is a corresponding increase in the frequency of information flow, resulting in dynamic and heightened network traffic and subsequent network load. For instance, Hu et al. [1] elucidated the challenges and potential solutions associated with XR-assisted teleoperation, including uplink capability, 3D coverage, mobility with multi-access edge computing, and the rapidly changing environment. Furthermore, traditional networks typically rely on fixed configurations, limited bandwidth allocations, and a lack of adaptive mechanisms to dynamically adapt to changing traffic patterns. These constraints can hinder the dynamic allocation of resources and the prioritization of critical data streams. As a result, in the context of robotic teleoperation with its diverse and demanding workloads, traditional industrial networks may encounter challenges in maintaining consistent performance levels and adhering to stringent Quality of Service (QoS) guarantees. Thus, there is a need for more agile and adaptive network architectures capable of accommodating the evolving demands of robotic teleoperation while ensuring robust performance and QoS guarantees.

To address these challenges, one potential solution involves the integration of Deterministic Networking (DetNet) technologies into robotic teleoperation systems. DetNet technologies can enhance traffic transmission between the robot and teleoperators by incorporating the traffic scheduling function into the networks, particularly when a high traffic load happens in the networks. A comprehensive deterministic network

Chao Yang, Jose Costa Requena, and Kari Tammi are with Aalto Univeristy, Espoo 02150, Finland. Hao Yu and Qize Guo are with ICTFicial Oy, Espoo 02130, Finland. Tarik Taleb is with Ruhr University Bochum, Bochum 44801, Germany. Hao Yu (hao.yu@ictficial.com) is the corresponding author.

framework including traffic analysis and time-critical flow scheduling functions is required for robotic teleoperation to facilitate real-time information exchange between robots and teleoperators.

The paper discusses the essential aspects of deploying multi-model perception teleoperation for industrial ground robots, with a subsequent analysis of network challenges. Furthermore, we explore the utilization of DetNet technologies to improve the QoS of robotic teleoperation services. To offer a comprehensive solution, we present a Deterministic Robotic Teleoperation (Det-RT) system where the control plane, including the traffic analysis engine and flow scheduling engine, is highlighted. Additionally, compared with the traditional flow scheduling method, a conflict graph-based group flow scheduling solution is proposed in this paper for time-critical traffic, ensuring deterministic QoS performance based on this framework.

II. Enabling Multimodal Perception-Enhanced Robotic Teleoperation

Robotic teleoperation user interfaces amalgamate multi-modal perception from various sensors and devices, offering users a comprehensive view of remote environments. Multi-modal perception in robotic teleoperation encompasses visual feedback such as video streams, images, textual data, and immersive 3D experiences, along with non-visual feedback, including audio, force, and tactile senses. Leveraging these modalities enhances operators' interaction with remote spaces, improving situational awareness and control accuracy. Fig. 1 shows the multimodal perception-enhanced robotic teleoperation scenario. The paper further classifies multimodal perception into five main categories, including visual perception, audio perception, haptic feedback, Augmented Reality (AR) interface, and Virtual Reality (VR) environment.

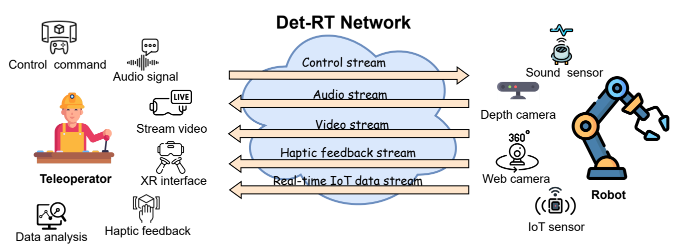

Fig. 1. Multimodal perception-enhanced robotic teleoperation over Det-RT network.

A. Multimodal Perception

Visual cues provide spatial context, object recognition, and depth perception, crucial for navigating complex environments and manipulating objects accurately. The most prevalent approach to robotic teleoperation involves utilizing video feeds as the primary means of control and feedback. For example, a common practice involves the deployment of onboard cameras to afford operators a first-person perspective of the remote environment. Moreover, the utilization of 360-degree cameras presents panoramic views, thereby enhancing the immersive nature of teleoperation experiences. However, the integration of these advanced camera systems results in heightened communication channel bandwidth demands, presenting a challenge for seamless data transmission in teleoperation setups.

Auditory information acts as a supplementary component alongside visual feedback. Real-time auditory feedback offers insights into system status or alerts, enhancing operator responsiveness and system efficiency. Additionally, integrating 3D sound in an immersive environment enables teleoperators to localize the origin of sound, enhancing their situational awareness and facilitating more precise interactions within the virtual environment.

Haptic feedback refers to the tactile sensation, the force, or vibration provided to the operator, allowing them to perceive and interact with the remote environment through physical sensation. Advanced haptic systems utilize force sensors, accelerometers, and actuators to simulate realistic interactions, providing a sense of touch. Network time delay significantly impacts haptic feedback by introducing discrepancies between the operator's actions and the corresponding tactile sensations from the remote environment.

AR interface in robotic teleoperation revolutionizes operator interaction by overlaying virtual information onto the physical environment. Typically, such information is stored in a private edge server to accommodate local computational constraints and address privacy concerns. Moreover, AR interfaces facilitate real-time visualizations of robotic operations, thereby augmenting situational awareness and improving task execution. These interfaces also feature intuitive control mechanisms, empowering operators to manipulate robots using natural gestures or voice commands.

VR environment immerse operators in simulated settings, enhancing their spatial awareness and interaction with remote robots. These environments seamlessly integrate visual, audio, and haptic cues to replicate real-world scenarios, allowing operators to visualize and manipulate virtual representations of robotic systems. Incorporating 3D scene reconstruction further enriches VR environments, enabling operators to perceive remote environments in realistic depth. VR environments impose elevated requirements on both network quality and computing capabilities.

B. QoS Analysis

QoS is crucial for managing traffic and ensuring the performance of critical applications within a network. In the context of robotic teleoperation, where multimodal perceptions are transmitted through a communication channel, network performance directly impacts teleoperator performance. The network performance is mainly determined by latency, jitter, packet loss rate, and bandwidth. For instance, a latency of 500 ms leads to a notable decrease in task completion time and an increase in the perceived physical workload [2]. Furthermore, a jitter exceeding 82 ms significantly degrades control precision, especially for tasks like precise placement that demand high levels of visual feedback Consequently, managing network parameters is vital for optimizing the performance of robotic teleoperation systems.

TABLE I
COMMUNICATION QOS REQUIREMENTS IN ROBOTIC TELEOPERATION.

Types	Latency	Jitter	Packet Loss Rate	Bandwidth	Ref
Control Signal	< 1ms	$< 1\mu s$	$< 10^{-9}$	> 1.5Mbps	[1], [3]
Audio Stream	< 100ms	< 50ms	$< 10^{-5}$	$22 - 200$Kbps	[4]
Video Stream	< 100ms	< 20ms	$< 10^{-3}$	> 40Mbps	[5], [6]
VR Stream	< 20ms	< 15ms	$< 10^{-5}$	> 100Mbps	[7], [8]
Haptic Feedback	$3 - 60$ms	$1 - 10$ms	$10^{-4} - 10^{-1}$	> 128Kbps	[4]

The QoS parameters for streams of multimodal perception are listed in Table I. For example, in the context of factory automation within a smart factory environment, it is imperative to ensure end-to-end latency of no more than 1 ms with a reliability level of $1 - 10^{-9}$. In the upstream direction, a 6-Degree-of-Freedom (6-DoF) robotic system transmits status messages at intervals of 2 ms, resulting in a generated upstream status traffic of 1.5 $Mbps$. Regarding VR stream, an acceptable latency threshold is approximately 20 ms, while an ideal experience necessitates latency to be below 8 ms. While the QoS requirements for teleoperation scenarios exhibit similarities between AR and VR streams, the specific values may vary based on the distinctive attributes and goals of each technology, as well as the nature of the teleoperation tasks involved.

III. ENABLING DETERMINISTIC ROBOTIC TELEOPERATION

In a reliable robotic teleoperation environment, the information exchange between the teleoperator and robot exposes strict requirements over the physical networks. The physical networks are supposed to provide robust connections with deterministic QoS, where the latency, jitter, and packet loss of the traffic flows should be bounded within certain values. In this section, a Deterministic Robot Teleoperation (Det-RT) system is illustrated where deterministic networking technologies are leveraged to support deterministic QoS for reliable robotic teleoperation.

A. Deterministic Networking Technologies

Ethernet technology has garnered great popularity as a widely used form of networking communication due to its basic connecting mechanisms and protocol. With the introduction of best-effort Ethernet services, the goal is to simplify protocol operations and minimize the complexity of the network, leading to lower costs for the operation and maintenance of the network. It is inherent to the Ethernet protocol that it does not provide an End-to-End (E2E) deterministic quality of service guarantee for data flows, although it has experienced major successes and extensive implementation. Through the implementation of a collection of established processes and principles, the IEEE 802.1 Time-Sensitive Networking Task Group (TSN TG) standards improve upon the ordinary Ethernet data link layer. A few examples of these include time-aware traffic shaping and frame preemption, both of which guarantee the delivery of time-sensitive flows with bounded latency, bounded delay fluctuations (jitter), and extraordinarily low packet loss. Having these characteristics is necessary for the successful operation of applications in the automotive and industrial control industries [9].

Layer-2 networking can be considered the TSN TG's core area of concentration. On the other hand, the IETF DetNet Working Group is focused on deterministic data pathways that operate on Layer-3 routed segments. Their objective is to broaden the scope of TSN technologies so that they can span a larger network size in comparison to local area networks. These pathways are distinguished by bounded latency, jitter, and high reliability for each deterministic flow. As a result, they reduce the likelihood of data loss in circumstances when networks are already experiencing an excessive amount of traffic. The deterministic forwarding approach involves the allocation of network resources, such as bandwidth or buffer, to time-sensitive flows. This is accomplished through mechanisms such as the Cycle-Specified Queuing and Forwarding (CSQF) mechanism [10], [11], which allows for the transmission of specific packets to be directed to a specific time. In addition, the configuration of the flow routing and scheduling can be accomplished through the employment of explicit routing methods, such as segment routing.

B. Deterministic Robot Teleoperation System

As the teleoperator exchanges information frequently with robots to enable interactive robotic teleoperation, a Det-RT system is described in this section, it is designed to support high-quality, reliable, and real-time data transmission among the entities. The proposed system consists of a management framework that manages the underlying networks. To support real-time data transmission capabilities of the underlying networks, TSN-capable switches are employed to provide reliable connectivity between robots and teleoperators, while enhancing the schedulability and real-time transmission. Accordingly, the clock should also be synchronized between the robots and user-related devices and the NICs of the devices comprise TSN functionality. For the legacy equipment without clock synchronization and TSN functionality, TSN agents should be developed and deployed at the network edge to shape the traffic from the robots. Given the mobility of robots, the 5G domain [12] should also be upgraded with TSN capabilities deployed in the core networks. Based on this, various traffic shaping mechanisms, such as Time-Aware Shaping (TAS), and

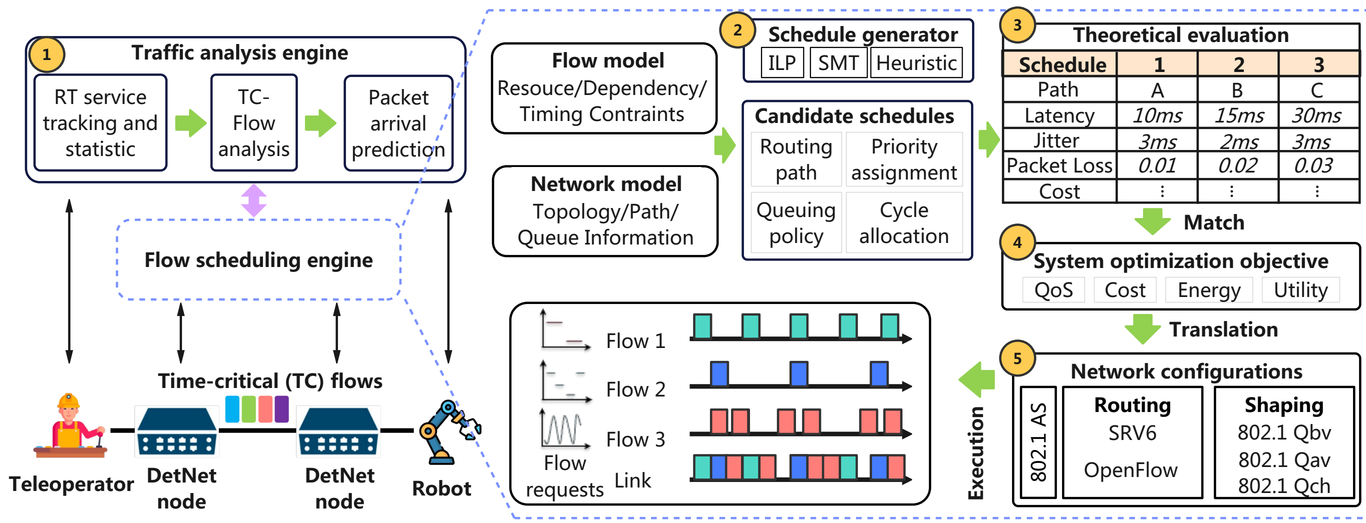

Fig. 2. Illustration of Det-RT system: (1) traffic analysis engine is responsible for collecting information concerning multimodal flow requirement; (2) flow scheduling engine for the flow schedule generation by heuristic algorithm or ILP; (3) theoretical evaluation for the QoS verification by the mathematical tool, e.g., network calculus; (4) set the system optimization objective for the appropriate flow schedules selection; (5) the selected flow schedule is translated into the concrete network configurations.

Asynchronous Traffic Shaping (ATS), can be performed on the TSN switches to schedule the packet forwarding behavior, consequently, the traffic can be transmitted and forwarded, as the control plane instructs, with deterministic QoS performance in terms of data rate, latency, jitter, etc.

C. Deterministic Network Management Solution

The management framework is supposed to make the decisions on how to schedule the traffic flows and then translate them into the corresponding network configurations. In addition, it should also enable adaptability in response to traffic dynamics and changing network conditions. In this section, we propose a general framework for flow scheduling where the scheduler can adapt to different traffic profiles, underlying traffic shaping mechanisms (e.g., synchronized shaping or asynchronous shaping), and system objectives (e.g., minimizing latency or system cost), as shown in Fig. 2.

The proposed framework consists of two components, a traffic analysis engine, and a flow scheduling engine. The whole process can be divided into five steps:

- **Step 1:** The traffic analysis engine is responsible for collecting information concerning multimodal flow requirements such as priority, packet length, and latency requirements. Note that, the flows with multimodal, e.g., video, control, and haptic data, should be coordinated so that the packets of multiple flows which are generated at the same time arrive at the destination simultaneously. Besides, it also traces the production process by learning the traffic pattern of teleoperators and predicting the packet arrivals based on the collected information. Subsequently, the flow request information is transferred into the network decision engine for flow schedule generation.
- **Step 2:** various methods can be employed to generate candidate schedules that meet flow requirements, including approaches such as Integer Linear Programming (ILP), Satisfiability Modulo Theory (SMT), or heuristic-based algorithms [13]. The candidate flow schedule may include route information, priority assignment, queuing policy, or time-slot allocation, depending on the traffic shaping mechanism.
- **Step 3:** Following this, theoretical evaluation should be performed on the candidate flow schedules before translating these schedules into network configurations and executing them. For instance, queuing theory or network calculus [14] can be leveraged to calculate the QoS metric of candidate schedules, such as latency, or packet loss. The objective of this step is to derive the boundary of the corresponding QoS metrics, and only the flow schedules whose boundaries meet the QoS requirements will be selected.
- **Step 4:** The network administrator can opt for the system objectives (e.g., deterministic QoS, minimizing bandwidth utilization, or a combination of them). Consequently, the most appropriate flow schedule is selected to match the given system objective.
- **Step 5:** The selected schedule will be translated into concrete network configurations according to the network protocol specification, for example, the configuration script on cycle time, and gate open time tick for each priority queue regarding 802.1 Qbv.

In the next section, we pay attention to the flow schedule generation strategy while overlooking the traffic analysis, theoretical evaluation, and schedule translation which existing solutions can accomplish [14]. A conflict graph-based flow schedule generation strategy will be proposed to manage the conflicts among the multiple valid schedule candidates, eventually reducing the delay violation and packet drop in the networks.

IV. Deterministic Flow Scheduling for Robotic Teleoperation

A. Cycle-based Traffic Shaping Mechanism

To concretize the proposed flow schedule generation strategy, we take the CSQF protocol as an example to illustrate the workflow of flow schedule generation and selection. Time cycle-based packet scheduling schemes have been investigated recently in [15]. For example, the IETF DetNet Working Group has developed the CSQF protocol as an emerging standard draft. CSQF assumes N queues on ports be alternately open and closed in a cyclic pattern, the open duration of a queue is defined as a cycle. It is designed to support more flexible transmission slot allocation for packets by indicating which queue to insert, where the packets can be delayed by at most $N-1$ cycles. Therefore, the control plane needs to decide on queue assignment and cycle allocation for the packets in time-critical flows to meet the timing requirements. The cycle-based forwarding mechanism is different from the TAS mechanism and no-wait flow scheduling discipline, also known as the zero-queuing principle. In TAS, a gate control list is defined with gate control entries to control the transmission window for a time-sensitive flow in terms of starting time, and transmission duration. The synthesis of gate control and queue management usually makes the scheduling problem with high complexity. Unlike the TAS mechanism where packets can be enqueued, the no-queue principle assumes packets must not be buffered in the networks. The packet that arrives at each network switch must be forwarded immediately to the next hop. It exposes an extremely high requirement for the scheduling between different time-sensitive flows. Conflicts may frequently happen and result in a huge amount of packet drops. Compared with the two solutions mentioned above, the CSQF mechanism overcomes the restriction brought by the TAS mechanism and zero-queue principle. It leverages a simple cyclic queue and forwarding mechanism to schedule time-sensitive traffic by increasing the scheduling space while reducing the complexity of scheduling.

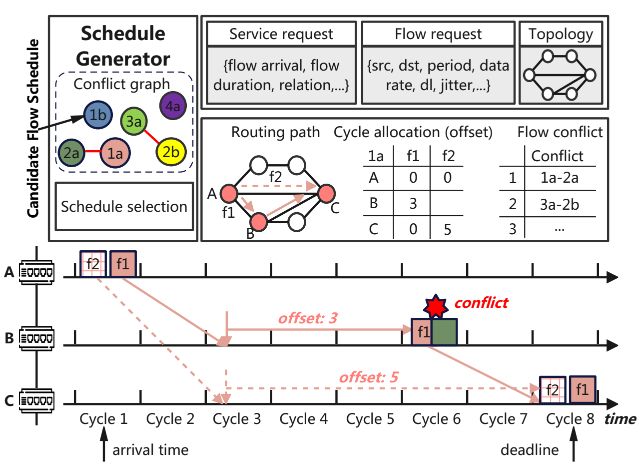

Fig. 3. Conflict graph-based flow schedule generation, a graph which consists of the flow schedules and the conflict between them is established to facilitate the optimal flow schedule selection.

B. QoS Consideration

Upon the arrival of a frame at a specific switch, it undergoes processing for routing and scheduling decisions, which are directly associated with the hardware characteristics and protocol processing. For instance, this may involve tasks such as removing outdated headers and inserting new headers into the payload of an Ethernet frame. Once the determination of the output port and queue for the packet insertion has been made, the frame is subsequently enqueued into the designated queue and awaits transmission, resulting in the occurrence of queuing delay. When a packet is eligible to be transmitted, the transmission delay occurs, which represents the time required to place the packet on the physical link. This delay is usually determined by dividing the frame size by the transmission rate. As an illustration, a 1 $Gbps$ link where an MTU-sized IEEE 802.1Q Ethernet frame, consisting of 1542 $bytes$ to be transmitted, would require a transmission time of 12.336 ms. Ultimately, the frame traverses the physical link, wherein the propagation delay is determined based on the physical distance between the two adjacent nodes in the networks. Generally, we can assume that the transmission latency, processing latency, and propagation latency are deterministic in a small-scale network, as the frame size and transmission rate are known. Hence, it is necessary to regulate the routing and scheduling of flow to derive deterministic queuing delay, eventually to make deterministic E2E latency. Although there are no strict conflicts when implementing CSQF, like TAS and No-wait scheduling, we still need to address the interference between different flows within the same switch, which can be also seen as conflicts. A well-designed flow routing and scheduling strategy should not only satisfy the QoS requirements of the flows, but it can also reduce the network congestion and maximize the network throughput by accepting more time-critical flows without conflicts.

C. Conflict Graph-based Group Flow Scheduling

Different from the traditional industrial scenarios where the time-critical flows are scheduled individually, the traffic flows generated by the robotic teleoperation present some unique features. Robotic teleoperation generates multimodal flows by multiple devices, such as sound sensors, depth cameras, and tactile sensors. On the one hand, the traffic patterns, such as packet size and period, of these flows are different, e.g., the bandwidth haptic feedback is $\geq 128Kbps$ while the VR stream poses huge bandwidth demand $\geq 100Mbps$. On the other hand, the transmission of these flows is supposed to be aligned in terms of timing. In other words, the video data, audio data, and haptic data which are generated at the same time ought to arrive at the destination simultaneously, otherwise, the inconsistent data transmission may result in risky misoperation or operation failure. As a result, the multimodal flows in one robotic teleoperation service need to be scheduled jointly to ensure the consistent QoS metric of flows, which is defined as the upgraded QoS of robotic teleoperation. To this end, we propose a group flow schedule generation approach and discuss how to manage the conflicts between each schedule group in the following sub-sections.

Schedule Generation: To generate the flow schedule for a robotic teleoperation service, the set of multimodal flows should be taken as a whole and modeled as $M = \{src, dst, [l_1, l_2,], [t_1, t_2], dl\}$ if we assume the service contains two flows f_1, f_2, where the src, dst denote the source and destination nodes, $[l_1, l_2]$ and $[t_1, t_2]$ represent the packet size and transmission period, dl is defined as the unified delay requirement. Accordingly, the flow schedules for this application are defined as $S = \{[p_1, p_2], [c_1, c_2]\}$, where $[p_1, p_2]$ and $[c_1, c_2]$ represent the routing path and cycle allocation for flow f_1, f_2. As shown in the bottom of Fig. 3, a candidate flow schedule a for service 1, e.g., $1a$, should include the routing information (e.g., route A-B-C for flow f_1), and cycle allocation (e.g., cycle (0,3,0) for flow f_1). Note that, the transmission cycle allocation is decided by the offset from the arrival time. In a time-slotted system, the flows f_1, f_2 transmitted from the same cycle may arrive at the same time slot through different paths and cycles. As shown in Fig. 3, the flow f_1 and f_2 which belong to the same service are routed through two different paths $A - B - C$ and $A - C$, but their arrival time to the destination should be aligned. Therefore, the group flow schedule $1a$ for f_1 and f_2 is $S = \{[A-B-C, A-C], [(0, 3, 0), (0, 5)]\}$. For instance, the packet of f_1 arrived on node B at $cycle$ 3 and is delayed for 3 cycles. As there might be multiple valid flow schedules for a service, the challenge of traffic scheduling is how to solve the conflict between different group flow schedules.

Conflict Management: As shown in the left side of Fig. 3, there are two valid schedule candidates for service 1: $1a$ and $1b$. The challenge of traffic scheduling is that the schedule selected for a new flow may conflict with the existing flows in the networks, subsequently leading to a degradation in the QoS for the existing flows. For example, the schedule $2a$ for a certain flow of service 2 may result in the overlapping with flow 1 of schedule $1a$ due to the limited cycle capacity of cycle 6 (assuming other packets will also be transmitted in this capacity), which may induce the packet drop of flow 1, so there is a conflict between $1a$ and $1b$. As a result, we need to construct a conflict graph to manage the conflicts between the group flow schedules. The conflict graph helps identify and resolve conflicts between different traffic flows that might arise when they compete for shared resources in a network. To select a valid schedule that does not conflict with other schedules, enough group flow schedules for each service should be first generated based on the service request, e.g., source, destination, and latency requirement. Then they are added to the conflict graph. Next, a conflict table is generated by examining if there is a conflict between two schedules (e.g., $1a-2a$, $3a-2b$), where a link is added between two nodes with conflict (red lines in conflict graph). Eventually, a candidate schedule without conflict can be selected as the active schedule for the service.

D. Flow Scheduling Workflow

Next, we illustrate the workflow of conflict-graph-based group flow schedule generation and selection under the proposed management framework. As shown in Fig. 4, the process starts with two inputs: 1) service requirements, which specify the source, destination, and volume of traffic flow, along with specific requirements like bandwidth, latency, or flow constraints, 2) network information, including details about the network topology, such as links, available capacities, and existing traffic flows. Then, the candidate schedule is generated based on the collected information. Based on the provided inputs, the routing algorithm calculates the candidate paths between source and destination. Subsequently, the transmission cycle allocation is determined, which defines how traffic flows are forwarded when they arrive at each network node. Different allocation policies prioritize flows differently, impacting overall network performance. The system generates potential schedules for all traffic flows, taking into account their assigned priorities, calculated routing paths, and the cycle allocation policy. Next, a conflict graph is created to visually represent potential conflicts based on the generated schedules. An edge between two schedules in the graph indicates that they conflict if implemented simultaneously, as they attempt to use the same resource at the same time. The weight of the schedule in the graph is determined by how many conflicts it contains with other schedules. If conflicts are detected in the graph when selecting a candidate schedule for flows, the system attempts to resolve them. Accordingly, the new schedule should be selected which might involve 1) re-scheduling: adjusting the timing of traffic flows to avoid overlapping resource usage, 2) re-routing: modifying the path of a flow to avoid congested resources, 3) flow rejection: in extreme cases, low-priority flows might be rejected to make way for higher-priority ones. If there is no conflict, the system checks if the selected schedule meets the defined objectives, such as optimizing resource utilization, minimizing latency, or adhering to fair flow allocation. If the schedule is considered suitable then it proceeds to the next step. If not, the system likely goes back to adjust the scheduled ranking and repeats the conflict check and objective evaluation. Finally, the chosen schedule is translated into a configuration script that network devices can understand and implement. This script specifies how traffic flows should be routed and prioritized based on the chosen schedule. The process outputs the generated configuration script, which can be deployed to the network devices to direct traffic flow according to the optimized routing plan.

V. CASE STUDY

We evaluated the performance of the proposed flow scheduling solution for accommodating the time-critical flows derived from the teleoperation services in a Python-based simulation environment. In addition, based on the generated flow schedules, we have translated them into the underlying TSN configurations (i.e., GCL) and sent them to the established testbed to validate the generated flow schedule. In the simulation, we considered a small network topology consisting of four TSN switches and six end nodes, as depicted in Fig. 5(a). In this topology, end node 1 emulates the interactive controller, while end node 2 represents the video stream source, and end node 3 denotes the robot. In addition, three other end nodes act as

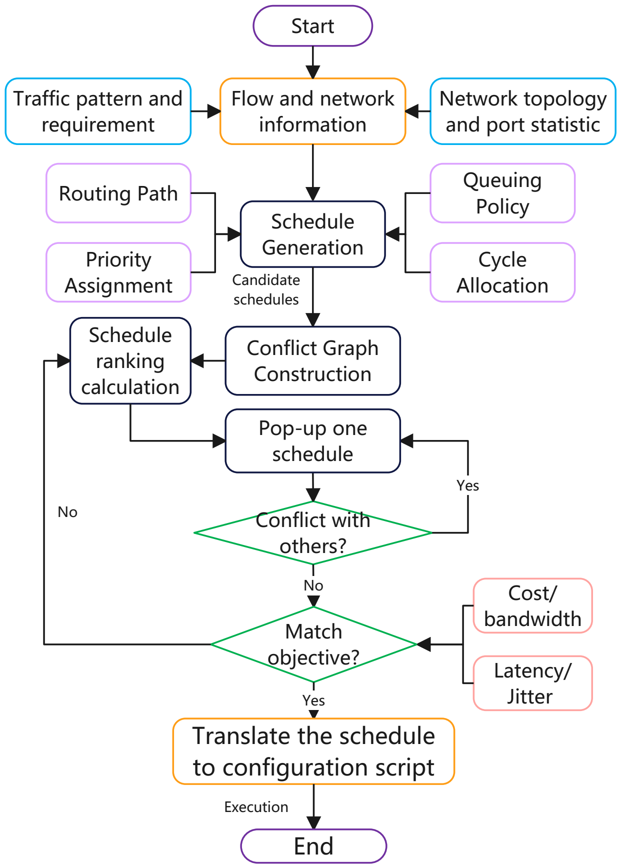

Fig. 4. Workflow of conflict graph-based schedule generation and selection.

background traffic generators. In this scenario, the end node 3 initially transmits the video stream to the end node 2 via TSN switches. Then, the teleoperator makes an operational decision, and the control data are transmitted from the end node 1 to the end node 3. Finally, the end node 3 will return control feedback to end node 1. At the same time, the end nodes 4-6 generate background flows as best-effort traffic, which the TSN switches would not schedule. We generate 10 teleoperation services between the teleoperator and robot, each service consists of the same flows of video and control loop. Within the TSN switch, we assume the CSQF mechanism is supported while enabling the scheduling of time-critical flows based on the priority tag in the MAC header. The frame size of all flows is set to 1500 $bytes$, the link bandwidth is 1 $Gbps$. The sending period of video traffic is set as 1 ms, while the transmission period of control traffic is set as 10 ms, therefore the bandwidth of the video traffic is 10 times the bandwidth of the control stream. The processing and propagation latency were assumed to be zero. For simplification, the cycle duration is set to be 1 ms; therefore, the E2E latency of the flow is a multiple of the millisecond.

In the simulation, the background traffic was first generated and inserted into the networks. Subsequently, the flow scheduling engine created valid schedules for the time-critical flows (i.e., video traffic and control traffic) transmitted between the robot and the teleoperator. As shown in Fig. 5(b), the percentage of background traffic compared to the link bandwidth is gradually increased from the lightly-loaded scenario (10%) to the highly-loaded scenario (90%) and the latencies of video and control stream meet the QoS requirement as specified in Table. I. It is observed that the latency distribution of time-sensitive packets without flow scheduling is higher than the one with flow scheduling mechanism (i.e., jitter), as the transmission time slots for these packets are random and not scheduled. Conversely, the proposed conflict graph-based flow scheduling solution attempts to find the available time slots without conflict according to the latency requirement, as a result, the specific queues in the TSN switches are specially reserved for these streams. In addition, the latency of the video stream is higher than the one of the control stream as we assumed that the traffic load of the video stream is higher than the control stream. It will result in more network congestion which decreases the valid schedule candidates for video traffic.

Next, we evaluate the performance of the proposed group flow schedule approach compared with the individual flow schedule approach. We first define the evaluation metric of the proposed approach. Different from the QoS metric of an individual flow, which is defined as the value of latency requirement divided by the actual latency experienced by the flow (the higher the actual latency, the lower the QoS of this flow), the upgraded metric of a teleoperation service should be defined as the average QoS value of flows within this service. We set the number of the flows that a service contains as N, and the group scheduling case can be seen as the individual scheduling case when $N = 1$. As shown in Fig. 5(c), the upgraded QoS of two approaches start from the same point when $N = 1$, however, the upgraded QoS of the individual scheduling approach will decrease significantly with the increasing of the number of flows within a teleoperation service. If the flow is scheduled individually, schedules should be generated for each single flow without considering the other flows within one service. The schedule generation of a flow may conflict with another flow in one group, resulting in longer delay or packet loss, ultimately, degrading the service QoS.

To demonstrate the effectiveness of the generated flow schedule by our proposed solution, we have established a testbed where two TSN switches (B&R and Kontron TSN switches) are connected with two clients and two servers, as shown in Fig. 5(d). Client 2 emulates the end node of the teleoperator and client 1 for the background traffic node. Client 1&2 sends traffic packets with different profiles to server 1&2 respectively, which represent two robots. Due to the hardware property, the packet sending interval is 1 second (the packet size is the maximum Ethernet packet length) in experiment 1 to emulate e.g., haptic flow, as in Fig. 5(e), while the packet will be sent with a 0.1 second interval in experiment 2 to represent e.g., control flow, as in Fig. 5(f). The experiment duration is 10 seconds and will be run by 10 times in each experiment. We use the proposed flow schedule generation method to derive flow schedules for client 1 in two experiments (no TSN scheduling for client 2), the generated flow schedule will be sent to the Centralized Network Controller (CNC), the CNC will translate the schedule into the TSN configurations and send them to two TSN switches. The results are measured in the bitrate in Mb per second. The obtained results show that

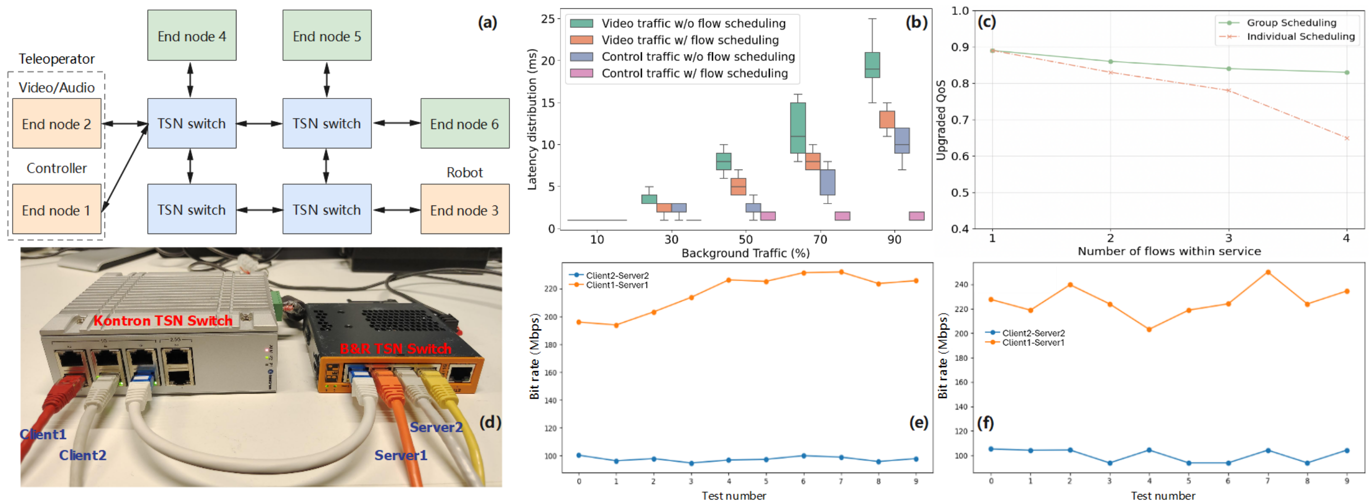

Fig. 5. Simulation and experiment results: (a) the simulated network topology; (b) the comparison of the latency under different background traffic loads; (c) the comparison of group scheduling and individual scheduling in terms of the upgraded QoS; (d) the physical experiment network topology; (e) the received data rate of client1-server1 and client2-server2 under the sending period of 1 second; (f) the received data rate of client1-server1 and client2-server2 under the sending period of 0.1 second.

the client-server pair (client 2-server 2) with TSN scheduling (i.e., reserved transmission window) has a higher bitrate, while the bitrate of background traffic can't be guaranteed due to the best-effort transmission.

VI. Conclusion and Future Directions

To ensure a harmonious integration of human operators and robotic systems, we introduced a system framework for robotic teleoperation. By addressing the challenges and the QoS requirements between the teleoperator and robotic, we proposed the deterministic networking-empowered robotic teleoperation system along with a deterministic network scheduling framework for time-critical robotic teleoperation services. The outcomes of this research contribute significantly to the progression of robotic teleoperation technology. Additionally, they unlock novel avenues for enhancing the integration and cooperation between humans and robots, especially within scenarios involving both proximity and remote operation. In the end, this work identifies several future research directions:

1) **Real-time Dynamic Reconfiguration**: TSN's traffic management, based on predefined configurations, is vulnerable to unexpected network changes. Future work should focus on real-time network status recognition and dynamic traffic reconfiguration without interruptions. Developing intelligent algorithms for continuous monitoring, predictive analysis, and adaptive traffic management, will ensure uninterrupted service quality. Especially, the implementation of intelligent algorithms in a testbed for real-time configuration requires particular attention.

2) **OPC UA over TSN:** OPC Unified Architecture (OPC UA) is a platform-independent, service-oriented architecture widely adopted for industrial automation due to its robust data exchange and interoperability capabilities. OPC UA over TSN will enhance deterministic communication, ensuring low-latency and high-reliability data transmission essential for real-time control at the field level. Future research should focus on developing standardized interfaces and protocols that facilitate seamless OPC UA implementation over TSN.

3) **Integration of 5G and TSN:** Extending TSN to wireless and 5G networks is crucial for more flexible industrial scenarios. This integration promises deterministic, low-latency transmissions essential for industrial applications. Research should adapt TSN standards for wireless environments, focusing on time synchronization, interference management, and bandwidth optimization.

Acknowledgment

This research work is partially supported by the Necoverse project under Grant 10768/31/2022, the Business Finland co-innovation project. This work is also supported in part by the European Union's HE research and innovation program HORIZON-JUSNS-2023 under the 6GPath project (Grant No. 101139172), the European Union's HE research and innovation program HORIZON-JUSNS-2022 under the 6GSandbox project (Grant No. 101096328); and by the AerOS project funded by the European Union's Horizon Europe, the EU's key funding program for research and innovation under Grant No. 101069732. The paper reflects only the authors' views, and the European Commission bears no responsibility for any utilization of the information contained herein.

References

[1] F. Hu, Y. Deng, H. Zhou, T. H. Jung, C.-B. Chae, and A. H. Aghvami, "A vision of an xr-aided teleoperation system toward 5g/b5g," *IEEE Communications Magazine*, vol. 59, no. 1, pp. 34–40, 2021.

[2] A. Khasawneh, H. Rogers, J. Bertrand, K. C. Madathil, and A. Gramopadhye, "Human adaptation to latency in teleoperated multi-robot human-agent search and rescue teams," *Automation in Construction*, vol. 99, pp. 265–277, 2019.

[3] S. Laki, C. Györgyi, J. Pető, P. Vörös, and G. Szabó, "{In-Network} velocity control of industrial robot arms," in *19th USENIX Symposium on Networked Systems Design and Implementation (NSDI 22)*, pp. 995–1009, 2022.

[4] H. N. Qureshi, M. Manalastas, A. Ijaz, A. Imran, Y. Liu, and M. O. Al Kalaa, "Communication requirements in 5g-enabled healthcare applications: review and considerations," in *Healthcare*, vol. 10, p. 293, MDPI, 2022.

[5] European Telecommunications Standards Institute (ETSI), "Speech and multimedia transmission quality (STQ);QoS parameters and test scenarios for assessing network capabilities in 5G performance measurements," Tech. Rep. ETSI TR 103 702, European Telecommunications Standards Institute (ETSI), 2020.

[6] J. Frnda, M. Voznak, and L. Sevcik, "Impact of packet loss and delay variation on the quality of real-time video streaming," *Telecommunication Systems*, vol. 62, no. 2, pp. 265–275, 2016.

[7] D. J. Vergados, A. Michalas, A.-A. A. Boulogeorgos, S. Nikolaou, N. Asimopoulos, and D. D. Vergados, "Adaptive virtual reality streaming: A case for tcp," *IEEE Transactions on Network and Service Management*, vol. 21, no. 2, pp. 1518–1533, 2023.

[8] Telecommunication Standardization Sector of ITU (ITU-T), "Quality of service assurance-related requirements and framework for virtual reality delivery using mobile edge computing," Tech. Rep. ITU-T Recommendation Y.3109, International Telecommunication Union (ITU), 2021.

[9] D. Bruckner, M.-P. Stănică, R. Blair, S. Schriegel, S. Kehrer, M. Seewald, and T. Sauter, "An introduction to opc ua tsn for industrial communication systems," *Proceedings of the IEEE*, vol. 107, no. 6, pp. 1121–1131, 2019.

[10] H. Yu, T. Taleb, K. Samdanis, and J. Song, "Towards supporting holographic services over deterministic 6g integrated terrestrial & non-terrestrial networks," *IEEE Network*, vol. 38, no. 1, pp. 262 – 271, 2023.

[11] S. Wang, B. Wu, C. Zhang, Y. Huang, T. Huang, and Y. Liu, "Large-scale deterministic ip networks on ceni," in *IEEE INFOCOM 2021 - IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, pp. 1–6, 2021.

[12] X. Wang, H. Yao, T. Mai, S. Guo, and Y. Liu, "Reinforcement learning-based particle swarm optimization for end-to-end traffic scheduling in tsn-5g networks," *IEEE/ACM Transactions on Networking*, vol. 31, no. 6, pp. 3254–3268, 2023.

[13] H. Yu, T. Taleb, and J. Zhang, "Deep reinforcement learning based deterministic routing and scheduling for mixed-criticality flows," *IEEE Transactions on Industrial Informatics*, vol. 19, no. 8, pp. 8806–8816, 2022.

[14] L. Zhao, P. Pop, and S. Steinhorst, "Quantitative performance comparison of various traffic shapers in time-sensitive networking," *IEEE Transactions on Network and Service Management*, vol. 19, no. 3, pp. 2899–2928, 2022.

[15] A. Nasrallah, V. Balasubramanian, A. S. Thyagaturu, M. Reisslein, and H. ElBakoury, "Cyclic queuing and forwarding for large scale deterministic networks: A survey," *ArXiv, vol. abs/1905.08478*, 2019.

Chao Yang (chao.1.yang@aalto.fi) received the B.S. degree in mechanical engineering and automation from Jilin University, China, in 2013, and the M.S. degree in mechanical engineering from Aalto University, Finland, in 2021, where he is currently pursuing the Ph.D. degree. His research interests include industrial digital twins, industrial ontologies, and model-based system engineering.

Hao Yu (hao.yu@ictficial.com) received the B.S. and Ph.D. degrees in communication engineering from the Beijing University of Posts and Telecommunications (BUPT), Beijing, China, in 2015 and 2020. He was also a Joint Supervised Ph.D. Student at the Politecnico di Milano, Milano, Italy from 2018 to 2019. He was also a Postdoctoral Researcher in the Department of Information and Communications Engineering, Aalto University, Finland from 2021 to 2022, and in the Center of Wireless Communications, University of Oulu, Finland from 2022 to 2024. Currently, he is working in the ICTFicial Oy, Finland as a senior researcher. His research interests include network automation, SDN/NFV, edge intelligence, time-sensitive networks, and 6G deterministic networking.

Qize Guo (qize.guo@ictficial.com) received the B.E. degree in communication engineering from Information Engineering University, China, in 2009. He received the Ph.D. degree in communication engineering from the Beijing University of Posts and Telecommunications (BUPT), Beijing, China, in 2023. He was a Postdoctoral Researcher at the Center of Wireless Communications, University of Oulu, Finland. He is currently working in the ICTFicial Oy, Finland as a senior researcher. His research interests are mainly in the field of optical transport networks, mobile networks, and network slicing optimization.

Tarik Taleb (tarik.taleb@ruhr-uni-bochum.de) is currently a Full Professor at Ruhr University Bochum, Bochum, Germany. He was a Professor at the Center of Wireless Communications, University of Oulu, Oulu, Finland. Between Oct. 2014 and Dec. 2021, he was a Professor at Aalto University. He also worked as an assistant professor at Tohoku University. He holds a B.E. degree in information engineering and M.Sc. & Ph.D. degrees in information sciences from Tohoku University.

Jose Costa Requena (jose.costa@aalto.fi) is working continuously in academia and industry. He has been a researcher at TKK now Aalto University since 1998 and coordinates national and EU projects. In parallel, he worked at Nokia from 1999- 2009 as Senior Technology Manager. He also has an entrepreneurial path and founded Cumucore as an Aalto spinoff.

Kari Tammi (kari.tammi@aalto.fi) has been a Professor at Aalto University since 2015. He currently works as Dean for the School of Engineering. He also serves the Finnish Administrative Supreme Court as a Chief Engineer Counselor. He received the M.Sc., Lic.Sc., and D.Sc. degrees from the Helsinki University of Technology, in 1999, 2003, and 2007, respectively. He received a Teacher's pedagogical qualification at the Hame University of Applied Sciences, in 2017. He was a Researcher with CERN, the European Organization for Nuclear Research, from 1997 to 2000, and a Postdoctoral Researcher with North Carolina State University, USA, from 2007 to 2008. From 2000 to 2015, he held a Research Professor, a Research Manager, a Team Leader, and other positions at the VTT Technical Research Centre of Finland. He has authored over 120 peer-reviewed publications cited in over 7000 other publications. He is a member of the Finnish Academy of Technology and the President of The Association of Automotive Technical Societies in Finland.