Demo: DetNet Packet Replication and Elimination for Resilience in 6G

Florian Raskob[1], Eric Mittag[2], Tobias Meuser[1], Hans D. Schotten[2,3], Björn Scheuermann[1]
[1]Communication Networks Lab, Technical University of Darmstadt, Germany
[2]German Research Center for Artificial Intelligence (DFKI), Kaiserslautern, Germany
[3]RPTU University Kaiserslautern-Landau, Germany
E-Mail: {florian.raskob, tobias.meuser, scheuermann}@kom.tu-darmstadt.de, eric.mittag@dfki.de, schotten@rptu.de

***Abstract*—This demonstration showcases how integrating the IETF Deterministic Networking (DetNet) Packet Replication, Elimination, and Ordering Functions (PREOF) into the 6G architecture improves resilience and latency in an industrial scenario. Our setup connects two synchronized robotic arms via two disjoint and heterogeneous 6G networks, one employing a hardware-based and the other a software-based User Plane Function (UPF). Packets are replicated across both paths and eliminated upon reception, ensuring robust communication even in the presence of link or component failures. Additionally, latency is minimized as the receiver forwards the first arriving packet.**

***Index Terms*—6G, DetNet, mobile networks, resilience, URLLC, heterogeneous networks**

I. INTRODUCTION

Emerging technologies such as the Internet of Everything (IoE), Vehicle-to-Everything (V2X), and digital twins increasingly rely on 6G mobile networks, making society itself ever more dependent on this critical infrastructure [1]. Consequently, ensuring the resilience of 6G against potential threats becomes paramount [2]. Furthermore, these technologies make stringent demands on future 6G networks in terms of reliability and latency. To meet these quality of service (QoS) requirements, the user plane of 6G networks plays a crucial role, as it forwards all data traffic. In the core network, data traffic is forwarded by the User Plane Function (UPF), making its reliability and overall performance critical factors [3].

Deterministic Networking (DetNet) is a common approach to achieving very high reliability and is actively developed within the IETF. RFC 8655 [4] specifies the Packet Replication, Elimination, and Ordering Functions (PREOF), which enables the use of redundant data paths by replicating packets at the sender and encapsulating them with headers carrying sequence numbers or timestamps. Each replica is sent along a separate, disjoint path to reduce the impact of component failures. At the receiver, only the first-arriving replica is forwarded, as identified by the additional header information. Finally, packets that arrive out of order are reordered based on the sequence information. Hence, PREOF improves reliability, as packets can still be transferred even if one of the paths fails. In addition, it can reduce latency because packets are always selected from the fastest path.

Ihle et al. [5] explored how DetNet's PREOF can be applied in 6G networks, proposing two approaches for redundant 6G user planes, each resulting in two disjoint data paths. To validate the in-field effectiveness of this concept in improving latency and reliability, we built a demonstrator, inspired by this work. It comprises two heterogeneous UPF implementations that connect two synchronized robotic arms for remote operation, representing a potential use case in a smart factory scenario.

In Section II, we present the system architecture of our demonstrator, and in Section III, we describe its implementation and presentation. Finally, we present our conclusions in Section IV.

II. SYSTEM ARCHITECTURE

Inspired by the conceptual integration of DetNet's PREOF in 6G by Ihle et al. [5], we design a system that demonstrates this concept in an industrial setting. Figure 1 outlines the system, comprising two robotic arms, a Packet Replication and Elimination Function (PREF), and two disjoint 6G networks. Within each 6G network, a gNodeB (gNB) provides the radio access link connecting the PREF to the UPFs. The robotic arms cannot communicate directly; instead, we forward all packets they send to the PREFs, which send one replica to each of the disjoint 6G networks. On the data network (DN) site of each 6G network, we deploy a backend sending all uplink packets back downlink to the robotic arms. Deploying two redundant backends increases the robustness against failing components; however, depending on the scenario, there might be only one for both networks. In the following, we discuss the robotic arms and our PREF in more detail.

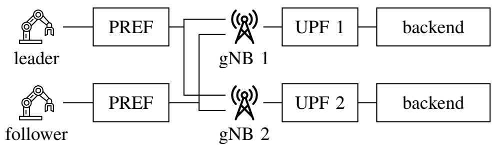

Fig. 1. System Design.

A. Robotic Arms

The robotic arms demonstrate a teleoperation scenario, where one of the collaborative robots (the follower) mimics the movements of the other robot (the leader). The setup builds on the open-source teleoperation example provided by

ISBN 978-3-903176-82-9 © 2026 IFIP

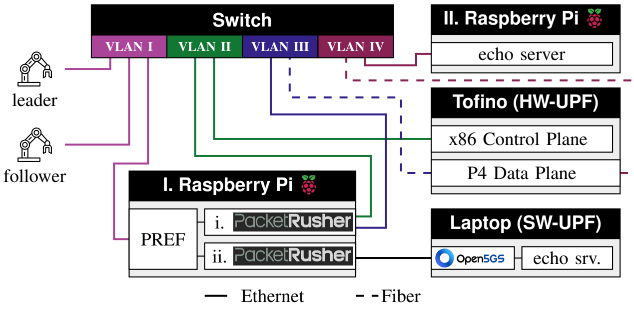

Fig. 2. Demonstrator Outline.

Fig. 3. Demonstrator.

Franka Robotics [6] and was extended by Petershans et al. [7] to run on two separate control computers, enabling the integration of different communication technologies between them. Mittag et al. [8] subsequently migrated the system from the Robot Operating System (ROS) to ROS 2, incorporating the Data Distribution Service (DDS) as middleware to support multi-connectivity on the application layer and to demonstrate redundant communication links. In the present work, however, only one communication path is employed, connecting the teleoperation system to the PREOF.

After aligning their initial poses, the leader continuously transmits its joint states (position, velocity, and effort) to the follower, which mirrors them to reproduce the leader's motion. In the opposite direction, the follower sends its estimated external torques to the leader to enable force feedback to the operator. During this teleoperation phase, the two robots exchange bidirectional high-frequency traffic. If the communication latency or jitter between them is high, the robots can desynchronize, leading to rapid movement of the leader and potential danger to the operator. This high-frequency bidirectional traffic represents a realistic and demanding ultra-reliable and low-latency communication (URLLC) workload, serving as an ideal basis to evaluate the performance of the integrated 6G communication system.

B. *Packet Duplication & Duplicate Elimination*

PREOF defined by RFC 8655 comprises three functions [4]. The Packet Replication Function (PRF) replicates incoming packets, adds sequence information to them, and sends them out over multiple disjoint paths. The Packet Elimination Function (PEF) is placed at the receiver's end and eliminates replicas using the sequence information added by the PRF, forwarding only the first-arriving packet and dropping all subsequent ones. The Packet Ordering Function (POF) reorders packets received out of order based on the sequence information.

To create two redundant data paths, we implement a PREF integrating the PRF and the PEF. It replicates packets received from the robotic arms, inserts an additional replication header between the IP and TCP layers that contains an incrementing 16-bit sequence number, and finally sends one replica over each path. Whenever it receives a packet from the 6G network, it only forwards the first replica and drops any subsequent ones. It uses a bitmap initialized with zeros, where each entry corresponds to a sequence number. In case the entry for the sequence number of a received packet is `0`, the replication header is removed, the bit is set to `1`, and the packet is forwarded to the Radio Access Network (RAN). If the bit for the sequence number of a packet received is already `1`, the packet is dropped. Every time the bitmap is accessed, the opposite half of the bitmap is cleared.

III. DEMONSTRATION

Based on our system design, we built a demonstrator, which is outlined in Figure 2 and shown in Figure 3. It consists of two Franka Research 3 robotic arms, two Raspberry Pis, a laptop, an Intel Barefoot Tofino, and a switch providing network connectivity. In the following, we present our demonstrator, which employs two redundant, heterogeneous 6G networks. We first present the individual components and subsequently demonstrate how our setup showcases the effectiveness of our system design in improving latency and reliability.

A. *Robotic Arms*

Both Franka Research 3 robotic arms and the first Raspberry Pi connect to VLAN I, through which packets sent between the robotic arms are routed to the Raspberry Pi, so the two robotic arms cannot exchange packets directly. Each robot is controlled by an industrial PC. Both systems run Debian 12.8 with a real-time Linux kernel (version 6.1.0-27-rt) to ensure deterministic timing behavior. The teleoperation software runs inside Docker containers based on Ubuntu 22.04, which host the ROS 2 Humble environment. The industrial PCs connect to VLAN I via 10 Gbit dual-port network interface cards based on the Intel X540-T2 chipset. The ROS 2 topic messages are published at 1 kHz, resulting in data rates of approximately 1.2 Mbit/s from the leader to the follower and 0.7 Mbit/s in the reverse direction for the raw ROS 2 topics. When measuring encapsulated DDS traffic on network interfaces, these rates increase to around 4 Mbit/s leader-to-follower and 3.5 Mbit/s follower-to-leader due to protocol overhead.

B. *PREF and RAN*

Since all packets from the robotic arms are routed to the first Raspberry Pi, we deploy our PREF on it to replicate all incoming packets. The replicas are forwarded to two separate network namespaces (NSs), each running an instance of PacketRusher [9] that simulates a user equipment (UE) and a gNB. VLAN II and VLAN III connect the gNB in the first NS to the Tofino, which runs a 6G core network with a hardware-based user plane. VLAN II carries the control traffic, while VLAN III handles the user traffic. This is necessary as two different interfaces are used on the Tofino for control and user traffic. User traffic is sent from the Raspberry Pi via Ethernet to the switch on VLAN III, which then forwards it to the Tofino via a 100 Gbit/s fiber cable.

The gNB in the second NS connects to the Laptop running a purely software-based 6G core network. In this case, both control and user traffic are sent directly to the Laptop over one Ethernet cable.

C. 6G Core Network

Our setup comprises two 6G core networks, one on the Tofino switch and one on the laptop. On the x86 unit of the Tofino switch, we deploy the free5GC[1] control plane, while on the Tofino chip, we deploy the P4 UPF implementation provided by Kundel et al. [10]. Since this UPF is hardware-based, it provides significantly lower latency than software-based implementations [10]. VLAN IV connects the second Raspberry Pi, which serves as an echo server with the DN interface of the UPF.

On the laptop, we deploy Open5GS[2], another implementation of the 6G core network, including a software-based UPF. To echo all packets leaving the UPF via its DN interface, we configure the laptop by setting respective nftables and routing rules.

D. Demonstration

To demonstrate our system, we disconnect the fiber cables connected to the Tofino, as shown in Figure 4, or the Ethernet cable connected to the laptop while using the robotic arms. As a result, one can observe that the robotic arms remain operational and synchronized without any notable delays or unintended movements. When we measure the latency between both robotic arms, we observe that when disconnecting the fiber cables connected to the Tofino the average Round-Trip Time (RTT) increases from 0.21 ms to 1.03 ms and the jitter increases from 0.01 ms to 0.24 ms, as the packets are forwarded solely by the software-based UPF. When disconnecting the cable connected to the laptop, RTT and jitter remain the same. As the P4 UPF archives lower latency than the software-based UPF, this behavior is expected. When reconnecting the fiber of the Tofino, the redundant hardware-based path is reestablished and the RTT decreases again without any degradation in synchronization.

E. Presentation

Since transporting the demonstrator to the conference venue is not feasible due to logistical constraints, we created a video[3] to be presented during the demonstrator session, along with a poster providing a more detailed overview.

Fig. 4. Unplugging the hardware-based UPF.

IV. CONCLUSION

Inspired by Ihle et al. [5], this demonstration integrates DetNet's PREOF into the 6G network architecture. By connecting two Franka Research 3 robotic arms via redundant 6G networks employing heterogeneous UPFs, we demonstrate the practical benefits of this concept in improving latency and reliability through packet replication and elimination.

ACKNOWLEDGMENT

The authors acknowledge the financial support by the Federal Ministry of Research, Technology and Space of Germany in the project “Open6GHub+” (grant numbers: 16KIS2407 and 16KIS2402K), and by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) – 503231190.

REFERENCES

[1] C.-X. Wang *et al.*, “On the Road to 6G: Visions, Requirements, Key Technologies, and Testbeds,” *IEEE Commun. Surveys Tuts.*, vol. 25, no. 2, pp. 905–974, 2023.

[2] L. Khaloopour *et al.*, “Resilience-by-Design in 6G Networks: Literature Review and Novel Enabling Concepts,” *IEEE Access*, vol. 12, pp. 155 666–155 695, 2024.

[3] F. Raskob, W. A. Poth, T. Meuser, and B. Scheuermann, “Monitoring 6G UPFs: A Software-based Network Tomography Framework,” in *IFIP Netw. IOCRCI Workshop*, 2025.

[4] N. Finn, P. Thubert, B. Varga, and J. Farkas, “Deterministic Networking Architecture,” RFC 8655, 2019.

[5] F. Ihle, T. Meuser, M. Menth, and B. Scheuermann, “Packet Level Resilience for the User Plane in 5G Networks,” 2025, arXiv:2501.12345.

[6] “Franka Control Interface (FCI) documentation,” https://frankarobotics.github.io/docs/franka_ros/docs/index.html, accesed: Mar. 2026.

[7] J. Petershans, J. Herbst, M. Rüb, E. Mittag, and H. D. Schotten, “Robotic Teleoperation: A Real-World Test Environment for 6G Communications,” in *ITG-Fachtagung Mobilkommunikation*. VDE-Verlag, 2024, pp. 106–111.

[8] E. Mittag *et al.*, “Introducing a 6G-Enabled Multi-Connectivity Robotic Teleoperation Platform,” in *IEEE Int. Conf. Commun. Workshops*, 2025, pp. 1851–1857.

[9] V. D’Emmanuele and A. Raguideau, “PacketRusher: High performance 5G UE/gNB Simulator and CP/UP load tester,” 2023, https://github.com/HewlettPackard/PacketRusher.

[10] R. Kundel, T. Meuser, T. Koppe, R. Hark, and R. Steinmetz, “User Plane Hardware Acceleration in Access Networks: Experiences in Offloading Network Functions in Real 5G Deployments.” in *Hawaii Int. Conf. Syst. Sci.*, 2022.

---

[1] https://free5gc.org/

[2] https://open5gs.org/

[3] https://youtu.be/nKls2u-buh8