Telepresence-Enhanced Network Music Performance

George Xylomenos, Konstantinos Tsioutas, Yannis Thomas, Iakovos Pittaras, Chalima Dimitra Nassar-Kyriakidou,
Athanasia Maria Papathanasiou, Fotios Bistas, Ioannis Barous and Anna Kefala
Mobile Multimedia Laboratory, Department of Informatics,
School of Information Sciences and Technology, Athens University of Economics and Business, Greece

***Abstract*—5G can become an enabler for eXtended Reality (XR), especially considering the dropping cost of XR hardware. Network Music Performance (NMP) can greatly benefit from 5G, as its latency requirements (up to 30-40 ms) cannot be met by 4G. While audio can be sent uncompressed to reduce delays, 2D video is harder, as it is too bandwidth-heavy to send uncompressed, and compression introduces latency. Volumetric video can transform the NMP experience, if its bandwidth requirements can be met by 5G networks. The Telepresence-Enhanced Network Music Performance (TENeMP) project carried out a series of experiments in the Berlin 5G testbed provided by the SPIRIT project, to assess whether the integration of NMP with telepresence over 5G can make NMP a reality.**

***Index Terms*—XR, 5G, NMP**

I. Introduction

5G can become an enabler for eXtended Reality (XR), especially considering the dropping cost of XR hardware. Network Music Performance (NMP) can greatly benefit from 5G, as its latency requirements (up to 30-40 ms) cannot be met by 4G. While audio can be sent uncompressed to reduce delays, 2D video is harder, as it is too bandwidth-heavy to send uncompressed, and compression introduces latency. Volumetric video can transform the NMP experience, if its bandwidth requirements can be met by 5G networks. The Telepresence-Enhanced Network Music Performance (TENeMP) project carried out a series of experiments in the Berlin 5G testbed provided by the SPIRIT project, to assess whether the integration of NMP with telepresence over 5G can make NMP a reality.

II. Testbed setup

In NMP there are two basic communication scenarios, the Peer-to-Peer (P2P) scenario where each musician directly communicates with all others, and the Client-SFU (CSFU) scenario where musicians communicate indirectly, via an SFU, acting as a relay between them. The SFU must ideally be located close to the endpoints, e.g., at a Mobile Edge Cloud (MEC) server.

Although our primary goal was to test NMP in the SPIRIT 5G Standalone (5G-SA) testbed in Berlin, for development and testing we created a testbed in Athens, replicating as far as possible the setup of the Berlin testbed, with Intel RealSense D435 depth cameras, XREAL Air 2 Pro glasses and Teltonica RUTX-50 5G routers. As shown in Figure 1, the Athens testbed has two 5G endpoints, as well as a server in the MMlab. The endpoints used for the measurements were Asus TUF Gaming A15 laptops, running Ubuntu 24.04.2 LTS, with embedded Realtek sound cards and 720P HD Logitech USB web cameras (for 2D video).

The Athens testbed had three main differences with the Berlin testbed. First, in Athens we only had a 5G Non-Standalone (5G-NonSA) network (COSMOTE/TELEKOM), while in Berlin the network was 5G-SA. Second, the Berlin testbed included MEC servers in the 5G cell, while in Athens servers were deployed in our LAN. Third, the Berlin testbed was isolated from the Internet, using private IP addresses. The Athens testbed used a public 5G network, which used NAT. Initially, we relied on STUN and TURN servers for NAT traversal, deploying a signaling server [1], [2]. However, we eventually adopted UDP hole punching to establish communication between the endpoints in the Athens testbed.

III. Measurement procedures

For audio, we measured the Mouth to Ear (M2E) latency, which is the time between a user producing a sound and the sound reaching the ears of another user. We used the reflected pulse method, shown in Figure 2: we produce audio pulses, which are sent to a receiver, played back there, captured again and returned to the sender. We record both the original and the returned signal as stereo channels and inspect the recorded waveforms to calculate the round trip delay. With a symmetric connection, half of that is the M2E latency.

For 2D video, the Glass to Glass (G2G) latency is to the time it takes for a frame to be captured by a camera and the corresponding frame to be presented at a screen. We adapted the flashing LED method [3], shown in Figure 3, where a LED is pointed at a web camera, the captured video is displayed at the other end and a light detector captures the LED flash. The same microcontroller lit the LED and monitored the light detector, thus measuring G2G latency. For volumetric video, we tested various options to reduce the video stream size, including dropping color information and reducing resolution; this prevented us from using the flashing LED method. We thus measured volumetric latency by jointly analyzing the logs of the consumer and producer applications [4].

IV. Results

For audio and 2D video, the measurements at both Berlin and Athens used Gstreamer in Linux, in either P2P or CSFU

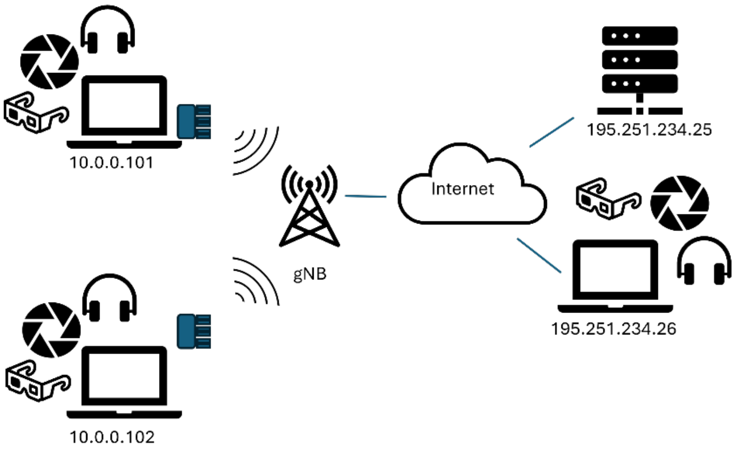

Fig. 1. Athens testbed.

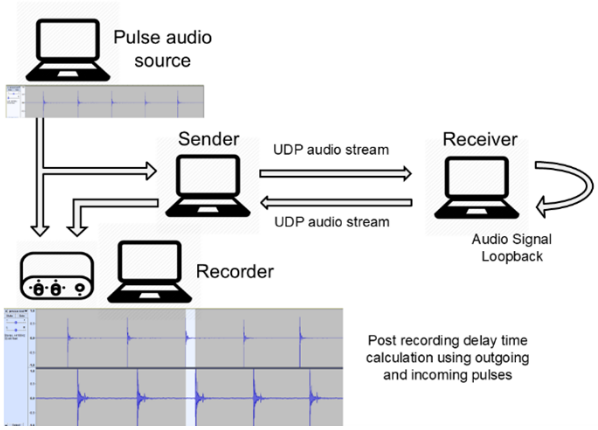

Fig. 2. Audio latency measurement setup.

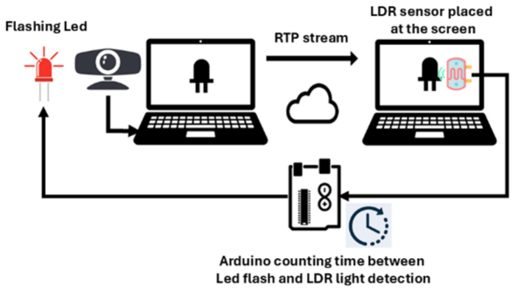
Flashing Led
RTP stream
LDR sensor placed at the screen
Arduino counting time between Led flash and LDR light detection

Fig. 3. Video latency measurement setup.

mode. For audio, we used PulseAudio to capture an audio channel at a 44.1 KHz with 16 bits per sample. The buffer size was set to 132 samples, or 3 ms of audio. The RTP packets for the uncompressed audio stream had a size of 276 bytes, including the 12-byte RTP header. In CSFU mode, we used another Gstreamer pipeline at the MEC server; in the Berlin testbed, this was inside the cell, in the Athens testbed it was in our lab. The pipeline simply relayed incoming UDP packets from one client to the other.

Figure 4 shows boxplots for the M2E audio delay. The median in P2P mode was 40.5 ms for the 5G-SA network, lower than the 45.1 ms for the 5G-NonSA network. In CSFU mode, there was a much larger difference between 5G-SA and 5G-NonSA (74.4 ms vs. 112.2 ms), since in the 5G-NonSA case the SFU was outside the 5G cell.

For 2D video we used the Video4Linux plugins to capture video and H.264 to compress it. In the Berlin testbed we used 1400-byte UDP packets with an RTP payload. For the CSFU topology, we again used a Gstreamer pipeline. In the Athens testbed, that packet size led to losses, therefore we reduced it to 300 bytes; this meant higher overhead due to headers, but led to good video quality. Figure 5 shows boxplots for the G2G video delay. The median latency in the Berlin 5G-SA testbed (87 ms) was lower than in the Athens 5G-NonSA testbed (111 ms), far more than in for audio. In CSFU mode, we also had a large difference between the two testbeds, since in the Athens testbed the SFU was located outside the 5G network.

For volumetric video, we developed our own Point Cloud (PC) streaming tool, using the Intel RealSense SDK, the Google Draco encoder and OpenGL for rendering. The PC frame was captured at 848x480, using a relatively complex scene with 1-2 m of depth, producing around 70% of the maximum PC size. In the experiments, we controlled four parameters: (a) Draco compression levels (1-10) (b) different PC sizes (full frame, 25% and 50% fewer points), (c) 1 or more compression threads and (d) color or no color information; details can be found in [4]. Figure 6 depicts the end-to-end latency of 1000 PCs when using 2 threads for compression with 50% dropped points. The processing latency was roughly 30 ms, supporting 30 FPS, with 97.2% of frames received correctly. The graph suggests that the minimum possible end-to-end latency was roughly 80 ms, albeit with a considerable variance.

References

[1] K. Tsioutas, Y. Thomas, F. Bistas, I. Barous, G. Xylomenos, and G.C. Polyzos, “Network Music Performance Beyond 4G,” in *Proceedings of the IEEE International Wireless Communications & Mobile Computing Conference (IWCMC)*, 2025.

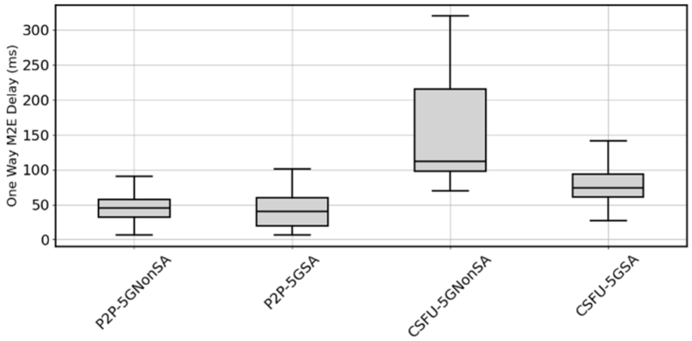

Fig. 4. Audio latency.

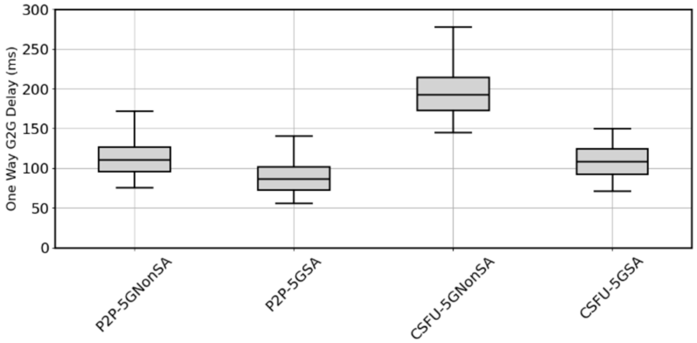

Fig. 5. 2D video latency.

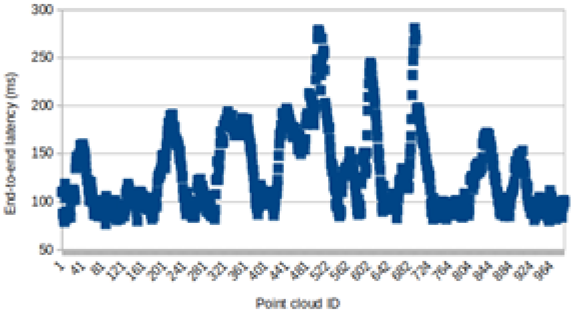

Fig. 6. Volumetric video latency.

[2] K. Tsioutas and G. Xylomenos. Polyzos. Network Music Performance in 5G Networks. Submitted to IEEE International Symposium on the Internet of Sounds (IS2), 2025.

[3] C. Bachhuber, and E. Steinbach, “A system for high precision glass-to-glass delay measurements in video communication,” in *Proceedings of the IEEE International Conference on Image Processing (ICIP)*, pp. 2132–2136, 2016.

[4] Y. Thomas and G. Xylomenos. Ultra-low Latency Point Cloud Streaming in 5G. Proceedings of EuroXR International Conference, 2025.