5G-Enabled Internet of Musical Things Architectures for Remote Immersive Musical Practices

Received 26 April 2024; revised 20 May 2024; accepted 28 May 2024. Date of publication 31 May 2024; date of current version 6 August 2024.

Digital Object Identifier 10.1109/OJCOMS.2024.3407708

5G-Enabled Internet of Musical Things Architectures for Remote Immersive Musical Practices

LUCA TURCHET[1] (Senior Member, IEEE), CLAUDIA RINALDI[2] (Member, IEEE), CARLO CENTOFANTI[3] (Member, IEEE), LUCA VIGNATI[1], AND CRISTINA ROTTONDI[4] (Senior Member, IEEE)

[1]Department of Information Engineering and Computer Science, University of Trento, 38123 Trento, Italy

[2]Research Unit of L'Aquila, National Interuniversity Consortium for Telecommunications, 43124 parma, Italy

[3]Department of Information Engineering, Computer Science and Mathematics, University of L'Aquila, 67100 L'Aquila, Italy

[4]Department of Electronics and Telecommunications, Politecnico di Torino, 10129 Turin, Italy

CORRESPONDING AUTHOR: L. TURCHET (e-mail: luca.turchet@unitn.it)

This work was supported in part by the European Union through the Italian National Recovery and Resilience Plan of NextGenerationEU, with the MUR PNRR PRIN 2022 Grant under Grant 2022CZWWKP, and with the Partnership on "Telecommunications of the Future" (Program "RESTART") under Grant PE00000001, and in part by the European Union through the Project H2020-MSCA-RISE-2019 OPTIMIST Grant under Grant 872866.

ABSTRACT Networked Music Performances (NMPs) involve geographically-displaced musicians performing together in real-time. To date, scarce research has been conducted on how to integrate NMP systems with immersive audio rendering techniques able to enrich the musicians' perception of sharing the same acoustic environment. In addition, the use of wireless technologies for NMPs has been largely overlooked. In this paper, we propose two architectures for Immersive Networked Music Performances (INMPs), which differ for the physical positions of the computing blocks constituting the 3D audio toolchain. These architectures leverage a backend specifically conceived to support remote musical practices via Software Defined Networking methods, and take advantage of the orchestration, slicing, and Multi-access Edge Computing (MEC) capabilities of 5G. Moreover, we illustrate how to integrate in the architectures machine learning algorithms for network traffic prediction and audio packet loss concealment. Traffic predictions at multiple time scales are utilized to achieve an optimized placement of Virtual Network Functions hosting audio mixing and processing functionalities within the available MEC sites, depending on the users' geographical locations and current network load conditions. An analysis of the technical requirements for INMPs using the two architectures is provided, along with their performance assessment conducted via simulators.

INDEX TERMS Internet of Musical Things, 3D audio, networked music performance, 5G, quality of service, software-defined networking.

I. INTRODUCTION

Nowadays, a wide range of music-related activities can be remotely supported by Web-mediated technologies, thus unleashing unprecedented opportunities to foster access and diffusion of musical cultural heritage at artistic and commercial levels. Among those, Networked Music Performances (NMPs) involve multiple geographically-displaced musicians performing together in real-time thanks to low-latency audio streaming over a telecommunication network [1], [2]. NMP systems are used in a variety of musical practices, including rehearsals, concerts, and pedagogy [3], and their need has become prominent during the recent COVID-19 pandemic [4]. NMP systems are one of the essential components of the Internet of Musical Things (IoMusT), the emerging field that extends the Internet of Things paradigm to the musical domain [5].

Noticeable examples are JackTrip [6], LoLa [7], fast-music [8], UNISON [9], and Elk LIVE [10].

Thus far, most of the research on NMP systems has focused on how to reduce the latency and improve the quality of the streamed audio content [11], [12], along with a set of perceptual studies validating the systems, as well as on the identification of design requirements for them [13], [14]. According to several studies [1], to guarantee performative conditions similar to those that would occur in a shared physical space, the experienced End-to-End (E2E) latency must be maintained below 30 ms and high-fidelity audio quality must be ensured, i.e., the audio artifacts caused by packet losses must be minimal. When latency is above such a threshold, a large body of research has consistently shown that musicians are not capable of synchronizing [13], [14], [15]. Other studies have assessed the perceived audio quality during an NMP session showing the need for minimizing packet losses to avoid negative effects on the musicians' playing experience [16], [17]. Nevertheless, another important aspect contributes to the musicians' perception of realism during remote musical interactions, i.e., the real-time rendering of the acoustic scene, such that each connected musician has the perception of sharing the same acoustic environment as the others. This perception relates to the so-called "social presence" (i.e., the sensation of "being there" in the virtual environment with other users), which is a crucial factor in collaborative virtual environments [18].

To enrich the musicians' perception of sharing the same acoustic environment, it is necessary to integrate NMP systems with spatialized audio rendering techniques, which enable a three-dimensional localization of audio sources [19]. Moreover, to further immerse the user in the acoustic scene there is the need of applying room acoustic modeling techniques [20], which can simulate the type of room in which musicians virtually play (e.g., a concert hall or a rehearsal room). Nevertheless, current immersive audio solutions have been optimized for local streaming or cloud transmission, without strictly adhering to low-latency and high-reliability requirements. Their integration in NMP systems entails a high number of audio channels to be streamed and mixed to ensure spatialized reconstruction of the musical scene at each remote location, thus pushing latency, synchronization, and bitrate requirements to become even more challenging. To date, the technical challenges underlying such integrations have been largely overlooked in both academia and industry, with only a handful of works preliminarily exploring this topic [21].

In a different vein, most NMP systems have been used with an underlying wired networking infrastructure. Some NMP scenarios involving wireless communications have also been considered [22]. However, NMP applications leveraging wireless transmission are still heavily constrained by technological limitations in terms of latency and reliability, since wireless communications must cope with much higher packet loss rates in comparison to wired media [23], [24]. In NMP, the audio transfer through a wireless channel must be extremely reliable and fast, and should experience little if any outage, so that low-complexity error correction schemes can compensate for missing data packets. These stringent requirements on the Quality of Service (QoS) and Quality of Experience (QoE) impose the use of ultra-reliable low-latency wireless communication, which is a promise of the fifth generation (5G) of cellular networks.

5G was conceived to provide significantly better Key Performance Indicators (KPIs) compared to its 4G counterpart and to overcome a number of shortcomings thereof [25]. Such KPIs include lower radio access network (RAN) latency, higher-bitrate data communications, faster and more scalable transmission scheduling, as well as a more flexible core network infrastructure, including virtualized network functions and edge-side computation (Multi-Access Edge Computing – MEC). In particular, 5G introduces the concept of network slicing, where the physical network can be divided into multiple isolated logical sub-networks of varying sizes and structures, which are dedicated to different types of services based on their requirements [26]. In 5G networks, Software-Defined Networking (SDN) and Network Function Virtualization (NFV) allow supporting programmable control and management of network resources. The 5G integration with SDN, NFV, and slicing technologies for IoT applications has already been widely investigated [27]. However, to the best of the authors' knowledge, this integration has not been performed yet for the specific case of NMP and the IoMusT [28]. Only a handful of studies have speculated the integration of 5G in NMP systems [16], [23], [24], [29]. Indeed, to ensure the QoS required by NMP applications, a specifically-tailored backend network infrastructure is necessary to guarantee i) low-latency transmission, ii) intelligent placement of audio mixing and processing VNFs, iii) integration of dedicated algorithms for traffic prediction (see [30] for a survey on such techniques) and audio packet loss concealment.

To bridge these gaps, in this paper we propose two architectures for INMPs, which leverage the 5G cellular network infrastructure and an SDN-enabled backend specifically conceived to support remote musical practices. Such architectures take advantage of the orchestration, slicing, and MEC capabilities of 5G. Moreover, we illustrate how to integrate in the architectures Machine Learning (ML) algorithms for network traffic prediction and audio packet loss concealment. Traffic predictions at multiple time scales are utilized to achieve an optimized placement of VNFs within the available MEC sites, depending on the users' geographical locations and current network load conditions. Concerning immersive audio rendering, two different architectures are designed, depending on the physical positions of the computing blocks constituting the 3D audio toolchain.

The main contributions of the paper are as follows:

- We identify the main hardware and software components of Immersive NMP (INMP) system;

- We identify the functional and performance requirements for INMP systems;
- We propose two 5G-based architectures for INMPs and describe the procedures involved by an INMP session;
- We present the performance metrics that can be adopted for the assessment of INMP systems;
- We offer a performance assessment via simulators along with design considerations following the achieved results.

As highlighted above, there have been integrations of NMP and 5G systems, as well as rather preliminary efforts in integrating immersive technologies with wired NMP systems. To the authors' best knowledge, the present study is the very first attempt to investigate how to integrate NMP systems with immersive technologies within wireless architectures. However, no existing system has combined yet these complementary aspects. This represents a significant advance towards the creation of enabling technologies not only for IoMusT applications, but also for those of the general Internet of Sounds field [31] and of remote collaborative virtual environments involving networked 3D audio [32]. The reported theoretical contributions in terms of key performance indicators, architectures and simulations, as well as the critical reflection on the achieved results, aim at providing designers of such emerging applications with concrete guidelines to follow to provide end-users with an optimal QoE

II. RELATED WORK

A. 3D AUDIO SYSTEMS

3D (or immersive) audio systems aim to deliver sounds surrounding a listener. The sounds are actually created by a sound diffusion system (constituted by headphones or a set of loudspeakers), but the listener's perception is that the sounds come from given points in space. Notably, for the case of NMPs, headphones are typically most used compared to a surround sound system, as they are typically more affordable and more practical (e.g., they occupy less space and allow to avoid issues of feedback loop with microphones external to the instrument). For this reasons, in the present study we focus only on the case of headphones.

To date, when leveraging headphones, the most accessible and widespread form of immersive audio is the binaural one. Binaural audio relies on the rendering of acoustic cues such as interaural time differences, interaural level differences, and acoustic filtering (i.e., the spectral information that depends on the specificities of the user's physical attributes such as the shape of ears, head, shoulders, torso) [33]. This rendering is achieved via head-related transfer functions (HRTFs), which are the acoustic transfer functions that encode the directionality of a sound source to the listener's eardrum. HRTFs are typically extrapolated from acoustic measurements [34] and organized into databases [35]. HRTFs may be personal or generic. The former ones relate to measures conducted on a specific individual, and are achieved using costly recording systems, as well as specialized facilities and hardware [36]. Because of the high cost, limited portability of the 3D recording system, and computation challenges, generic HRTFs are used at the cost of lower accuracy and a higher margin of error in sound localization. Generic HRTFs can be obtained through measurements on anthropomorphic mannequins or through binaural simulations of torso and head or by averaging a set of individual HRTFS for many subjects. Relevant examples of non-commercial and open-source binaural systems are the IEM Plug-in Suite [37], the 3D Tune-in Toolkit [38] and the Sparta & Compass [39].

A head-tracking system is typically utilized as an input to the immersive audio rendering algorithm. For the case of headphones, this is typically placed on the headphones themselves and generates data in the form of Euler angles or quaternions, which have a much lower sample rate (e.g., 10–200 Hz) than that of audio (e.g., 48.000 Hz). HRTFs are usually available for a discrete set of spatial positions. The HRTF to be used in a specific moment is dependent on the position of the head of the listener because the virtual location of the sound sources to be reproduced varies with the movement of the listener's head.

Together with the binaural spatialization of the virtual sound sources placed around the listener, it is possible to apply room modeling techniques to recreate the sensation that all such sources are produced from the same venue [19]. This may be achieved via ad hoc algorithms [20] or via binaural recordings of the impulse response of the venue (the latter may be performed via a dummy head with microphones in the ears or via dedicated sound field microphones). These techniques also aim at rendering the position of the sound source in the venue to be simulated. As a consequence, they produce for each sound source a specific set of impulse responses (2 or 4 channels) that are fed to the binaural algorithms, which can be interactively controlled via head trackers.

3D audio systems have been recently proven to confer the experience of playing together with immersiveness. The study reported in [40] simulated an NMP session enhanced with a 3D audio system that rendered the position of three connected musicians. Specifically, the system comprised ambisonics and head-tracking. Experiments were conducted to compare such a system with the simulation of a conventional NMP system that uses stereo diffusion and the mixing of all sound sources. Results provided evidence for musicians' preferences for spatialized listening during collaborative playing using headphones, as opposed to listening with classical stereophonic systems.

B. PACKET LOSS CONCEALMENT

Networking systems for audio-based interactions prioritize latency over reliability aiming at guaranteeing end-users with a fluid and uninterrupted user experience. For this purpose, such systems rely on best-effort protocols (such as UDP or RTP), which do not guarantee that all packets arrive at destination. At the receiver side these systems utilize queues of packets, i.e., jitter buffers, from which the received valid packets are constantly read. Packet losses occur when such queues become empty or when gaps in the bufferized data are generated by one or multiple lost packet. These gaps in the audio playout buffer at the receiver side can be caused by the unreliability of the network, traffic congestion, or uncompensated packet jitter.

Packet loss is a crucial problem of real-time audio streaming as it brings about detrimental effects on the QoE, such as audible artifacts. To cope with such issue, packet loss concealment (PLC) methods have been proposed, which gets invoked when the jitter buffer queue becomes empty or when one or multiple packets are missing. Such methods aim at reconstructing the content of the missing packets based on the content of the previous one [40], in order to mitigate the impact of audio gaps in the reproduced audio stream. The quality of PLC methods is typically assessed comparing the degree of similarity between the original and the reconstructed signal, leveraging objective and subjective metrics [41].

Nowadays PLC methods are integrated into the vast majority of audio codecs and are widely adopted in conventional audio streaming and videoconferencing systems. Unfortunately, traditional audio codecs that could offer PLC capabilities cannot be adopted in NMP scenarios, since the encoding/decoding process would significantly increase the E2E latency. In the context of hard real-time scenarios such as NMPs, PLC methods need to operate at zero delay in order to avoid introducing additional latency. The missing content may be synthesized using dedicated techniques, ranging from low-complexity Digital Signal Processing [17] to modern Deep Learning [42] and hybrid approaches [41].

It is worth noting that the context of INMP, which is the focus of the present study, entails the need for PLC methods to be applied not only to audio streaming, but also to the signals generated by the head tracking system. To the best of the authors' knowledge, a joint prediction of missing audio and head-tracking data portions in real-time streams has not been devised yet, which calls for dedicated investigation efforts.

C. TRAFFIC PREDICTION ALGORITHMS

Transmission of real-time multimedia is affected by the varying conditions of network traffic, as we frequently experience in videoconferencing applications where disconnections, stalls, and low-quality interruptions are very common, especially from domestic or wireless connections. The key necessity for quality improvement in the delivery of real-time audio streams is represented by the ability to adapt to the network conditions, ideally before variations happen. Therefore, the NMP backend infrastructure must integrate traffic prediction methods to operate at various time scales and proactively trigger adjustments in the audio streaming parameters with the aim of improving the QoS and thus the QoE perceived by the users. Traffic prediction is a widely investigated research topic and several supervised ML algorithms such as deep- and graph-based neural networks have proven to achieve high prediction accuracy at multiple time scales [43], [44], [45]

In our proposed architectures, we aim at exploiting traffic predictions not only to dynamically adapt streaming parameters but also to intelligently deploy and migrate VNFs hosting audio mixing/processing functionalities. Traffic prediction-based VNF migration has recently been investigated, e.g., in [46], though not in the context of NMP applications.

D. SDN AND 5G INFRASTRUCTURES FOR NMP

Currently, private and public deployments of 5G cellular networks are being rolled out worldwide. However, to date only a few designs of 5G infrastructures for NMPs have been investigated [29], along with a paucity of testbed deployments and in-depth statistical analysis on their latency and reliability performances [16]. The adoption of SDN technologies for NMP systems has been first envisioned in [47], which provides examples of possible interactions between a real-time network latency monitoring module and an NMP system.

In general, the categories of services identified for the 5G technology, [48], are known as Enhanced Mobile Broadband (EMBB), Massive Machine-Type Communications (MMTC), and Ultra-Reliable Low-Latency Communications (URLLC). Ideal NMPs requirements may be classified within the URLLC specifications, due to their consistency with the key requirements of URLLC specified by ITU in [49], i.e., i) 1 ms user plane latency from server to client or from client to server, in an ideal scenario; ii) 20 ms control plane latency; iii) reliability for one transmission of a packet close to $1-10^{-5}$ for 32 bytes with a user plane latency of 1 ms.

It is worth noticing that a key objective of the 5G PPP Phase 3 projects was to substantiate trials across various vertical industries. Despite various applications that have brought to definitions and measurements of 5G verticals KPIs [50], the NMPs scenario has not been considered and this represents a main lack to be fulfilled due to its unique requirements either in terms of constraints and variability of the geographical displacement of end users. Thus, the potential of 5G cellular systems in NMP contexts remains largely unexpressed.

III. COMPONENTS OF AN IMMERSIVE NETWORKED MUSIC PERFORMANCE SYSTEM

Fig. 1 depicts the main hardware and software component of an INMP system. It illustrates how each component contributes to the overall latency between a musician sending over the network a produced audio stream and a musician receiving it in spatialized form.

The total latency

$$\begin{aligned}\mathcal{L} = \lambda_{\text{ADC}} + \lambda_{\text{audio\_buffer}} \\ + \lambda_{\text{packetization}} + \lambda_{\text{network}} \\ + \lambda_{\text{jitter\_buffer}} + \lambda_{\text{depacketizazion}} \\ + \lambda_{\text{spatial\_audio}} + \lambda_{\text{DAC}} \quad (1)\end{aligned}$$

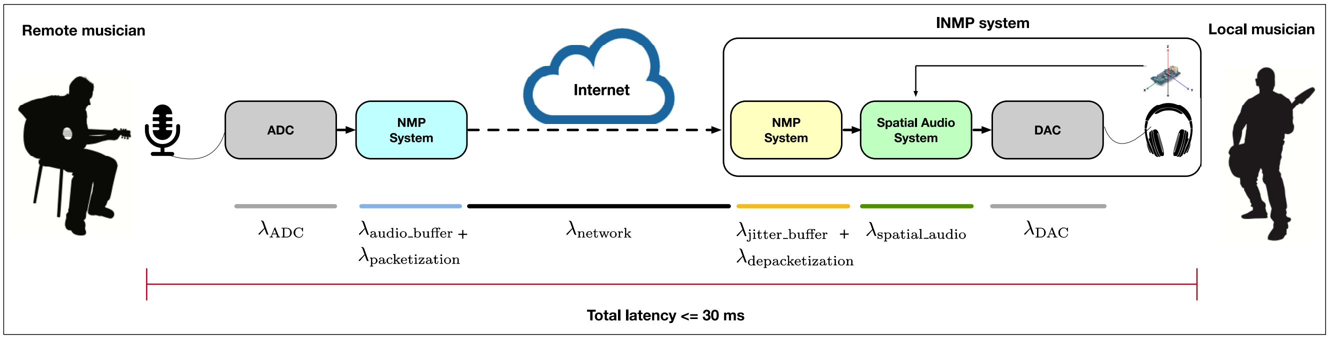


FIGURE 1. Diagram of the main hardware and software components contributing to the overall latency in an INMP.

where

- λ_{ADC} is the delay due to the acquisition of the signal to be sent (via an analog to digital converter);
- $\lambda_{\text{audio\_buffer}}$ represents the delay due to the acquisition of the digital signal in the audio buffer of the NMP system;
- $\lambda_{\text{packetization}}$ represents the delay due to the packetization of the digital signal via the NMP system;
- λ_{network} is the delay determined by the transport network latency;
- $\lambda_{\text{jitter\_buffer}}$ represents the delay caused by the jitter buffer used to compensate the network jitter for a sufficient number of packets, which relates to the buffer size;
- $\lambda_{\text{depacketization}}$ is the delay due to the received signal depacketization via the NMP system;
- $\lambda_{\text{spatial\_audio}}$ is the delay due to the spatial audio algorithm to generate a 3D rendering of the acoustic scene; this includes the delay introduced by the head-tracking system that feeds the head orientation to the spatial audio algorithm; this also includes the delay related to the mixing of the signals of the remote and local musicians;
- λ_{DAC} is the delay due to the delivery of the received signal (via a digital to analog converter).

According to the conventional spatial audio toolchain, the $\lambda_{\text{spatial\_audio}}$ delay can be further decomposed as follows (see Figure 2):

$$\begin{aligned}\lambda_{\text{spatial\_audio}} = \lambda_{\text{encoder}} \\ + \lambda_{\text{room\_simulation}} \\ + \lambda_{\text{decoder}}\end{aligned} \tag{2}$$

where

- λ_{encoder} is the delay taken by the binaural encoder;
- $\lambda_{\text{room\_simulation}}$ represents the delay due to the room simulation method;
- λ_{decoder} is the delay taken by the binaural decoder.

Fig. 2 also depicts $\lambda_{\text{sound\_scene\_rotation}}$, which refers to the delay introduced by the head-tracking system that feeds the head orientation to the spatial audio algorithm for the sound scene rotation. The study reported in [51] experimentally found that this motion-to-sound latency should be lower than 30 ms, although other studies suggested a higher threshold, up to about 50 ms [52].

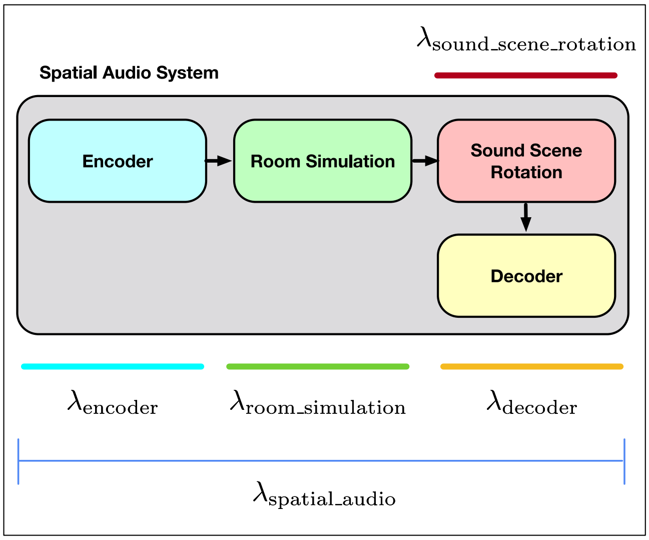


FIGURE 2. Schematic representation of the components of the spatial audio system contributing to latency.

However, in the context of INMPs the delay introduced by the head-tracking system does not sum up to the overall latency [53]. Firstly, the sound scene rotation algorithms work at zero processing latency. Secondly, the overall latency between two networked nodes is not affected by the motion-to-sound latency because the head-tracking control of the sound scene rotation algorithm acts in parallel to the rest of the spatial audio toolchain (i.e., encoder, room simulation, sound scene rotation, decoder).

IV. REQUIREMENTS FOR IMMERSIVE NETWORKED MUSIC PERFORMANCE SYSTEMS

In this section we analyze in detail the main requirements for the design of INMP systems, which can be summarized as follows:

1) strict latency requirements, in the order of at most 30 ms and with minimal variations (in order to ensure low jitter, that could be compensated with short buffers), to ensure realistic performative conditions;

2) integration of immersive audio rendering techniques, to increase the musicians' perception of sharing the same acoustic environment;
3) high reliability, to ensure high-fidelity audio playout, and large bitrate availability, to support the streaming of multiple audio channels;
4) significant computational capabilities for audio mixing and processing either at the users' premises or at VNFs available in the network backend infrastructure;
5) adequate throughput to support the service bandwidth demand.

We categorize requirements 1, 3 and 5 as performance requirements, while requirements 2 and 4 as functional requirements. The following subsections provide a thorough discussion of each of the above-listed requirements.

A. REQUIREMENT 1: LOW LATENCY AND RELIABLE COMMUNICATIONS

Focusing on latency constraints in NMP, several studies have determined that the maximum E2E latency that guarantees performative conditions to be as close as possible to traditional in-presence musical interactions amounts to 30 ms [1]. Moreover, a QoS vs latency tradeoff emerges due to a number of reasons: first of all, retransmission techniques such as those implemented by the Transmission Control Protocol (TCP) to ensure reliable data transfer cannot be exploited, as they would dramatically increase the E2E latency. Thus, unreliable connectionless protocols such as the User Datagram Protocol (UDP) are typically used in NMP systems. However, lost packets impact the quality of the audio playout by causing artifacts and glitches. The same negative impact occurs in the case of delayed packets, due for example to large jitter variations. To mitigate the effects of delayed packets, jitter buffers at the receiver side are needed: thus, the jitter buffer size not only has an impact on the overall latency (the larger the size, the higher the additional latency), but it may also affect reliability (the larger the size, the larger the compensation of jitter) [1], [16], [23]. Secondly, artifacts generated by missing packets cannot be concealed by exploiting traditional encoding/decoding schemes implemented in standard audio codecs, since their processing latency would further increase the overall latency. Thus, lightweight concealment methods that aim at reconstructing the missing signal without introducing additional latency are needed.

B. REQUIREMENT 2: IMMERSIVE AUDIO RENDERING

In NMP, the QoE perceived by remote players can be substantially improved by replicating at each remote stage the same auditory conditions as those experienced by other players, with the aim of providing a unified immersive 3D auditory perception. This is relevant not only for players but also for listeners. An immersive acoustic experience can be obtained by properly processing the sound sources for being spatialized through a high number of loudspeakers surrounding the listener, or through headphones [54]. Each solution has its own complexity, constraints, and drawbacks, and the NMP scenario is even more challenging since a communication link is placed between components of the audio toolchain. The advantages arising when using binaural audio techniques via headphones by exploiting edge computing on a 5G infrastructure have been discussed in [55], where the application scenario is related to cultural heritage. In the context of immersive NMP, a tradeoff emerges between QoE improvement versus the increased complexity of the overall system, thus pushing even further the need for low-latency communication.

C. REQUIREMENT 3: TAILORED PERFORMANCE METRICS FOR TRANSMISSION RELIABILITY

As far as network-layer packet transmissions are concerned, reliability is typically defined as Packet Error Ratio (PER), i.e., the percentage of the amount of sent packets that reach another system entity within the time constraint required by the targeted service, divided by the total number of sent packets. Concerning reliability in NMP, a commonly agreed threshold for PER has not been identified yet. Indeed, the relationship between amount packet losses, distribution of packet losses over time, and perceived audio quality has not been defined yet. Only a handful of studies have preliminary investigated such a complex matter [56], [57]. Nevertheless, the authors of [16] recently claimed that a potentially realistic PER requirement for NMP ranges from 10^{-6} up to 10^{-4}. On the other hand, the same authors also highlighted the fact that PER does not accurately reflect the requirements of NMP, where the distribution of lost audio information over time can have a significant impact on the audio quality perceived by the musician. Indeed, what also counts is the Maximum Number of Consecutive Lost Packets (MNCLP). For instance, considering 100 seconds of transmission, a 1% packet loss can describe a single burst of 1000 ms of lost audio, or 100 equally distant 10 ms-long audio gaps. A single 1000 ms burst will more likely affect the perceived audio quality than multiple 10 ms-long gaps. In particular, PLC methods may not be able to provide appropriate compensations for long bursts.

D. REQUIREMENT 4: ADEQUATE FUNCTIONAL AND COMPUTATIONAL CAPABILITIES

To enrich the musicians' perception of sharing the same acoustic environment, NMPs should include immersiveness of the audio experience, as suggested by the results reported in [40]. The three-dimensional representation and localization of audio sources entail a high number of audio channels to be streamed and mixed. However, current NMP systems are not equipped with a set of independent channels, one for each sound source representing a connected musician. Existing systems only provide a stereo mix of remotely connected musicians. For a binaural spatialization accounting for the rendering of the position of the connected musicians, it is necessary to provide at the receiver side the unmixed signals of each sound source. To enable such a scenario, it is necessary to advance the hardware and software components of NMP systems.

Moreover, to achieve optimal latency and bitrate conditions, the routing of audio streams should be dynamically adapted by jointly considering i) the physical location of the involved users and of the VNFs for low-latency audio mixing/processing and ii) the current network congestion level, by means of dedicated optimization algorithms. In turn, such algorithms need to leverage traffic predictions by ML algorithms, capable of estimating the future evolution of network load fluctuations. The execution of the above-mentioned algorithms requires the availability of adequate computational capabilities and integration with existing telemetry infrastructures.

E. REQUIREMENT 5: SUFFICIENT THROUGHPUT

Throughput, when combined with latency, jitter, and reliability, plays a pivotal role in determining whether a wireless technology can support a given use case. Thus, it is crucial to ensure that the involved communication link is capable of satisfying the bandwidth demand of an INMP system. The required throughput depends on a number of factors. Primarily the number of involved channels, i.e., the number of connected musicians, which relates to the number of audio streams that need to be communicated. Secondly, the rate and size of the packets. The former is dependent on the packet size, the periodicity at which packets are transmitted, and on the presence of retransmission mechanisms. The latter depends on the considered number of samples, sample rate, and bit depth, as well as on the presence of redundancy schemes. Thirdly, the geographical extension of the area of application, which may constrain the adoption of URLLC technologies.

V. PROPOSED ARCHITECTURES

This section describes our proposed IoMusT architectures for INMPs. These architectures are based on ETSI MEC model that relies on a low-latency backend infrastructure, leveraging the 5G mobile network for the access segment, as well as SDN and Service Orchestrator (SO) for network resources management and deployment of VNFs [58]. In addition, the architectures rely on optimization algorithms to minimize the E2E latency thanks to i) an intelligent placement of audio mixing and processing VNFs and ii) ML algorithms for traffic prediction and PLC. A pictorial representation of the backend infrastructure and of its functional building blocks is reported in Fig. 3.

A Service Management and Orchestration (SMO) platform is responsible to instantiate VNFs over the computing infrastructures and to allocate resources in the network segments involved to offer tailored performance according to E2E slice requirements. In particular, the deployment of VNFs will be realized by the SO while network resources will be managed by an SDN controller. Orchestration of VNFs involves not only INMP service components but also 5G core network components. In particular, the User Plane Function (UPF) core element permits to route user traffic toward the desired destinations without passing through the data centers of the legacy core network. This fully enables MEC capabilities in mobile networks. In turn, MEC servers may be allocated in different portions of the network, depending on the computational resource strategy, so that heterogeneous MEC systems could also be designed. This implies a multiple-tier site location architecture (e.g., cell level, metro network level, regional level), where MEC servers in different tiers show different computation and communication capabilities. For instance, a MEC node close to the end user will generically have low computational capabilities but will guarantee higher ubiquitous computing opportunities [59].

Figure 3 also shows different possibilities of MEC placement in the context of NMPs. Many different MEC nodes may be grouped into a MEC platform that is able to orchestrate the Network Service (NS) lifecycle. This calls for the deployed application to be an ETSI MEC Application, which means that the application needs to provide metrics to and from the MEC platform itself. This also enables application migration mechanisms in a transparent manner to the end users. A pair of players may indeed communicate through a MEC placed within the cell or through a MEC placed in the metro network or through a more remote connection obtainable with a MEC placed at the regional area. It is worth noting that this architecture does not take into consideration (for the sake of simplicity) the multi-operator case. However, the generalization can be easily considered taking into account higher values of network latencies and modelling accordingly the scenario into the proposed area. The multi-access part of MEC is meant to include multiple access networks directly connected to the MEC infrastructure. The area into which this connection happens determines if the use case can lay to one of the three considered areas [60].

For what concerns immersive audio rendering to be delivered via headphones, two different architectural solutions are proposed, depending on the physical positions of the computing blocks constituting the 3D audio toolchain. For both architectures, we assume that head tracking for all players is required for a proper immersive experience.

MEC-based IoMusT architecture. The first architecture considers a MEC server as well as a local device connected both to the headphones and the musical instrument (see Fig. 4). The local device (which could be either wearable or not) is based on an embedded system with sufficient computational power coupled with a hard real-time operating system specific to audio processing (e.g., Elk Audio OS [10]). Its task is that of digitizing the analog signal produced by the musical instrument as well as receiving the signals from the head-tracking. Subsequently, the device packetizes such data and delivers it to the chosen MEC server.

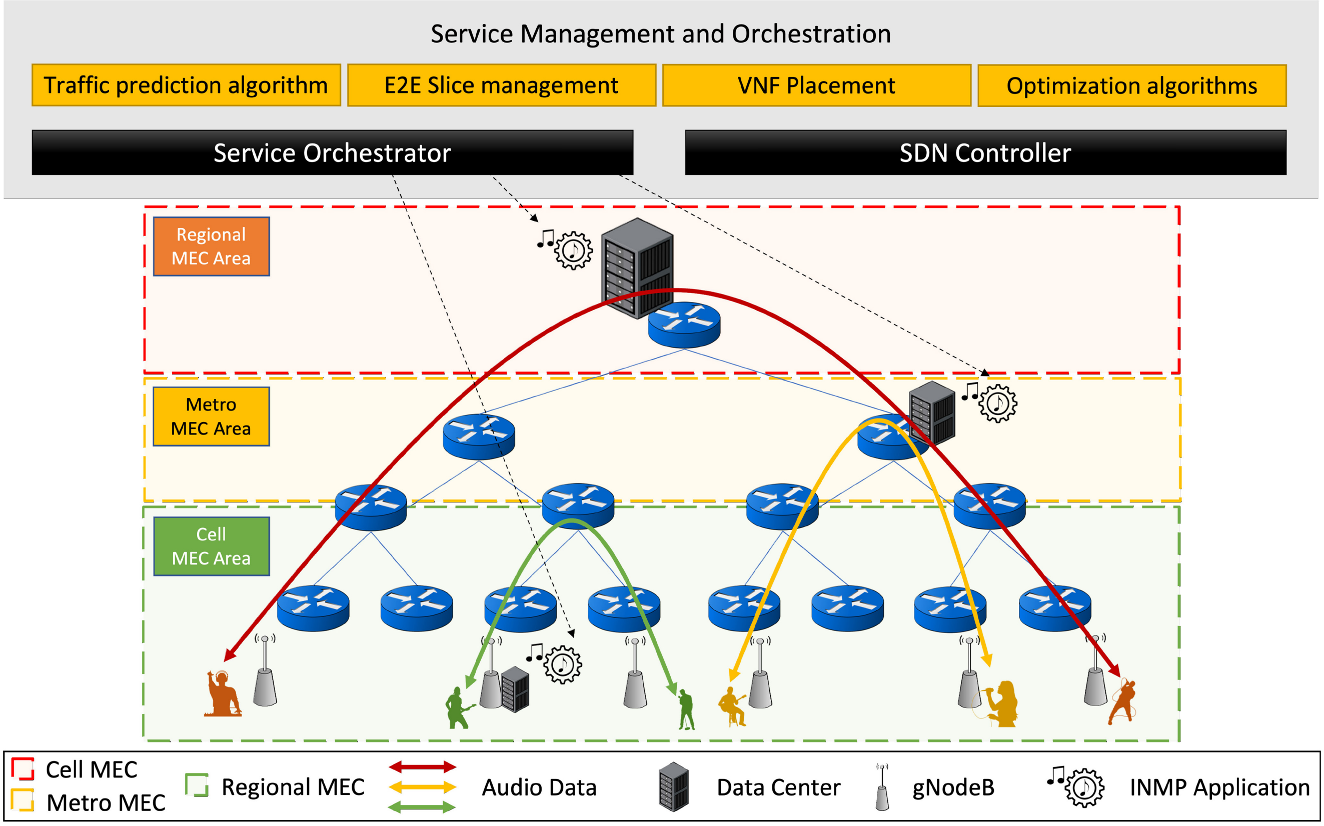


FIGURE 3. Overall architecture from the network layer point of view. Three representative use cases are presented. Pairs of musicians are connected to each other through: the same Cell MEC area (green); the same Metro MEC area (yellow); Regional MEC area (red).

The MEC server receives the audio streams from $k + 1$ players (the local player and the k remote players) of whom it knows the (fixed) position, as well as the current head position of the local player. Based on this data it computes the binaural audio stream including room modeling to be sent to the device of the local player (2 audio channels, one for each ear). The device then performs the depacketization and the digital-to-analog conversion of the audio signal to be delivered by the headphones. Notably, in our architectures we assume that the local player decides the fixed position of the other players and configures the binaural algorithm accordingly. Yet, we are aware that it is alternatively possible that the binaural algorithm receives the fixed or even dynamic position of the remote players from them.

The MEC server also performs the synchronization and then the mixing of the signals arriving from the musicians. This operation comes at a cost in terms of a variable delay: a mixing queue with a maximum but variable size (and thus a corresponding duration) is set in place which allows for the synchronization of the packets from the musicians. The mixed signal is delivered as soon as the data from all musicians (originated in the same time slot) arrive, if before the queue duration is elapsed. Otherwise, the server waits until the maximum duration of the queue and then mixes the available packets from that time slot (regardless of whether all of them have arrived or not), and then the resulting mixed audio data is delivered. This synchronization and mixing operation entails a variability in the latency, which however will be compensated by the jitter buffer at the receiver side (in the same way it occurs for the variability due to the wireless and wired links).

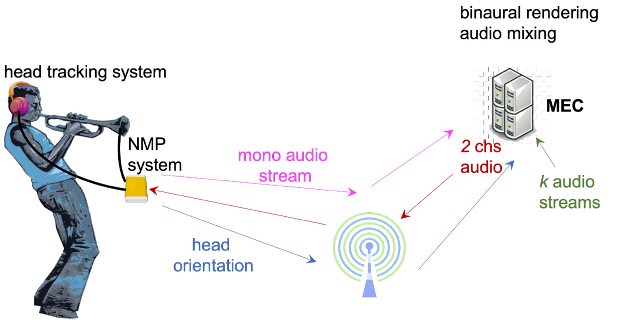


FIGURE 4. MEC-based IoMusT architecture: local user uplink and downlink signals exchange with the MEC server.

Embedded computing-based IoMusT architecture. In the second architecture the end user is equipped with a local device (connected to the headphones and the musical instrument), which is able to locally perform the computations of both the NMP system and the binaural audio system (see Fig. 5). As in the first architecture, the device may leverage an embedded system with sufficient computational power

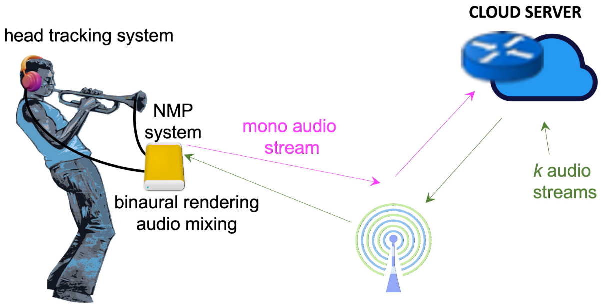


FIGURE 5. Embedded computing-based IoMusT architecture: local user uplink and downlink signals exchange with the network.

coupled with a hard real-time operating system specific for audio processing. In this architecture, the MEC server is not needed, since all computational tasks for immersive audio are assigned to the local device. Yet, a computational unit placed between musicians is needed to route the traffic (i.e., it acts as a relay server). Notably, the mixing queue involved in the MEC-based architecture is not necessary here, because at the receiver side there is already a jitter buffer. The jitter buffer has to account for the variability of the latency in the audio stream, which includes the variability introduced by the synchronization process. Thus, the delay introduced by the synchronization process does not sum up with the delay introduced by the jitter buffer.

A. SOFTWARE-BASED OPTIMIZATION OF LATENCY AND RELIABILITY

We envision the automated placement of VNFs with server instances for audio mixing and processing to operate as a backend infrastructure for the support of client/server-oriented audio streaming. The deployment of VNFs is achieved on-demand, possibly in the available resources of MEC nodes. The joint computation of optimal placement of 5G network functions and INMP service instances and their deployment is operated by the SMO platform. This allows for minimizing the latency by locating VNFs in strategic positions on the network infrastructure based on the users' physical location, bitrate, and latency requirements. It also allows for adapting to the current traffic congestion level along the path interconnecting users to servers. The SMO collects service monitoring information from the users currently participating in the musical session, as well as from measurement nodes that may be deployed within the network, and adjusts audio streaming parameters. Locations and routing of audio flows are dynamically updated via the SMO as traffic conditions evolve, based on the latency and packet losses experienced by the users.

The SMO also dynamically adjusts 5G radio resources based on channel and traffic conditions, exploiting traffic prediction methods based on ML to operate at various time scales. In turn, dedicated optimization algorithms allow for the reduction of network latency by exploiting the outputs of such traffic predictors, which enable them to operate proactively and to dynamically adjust VNFs' locations depending on the current and forecasted network congestion conditions, thus ultimately improving the QoE perceived by the users.

Moreover, ML approaches for audio PLC as well as head-tracking data PLC that are able to operate in real-time without introducing additional processing latency are applied. ML-based PLC methods are implemented in the mixing servers, possibly located at MEC sites, and/or in the end-users' devices.[1]

B. THE MULTI-DOMAIN CASE

The previously described architecture at the network layer does not specifically tackle the intricacies of multi-domain environments. This oversight is rectified in Figure 6, where we explore the dynamics of a multi-domain scenario. In this context, the problem of Internet Exchange Point (IXP) placing is crucial in terms of bandwidth and latency offered to the end users. IXPs, which enable direct interconnectivity between Internet Service Providers (ISPs) and facilitate efficient data exchange, offer direct interconnection as an alternative to routing through one or more third-party networks through the Internet. Significant advantages in terms of cost, latency, and bandwidth are achievable exploiting IPX.

The landscape of multi-domain configurations is diverse, with IXPs positioned across various geographic regions to meet their connectivity needs. While the contemporary network architecture often features regional-level IXPs, these can introduce latency and increase the risk of packet loss. Notably, in broader network arrangements, IXPs may be situated at more localized levels, potentially mitigating these connectivity challenges.

For analytical simplicity and without compromising the general applicability of our findings, our discussion is anchored to the scenario presented in Figure 3. In this scenario, we assume that the integration of a MEC facility within a specific geographical area can effectively approximate a multi-domain environment, provided that an IXP is situated at the same operational level as the MEC facility, thus providing low latency communication between multiple ISPs.

C. COMPARISON WITH SOA ARCHITECTURES FOR NMP

For the sake of completeness, it must be noted that conducting a comprehensive comparison with a baseline architecture is not feasible for the specific application under consideration, for two reasons. The first one is the novelty introduced in NMPs through the integration of immersive audio. To the best of the authors' knowledge, this is the first work discussing a wireless network architecture for such a framework.

The second motivation relates to an in-depth consideration of state of the art works on similar topics. Indeed [23] focuses

[1] A detailed description of the specific PLC mechanisms to be implemented is beyond the scope of this paper and thus omitted.

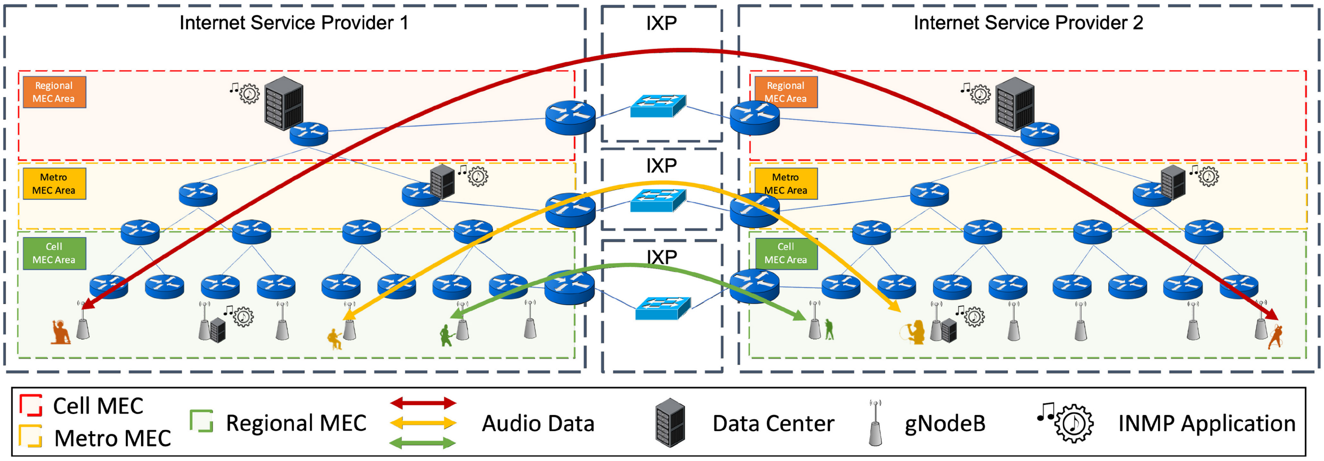


FIGURE 6. Representative use-cases for a multi-domain scenario.

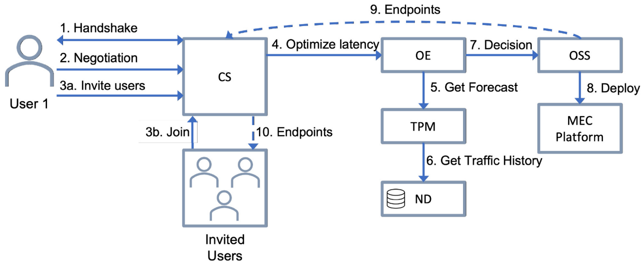


FIGURE 7. Communication Diagram of the Connection Setup procedure.

on a private, stand alone (SA), MEC-enabled network, where all users perform inference within the same local area. In contrast, [16] presents a scenario operating on a public SA network without accounting for the benefits arising from a MEC solution. Finally [29] refers to a 5G network with MEC exploitation but without considering multiple MEC servers displacements.

VI. PROCEDURAL EXAMPLES

In the following, we present the procedures involved by the three main phases that constitute an NMP session, namely *Connection Setup*, *Media Transmission* and *Connection Teardown*, discussing the logical flows of operations and the data exchanges among the various entities of our envisioned architectures. The procedural examples for MEC setup are based on [61], extending the application lifecycle procedures defined by ETSI in the context of our NMP architecture.[2]

1) Connection Setup:The first phase involves a user initiating an NMP session by instantiating a control channel with a remote NMP Control Server (CS) (assumed to be located in the cloud) which will set up the required NS composed by a set of VNFs. Required steps are shown in Fig. 7 and described as follows:

1) the user initiates a TCP connection handshake to set up a reliable control channel with the CS. This channel will handle parameter negotiation, signaling, user location, etc.;
2) the user negotiates media streaming parameters with the CS, such as audio sample rate, bit depths of audio samples, number of supported audio channels, type and number of requested VNFs (e.g., audio mixing, processing to add effects such as reverb and equalization, packet loss concealment, etc.);
3) the user sends invitations to other users via the CS to join the initiated session or waits for other participants. Note that the CS is assumed to maintain visibility of the availability status of each user and to display it to other users when necessary;
4) once all the participants have joined the NMP session, the CS queries the Optimization Engine (OE) sharing users' locations;
5) the aim of the OE is to minimize the end-to-end latency among users. To do this, the OE identifies a number of candidate MEC nodes, computes a number of candidate paths interconnecting each user with every candidate node, and then queries the Traffic Prediction Module (TPM) to provide traffic forecasts on network congestion along each candidate path, at various time scales (e.g., one minute, ten minutes, one hour, etc)[3];
6) the TPM queries the Network Database (ND) to retrieve historical traffic data collected among the network links traversed by the candidate paths. Traffic forecasts are created and forwarded to the OE;
7) the OE executes its internal routines, outputs the selected MEC node and VNFs, and provides its output to the Operations Support System (OSS);
8) the OSS takes charge of the procedure to deploy the VNF chain on the chosen MEC nodes;

[2] We acknowledge that the routines reported in this section have not yet undergone a formal verification procedure. However, the procedural correctness of the underlying communication protocols, i.e., TCP and UDP, has already been mathematically proved [62], [63]

[3] The detailed description of the optimization algorithms implemented by the OE and the prediction models adopted by the TPM is beyond the scope of this paper and left for future work.

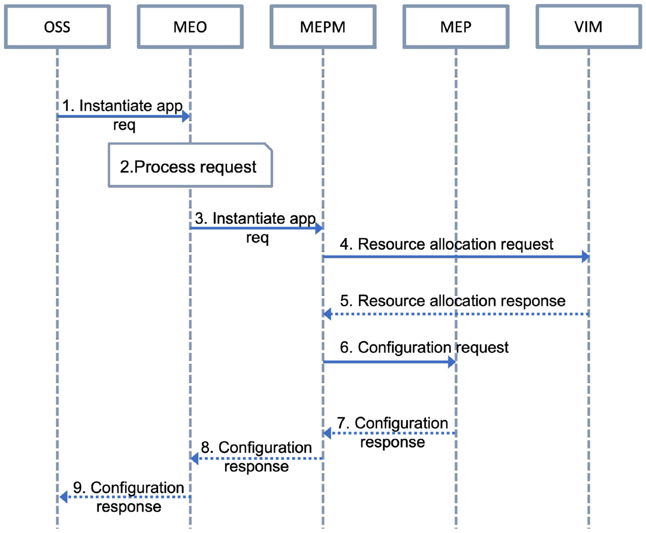

OSS
MEO
MEPM
MEP
VIM
1. Instantiate app req
2.Process request
3. Instantiate app req
4. Resource allocation request
5. Resource allocation response
6. Configuration request
7. Configuration response
8. Configuration response
9. Configuration response

FIGURE 8. Sequence Diagram of the VNF migration procedure handled by the SMO system.

9) the OSS informs the CS about the created endpoints
10) the CS forwards endpoint information to the end users that can connect to the instantiated VNF chain.

2) Media Transmission: Once the Connection Setup phase is concluded, the Media Transmission phase begins, during which a bidirectional UDP stream is instantiated between each user and the NS, carrying the negotiated audio streams. During the NMP session, connection statistics measured by the users and/or the VNF chain (such as average/minimum/maximum latency, packet loss percentage, and packet loss burstiness) are regularly collected by the CS. Moreover, on regular intervals (e.g., in the order of tens of seconds), the following routine is triggered:

1) the CS queries the OE to decide if the instantiated VNF chain should be migrated. The query includes the locations of the users and VNFs involved in the session and aggregated transmission statistics;
2) on input of the query from the CS, the OE queries the TPM to provide traffic forecasts along the paths between each user and the VNFs nodes, at various time scales;
3) the TPM queries the ND to retrieve traffic measurements collected during the last time interval along the network links traversed by the involved paths, produces traffic forecasts, and provides them to the OE;
4) on input of the requested traffic forecasts, the OE executes its internal routines and outputs its decision. If a migration of one or multiple VNFs in the chain is deemed necessary, it provides the location of the new MEC node(s) where the VNF(s) has/have to be migrated by the OSS;
5) if no migration is required, the routine ends. Otherwise, a VNF migration subroutine is triggered.

The migration subroutine is responsible to move the VNF(s) required to provide the NS from the source MEC node to the destination MEC node lowering the E2E latency. Many architectural components of the MEC system are involved to migrate the VNF. The OSS holds control of the overall system and takes decisions based on operational requirements. The MEC Orchestrator (MEO) is the Information Expert of available resources in each edge node and is delegated to allow or forbid the Virtual Infrastructure Manager (VIM) to reserve resources for a specific VNF. The Mobile Edge Platform Manager (MEPM) cooperates with the Virtualized Infrastructure Manager (VIM) to instantiate the virtualized platform supporting the VNF. The SMO layer is responsible to coordinate the management and orchestration of the overall process and it is composed of the VIM, the VNF Manager (VNFM), and NFV Orchestrator (NFVO). The NFVO is responsible for managing the NS lifecycle, the Resource Orchestration procedures, and the NS Orchestration (NSO). The VNF migration subroutine, showed in figure 8, operates as follows:

1) the OSS receives an operational requirement change by the OE and starts an *Instantiate app request*;
2) the MEO processes the request, verifies that enough resources are available, and forwards the *Instantiate app request* to the MEPM;
3) the MEPM sends to the VIM a *Resource allocation request* to preempt required physical resources;
4) the VIM sends back to the MEPM a *Resource allocation response* to inform the VIM of the allocation status;
5) the MEPM can now configure the MEP sending a *configuration request*;
6) the MEP replies with a *Configuration response* sent to the MEPM. The message is forwarded back to the OSS to inform it about the new configuration status;
7) the procedure may be repeated for each VNF composing the NS to be migrated;
8) the OSS asks the E2E Slice Management module to provide a network slice to serve the newly created VNF or VNF chain;
9) the OSS informs the CS about the modified endpoints.

It is worth noting that the presented procedure involves a joint orchestration of the MEC and cloud resources to optimize the overall perceived user experience.

3) Connection Teardown:Every time a user wants to leave the session, the following procedure is applied:

1) the bidirectional UDP stream between the user and NS is torn down and the user notifies the CS about its leave. If no other active users remain, the CS tears down the NS and releases the resources allocated in the hosting MEC node, otherwise the following steps are executed;
2) the CS notifies the remaining participants that the user has abandoned the NMP session;
3) steps 1-5 of the Media Transmission phase are executed to verify if the VNF chain should be migrated, since new MEC locations offering better QoS/QoE to the remaining participants may exist.

VII. KEY PERFORMANCE INDICATORS FOR NUMERICAL ASSESSMENT

In this Section, we present the performance metrics that can be adopted for the assessment of the proposed INMP system.

A. BITRATE

For each architecture, it is possible to characterize the bitrate of the communication (in terms of bits per second) as follows. Let's consider $\mathcal{B}_{\text{uplink}}$ and $\mathcal{B}_{\text{downlink}}$ respectively as the uplink/downlink bitrate along the wireless channel between the instrument and the 5G base station (including the MEC); $\mathcal{B}_{\text{upstream}}$ and $\mathcal{B}_{\text{downstream}}$ as the upstream and downstream bitrate over the wired network channel connecting two base stations respectively serving 2 players (i.e., assuming that only one player is connected to the same base station); and k as the number of remote players (the total number of players is $k+1$ considering also the local player).

For the MEC-based architecture:

$$\begin{aligned}\mathcal{B}_{\text{uplink}} &= \beta_{\text{audio,local\_player (1 channel)}} + \beta_{\text{head\_tracking}}\\ \mathcal{B}_{\text{upstream}} &= \beta_{\text{audio,local\_player (1 channel)}}\\ \mathcal{B}_{\text{downlink}} &= \beta_{\text{audio,3D\_spatialization (2 channels)}}\\ \mathcal{B}_{\text{downstream}} &= \beta_{\text{audio,remote\_players (k channels)}}\end{aligned} \tag{3}$$

For the embedded computing-based architecture:

$$\begin{aligned}\mathcal{B}_{\text{uplink}} &= \mathcal{B}_{\text{upstream}} = \beta_{\text{audio,local\_player (1 channel)}}\\ \mathcal{B}_{\text{downlink}} &= \mathcal{B}_{\text{downstream}} = \beta_{\text{audio,remote\_players (k channels)}}\end{aligned} \tag{4}$$

where:

- $\beta_{\text{audio,local\_player (1 channel)}}$ is the bitrate of the audio signal (1 channel) generated by the musical instrument;
- $\beta_{\text{head\_tracking}}$ is the bitrate of the signal generated by the head tracking system;
- $\beta_{\text{audio,3D\_spatialization (2 channels)}}$ is the bitrate of the 2 audio channels generated by the binaural audio system (1 channel per ear);
- $\beta_{\text{audio,remote\_players (k channels)}}$ is the bitrate of the audio signals generated by the k remote players (1 channel per player);

In the following, we provide some numerical estimates for the quantities described above, considering state-of-the-art components and the most common tuning of their parameters. The considered NMP system is Elk LIVE, which is based on the Elk Audio OS (a low-latency audio operating system optimized for embedded systems [10]) and an ad-hoc hardware device that digitalizes analog audio signals and packetizes them prior to transmission onto the telecommunication network. The system enables deterministic processing for high-precision packet pacing, where audio packets are periodically transmitted according to a given rate, which is defined by the sampling frequency and the block size (i.e., the number of audio samples over which any processing is performed): $packet\_rate = block\_size/sampling\_frequency$. The device works with a sampling frequency of 48 kHz and a block size of 64 samples per channel where each sample has a bit-depth of 16 bits (i.e., 2 bytes). Therefore, the packet transmission rate is one packet every $64/(48 \cdot 10^3) \approx 1.34$ ms. To optimize for latency, the UDP is utilized for transport, without including any retransmission scheme at the application layer. The UDP packetization introduces a header of typically 8 bytes. The audio packet data unit (APDU) is given by $APDU = (block\_size \cdot bit\_depth) + UDP\_header$. Given the parameters above, for a single channel the APDU $=$ (64 samples $\cdot$ 2 bytes) $+$ 8 bytes $=$ 136 bytes. To compute the bitrate, we need to compute how many packet transmissions are performed in one second and then multiply this number by the APDU. Such a number is given by 1 $second/packet\_rate$, which according to the values above is 1000 ms / 1.34 ms $\approx$ 750. Therefore, $\beta_{\text{audio,local player (1 channel)}} \approx 750 \cdot 136 \approx$ 102.000 bytes per second.

Concerning the head-tracking data, we can consider a commercial head-tracking system able to provide quaternions (in the form of 4 floats, where each float is one byte) every 1 ms. The head-tracking packet data unit (HTPDU) is given by the 4 floats plus the UDP header, i.e., HTPDU $=$ (4 floats $\cdot$ 4 bytes) $+$ 8 bytes $=$ 24 bytes. The number of packets per second is 1000, thus $\beta_{\text{head-tracking}} = 1000 \cdot 24$ bytes $=$ 24.000 bytes per second. Therefore, for the MEC-based architecture $\mathcal{B}_{\text{uplink}} \approx (102.000 + 24.000) \cdot 8 \approx 1.008$ Mbit per second. Utilizing similar computations for the other two IoMusT architectures, and considering the realistic scenario of a total of 4 connected musicians (thus $k = 3$), it is possible to determine the other components contributing to the bitrate as shown in Table 1.

TABLE 1. Bitrate estimates (in Mbit per second), for each component of the two architectures, considering a total of 4 musicians playing together.

Bitrate component	MEC	Embedded
$\mathcal{B}_{\text{uplink}}$	1.008	0.816
$\mathcal{B}_{\text{upstream}}$	0.816	0.816
$\mathcal{B}_{\text{downstream}}$	2.448	2.448
$\mathcal{B}_{\text{downlink}}$	1.632	2.448

B. LATENCY

For each architecture, we characterize the end-to-end latency of the communication as the time elapsed from the moment when the audio and head tracking signals are generated by the sender until their reproduction at the remote musician's side. It is worth noting that overloading and priority policies for MEC servers are out of the scope of this work and are thus not considered in the overall latency assessment.

For the MEC-based architecture:

$$\begin{aligned}\mathcal{L}_{\text{MEC}} = \tau_{\text{audio,ADC,packetization}}&\\ + \tau_{\text{uplink}} + \tau_{\text{upstream}} + \tau_{\text{MEC,processing}}&\\ + \tau_{\text{downlink}} + \tau_{\text{audio,DAC,depacketization}}&\end{aligned} \tag{5}$$

For the embedded computing-based architecture:

$$\mathcal{L}_{\text{Embedded}} = \tau_{\text{audio,ADC,packetization}} + \tau_{\text{uplink}} + \tau_{\text{upstream}} + \tau_{\text{downlink}} + \tau_{\text{audio,DAC,depacketization}} + \tau_{\text{embedded,processing}} \tag{6}$$

where:

- $\tau_{\text{audio,ADC,packetization}}$ is the time taken by the device to perform the analog-to-digital conversion of the mono audio signal from the musical instrument as well as to create the UDP packets before passing them to the wireless transmission module;
- τ_{uplink} includes the processing delay at the wireless transmission module, the transmission time over the wireless link, and the processing time at the base station side;
- τ_{upstream} is the delay component caused by the transmission of the packetized data from the base station that serves the transmitting device towards the base station that serves the receiving device;
- $\tau_{\text{MEC,processing}}$ is the time taken by the MEC server to process the incoming service request (i.e., binaural rendering, room modeling, synchronization and mixing);
- τ_{downlink} is the counterpart of τ_{uplink} and includes the processing time at the base station side, the transmission time over the wireless link, and the processing delay at the wireless transmission module. Note that, due to the different direction of the transmission (downlink vs. uplink), it is likely that $\tau_{\text{downlink}} \neq \tau_{\text{uplink}}$;
- $\tau_{\text{audio,DAC,depacketization}}$ is the time taken by the computing unit to perform the depacketization of the UDP packets as well as the digital-to-analog conversion of the audio signals from the MEC (for the MEC-based architecture) or from the network (for the embedded computing architecture). Note that this delay also includes the time taken by the jitter buffer.
- $\tau_{\text{embedded,processing}}$ is the time taken by the embedded device to process the incoming service request (i.e., binaural rendering, room modeling, and mixing);

In Table 2 we provide some numerical estimates for the latency components described above. These estimates have been derived from the experiments reported in [23], which involved a session of 4 musicians using the Elk Live NMP system (configured with a jitter buffer of 10.66 ms and packetization and de-packetization taking 1.33 ms, and where the ADC/DAC conversions took 0.5 ms). The estimates also consider the data reported in [53] for the measurement of the fastest spatial audio system (0.33 ms). For the mixing queue at the MEC side, an average delay of 3 ms was considered. Note that a generic time of 7 ms has been added to account for the wired network component, so to stay below the 30 ms threshold. Note also that the downlink time for the MEC-based architecture is lower than that of the embedded computing-based architecture because in the former 2 channels are transmitted, in the latter 4.

TABLE 2. Latency estimates (in ms), for each component of the two architectures, considering a total of 4 musicians playing together.

Latency component	MEC	Embedded
$\tau_{\text{audio,ADC,packetization}}$	1.83	1.83
τ_{uplink}	2.6	2.6
τ_{upstream}	7	7
$\tau_{\text{MEC,processing}}$	3.33	-
τ_{downlink}	2.8	2.9
$\tau_{\text{audio,DAC,depacketization}}$	12.43	12.43
$\tau_{\text{embedded,processing}}$	-	0.33

It is worth noticing that the overall latency computations above are not affected by the motion-to-sound latency introduced by the head-tracking system for the sound scene rotation, as the nature of these two sources of delay is different (as discussed in Section II-A). Nevertheless, the motion-to-sound latency ($M2S$) differs for the two architectures and is as follows.

For the MEC-based architecture:

$$M2S_{\text{MEC}} = \tau_{\text{ht}} + \tau_{\text{ht\_packetization}} + \tau_{\text{uplink}} + \tau_{\text{MEC,processing}} + \tau_{\text{downlink}} + \tau_{\text{audio,DAC,depacketization}} \tag{7}$$

For the embedded computing-based architecture:

$$M2S_{\text{Embedded}} = \tau_{\text{ht}} \tag{8}$$

where

- τ_{ht} is the time taken by the head-tracking device to read the motion of the head and produce in output the digital values (e.g., quaternions);
- $\tau_{\text{ht\_packetization}}$ is the time taken to create the UDP packets before passing them to the wireless transmission module;
- τ_{uplink} as above;
- $\tau_{\text{MEC,processing}}$ as above;
- τ_{downlink} as above;
- $\tau_{\text{audio,DAC,depacketization}}$ as above.

As stated in Section II-A, it is necessary that $M2S_{\text{MEC}}$ and $M2S_{\text{Embedded}}$ are below 30 ms to avoid perceivable discrepancies between the listener's head movements and the resulting spatialized sound [51]. According to the datasheet of commercial and open-hardware head-trackers, the τ_{ht} can be as low as 10 ms, and in some cases even reaches 5 ms.

C. RELIABILITY

As far as network-layer packet transmissions are concerned, reliability is typically defined as packet error ratio (PER), i.e., the percentage of the amount of sent packets that reach another system entity within the time constraint required by the targeted service, divided by the total number of sent packets. We consider this as a measure of reliability in absence of a more comprehensive metric that would also encompass the maximum number of lost packets, as discussed in Section IV. Similarly to what we did with latency, we propose to decompose the E2E reliability into multiple reliability components as follows.

For the MEC-based architecture:

$$\begin{aligned}\mathcal{R}_{\text{MEC}} = p_{\text{audio (1 channel),succ,uplink}} \\ \cdot p_{\text{audio (1 channel),succ,transport,upstream}} \\ \cdot p_{\text{audio (2 channels),succ,downlink}} \\ \cdot p_{\text{ht,succ,uplink}}\end{aligned} \tag{9}$$

For the embedded computing-based architecture:

$$\begin{aligned}\mathcal{R}_{\text{Embedded}} = p_{\text{audio (1 channel),succ,uplink}} \\ \cdot p_{\text{audio (1 channel),succ,transport,upstream}} \\ \cdot p_{\text{audio (k channels),succ,downlink}}\end{aligned} \tag{10}$$

where:

- $p_{\text{audio (1 channel),succ,uplink}}$ is the success probability of the uplink transmission for both the 1-channel audio data generated by the transmitting musician;
- $p_{\text{ht,succ,uplink}}$ is the success probability of the uplink transmission of the head-tracking data generated by the receiving musician;
- $p_{\text{audio (1 channel),succ,transport,upstream}}$ is the success probability of the packet forwarding across the transport network from the base station that serves the transmitting device towards the base station that serves the receiving device;
- $p_{\text{audio (2 channels),succ,downlink}}$ is the success probability of the downlink transmission of the 2-channel audio data generated by the MEC as a result of the 3D rendering, mixing, and room modeling;
- $p_{\text{audio (k channels),succ,downlink}}$ is the success probability of the downlink transmission of the k-channel audio data generated by the remote musicians;

Note that we did not consider any packet losses or processing errors at the device and MEC server side, i.e., the reliability calculation takes into account only the contributions of the wireless and wired network transmission components.

Table 3 reports the reliability estimates (in terms of packet error ratio), for each component of the two architectures, considering a total of 4 musicians playing together. Such estimates have been based on the measurements reported in [23]. For the wired network contribution we considered a generic packet loss ratio of 0.002 which is reasonable if considering an ad-hoc network with reserved resources (e.g., inter-universities networks).

VIII. SERVICE COVERAGE CHARACTERIZATION

The extent of INMP service coverage is reliant on various factors such as the type of scenario and network deployment conditions, which include the range of coverage for the 5G system and the location and instantaneous load of MEC servers. As a result, the assessment of service coverage presents an intriguing open research question. To address this gap, we conducted system-level simulations using ns-3 software together with the 5G-LENA module [64] that enabled 5G communications. ns-3 is a well-known, open-source simulation engine that is supported by the community and continually maintained. By employing this software, we were able to obtain valuable results without deploying actual devices.

5G-LENA is the next iteration of LENA, a module initially developed for 4G communications to implement radio access and core networks. It implements the fundamental 5G PHY-MAC features in accordance with NR specifications. Our analysis evaluates whether 5G networks are able to support the scenarios presented in Section V while introducing acceptable latency, which is the essential KPI for an effective INMP service. Ultimately, our aim is to validate the proposed scenarios in the context of low-latency INMP.

TABLE 3. Reliability estimates (in terms of packet error ratio), for each component of the two architectures, considering a total of 4 musicians playing together.

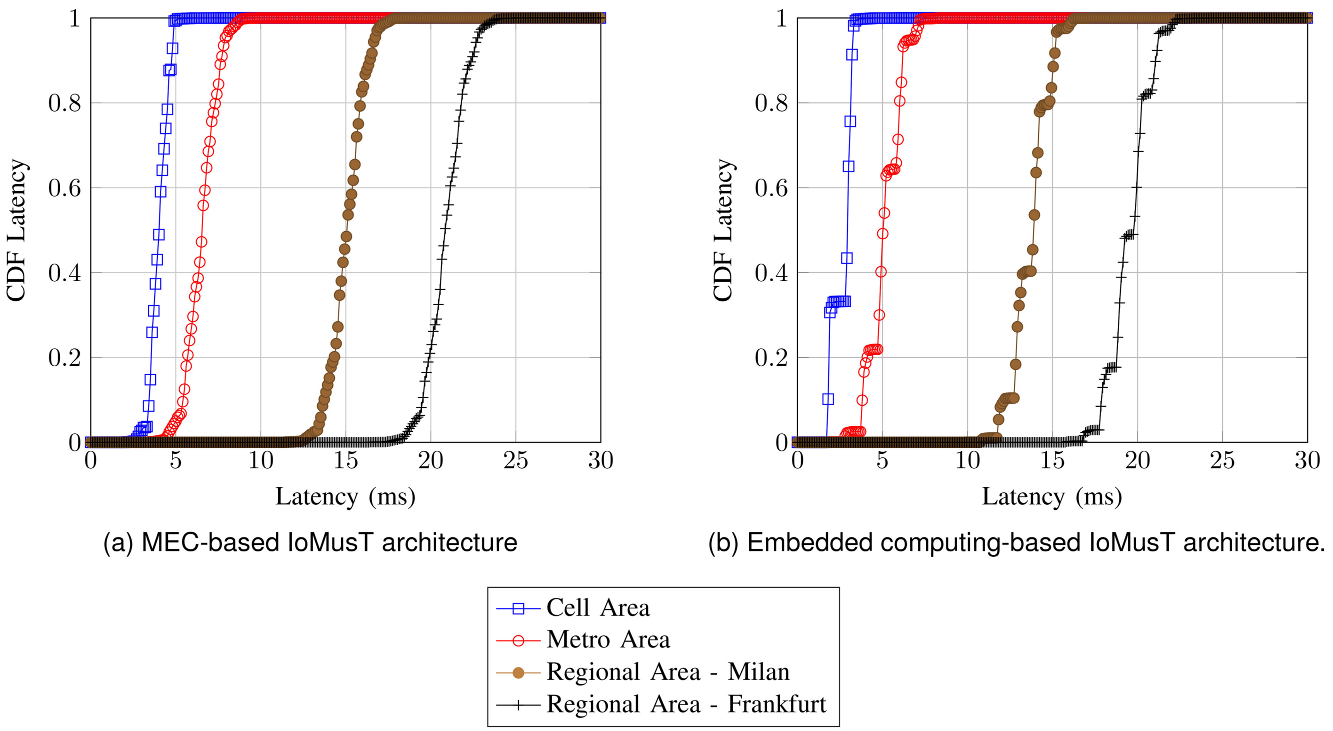
Reliability component	MEC	Embedded
$p_{\text{audio (1 channel),succ,uplink}}$	0.0055	0.0055
$p_{\text{audio (1 channel),succ,transport,upstream}}$	0.002	0.002
$p_{\text{audio (2 channels),succ,downlink}}$	0.0055	-
$p_{\text{ht,succ,uplink}}$	0.003	-
$p_{\text{audio (4 channels),succ,downlink}}$	-	0.006

TABLE 4. Simulation parameters (UDP is utilized as transmission protocol).

	Parameter	Value
Traffic model	Number of participants	6
	Packet size (audio)	136 Bytes
	Packet size (head tracking)	24 Bytes
	Packet rate (audio)	750 packets / s
	Packet rate (head tracking)	1000 packets / s
Channel model	Numerology	2
	Shadowing	Enabled
	Type of Scenario	3GPP TR 38.901 (UMa)
	Central Frequency	2035 MHz
	UL and DL Bandwidth	20 MHz
	Error Model	NrEesmIrT1
Deployment	Max UE to gNB distance	150 m
	UE Height	1.5 m
	gNB Height	25 m

A. SIMULATION SETUP

The two architectures presented in Section V have been implemented as two separate network simulation scripts inside ns-3. The two scenarios mainly differ in how the packets are routed from the sender to the receiver. Most simulation parameters are shared between the two scenarios,



(a) MEC-based IoMusT architecture

(b) Embedded computing-based IoMusT architecture.

(c) Legend

FIGURE 9. Simulations results for the MEC-based (a) and the embedded computing-based (b) architectures showing the Cumulative Density Function (CDF) of the end-to-end latency. Each CDF is generated by considering all the samples from 60 realizations that use distinct seeds for the random functions.

and they are shown in Table 4. In both scenarios, the positioning of the UEs inside the cell happens by randomly drawing the *x* and *y* coordinates from a uniform distribution and then imposing the distance from the base station to be less than 150m. Changing the seed of the random functions shuffles the positions of all the UEs.

- *MEC-based IoMusT architecture:* there are many MEC servers (one per user) so each packet is sent to the MEC server closest to the sender, which forwards it to each other MEC server. Every MEC server then, for each time slot, waits to receive the packets from all the remote users as well as the head tracking information from the local user. When all the packets of a time slot are available to the MEC server, their content is mixed and audio spatialization is applied. If some packets of a time slot are not available 10 ms after the reception of the first one, the MEC server proceeds to mix and spatialize the content of the available ones. The resulting stereo audio is then sent to the local user.
- *Embedded computing-based IoMusT architecture:* there is just one cloud server that serves all the users. Each packet is sent to the cloud server that forwards it directly to all the other users (i.e., it acts as a relay server). The users perform audio mixing and spatialization locally so the waiting phase, in this case, happens after the network communication is completed.

The output of these network simulations is constituted by the end-to-end latency of each packet. We can expect the MEC-based architecture to have slightly worse latency because the waiting phase that leads to the mixing and spatialization of sound occurs before the end of the packet's network travel. In the embedded computing-based architecture, in fact, that waiting phase happens after the packets are received by the users, so outside of the networking operations.

B. SIMULATION RESULTS

The results of the simulations are shown in Fig. 9a and 9b for scenarios 1 and 2 respectively. The figures show the Cumulative Density Function of the end-to-end latency measured in the simulations. For each scenario, the three sub-scenarios depicted in Fig. 4 were considered:

- *Cell area:* in this case, the gNBs of the users are directly connected with no added delay in between, simulating communication between users who are very close to each other.
- *Metro area:* in this case, a delay was introduced between the gNBs to simulate a metropolitan area where the users are still in the same city but no longer close to each other.
- *Regional area:* here a delay was introduced between the gNBs to simulate users located at a much greater distance than the metro area. Multiple plots pertaining to the regional area can be seen in Figure 9. This is because multiple distances were simulated for the regional area.

TABLE 5. Latencies distributions.

Source/Destination	Distribution parameters (ms)
Milan/Milan	$\mathcal{N} \sim (3, 0.7)$
L'Aquila/Milan	$\mathcal{N} \sim (11.5, 0.8)$
L'Aquila/Frankfurt	$\mathcal{N} \sim (17.3, 0.94)$

Real-world measurements were taken over the public Internet to infer realistic mean and standard deviation of the network latency over specific paths. Latency measurements were evaluated using a series of virtual machines leased from Amazon Web Services. These machines were located in various cities. For each city, latency was measured a thousand times at various times of the day, to ensure statistical reliability. The measured parameters were then used to define the normal distributions which actual delay values were drawn from to simulate the metro and regional areas. In particular, the delay in the metro area is based on measurements taken from a household in Milan to a data center in Milan. The delays in the regional areas are all taken between a household in L'Aquila and different locations around the world, as specified in Table 5. Each CDF is generated by considering all the samples from 60 simulation runs, each with a different seed for the randomization functions, resulting in different positions of the users around the gNBs.

By inspecting Fig. 9a, it emerges that the worst-case latency of the cell area is below 7 ms, while for the metro area it is below 10 ms. The regional area is below 20 ms for Milan, but for the Frankfurt case it is above 20 ms and barely below 25 ms. Comparing Fig. 9a and Fig. 9b we can notice that the latency of the MEC-based architecture is slightly higher overall, as expected from the discussion in Section VIII-A. The smoothing of the curves in Fig. 9a is also attributed to the same reason.

Finally, it is worth noting that the plot for the cell area in Fig. 9b provides insights into the $M2S_{\text{MEC}}$ latency, because it is essentially showing the sum of the uplink and downlink latencies. So it is interesting to see that the networking component of $M2S_{\text{MEC}}$ is less than 5 ms. Since the threshold for the motion-to-sound latency amounts to 30 ms according to [51], there is room for around 25 ms to account for the delay introduced by the head-tracker and the spatialization algorithm.

IX. DISCUSSION AND CONCLUSION

This paper proposed two architectures for INMPs based on a backend leveraging SDN methods and on the orchestration, slicing, and MEC capabilities of 5G. Moreover, the architectures leverage ML algorithms for network traffic prediction and audio PLC.

From the numerical estimates and simulations reported in Sections VII and VIII it is possible to conclude that the two architectures have specific advantages and disadvantages. The MEC-based architecture leads to a greater amount of latency compared to the embedded computing-based architecture, and also involves the transmission of the head-tracker data over the 5G link, which is an additional source of potential packet losses. The higher latency has implications on the choice of the size of the jitter buffer, which is related to the reliability: the MEC-based architecture entails the reduction of the size of the jitter buffer to achieve a same latency of the embedded computing-based architecture, but at the cost of a likely lower reliability. In turn, larger packet losses due to shorter jitter buffers entail the use of more efficient (and computationally-demanding) packet loss concealment methods. On the other hand, while the embedded-computing based architecture costs less in terms of latency, and allows for longer jitter buffers (thus increasing the reliability), it has the drawback of delegating all the computations to the embedded system. This might not be able to cope with the required computational load, especially if the spatial audio algorithms involved are computationally complex.

QoS and QoE are crucial for INMP, as the E2E communication of audio data through the network requires very low latency, low jitter, and high audio quality (i.e., low packet losses that generate imperceptible dropouts in the signal). The proposed architectures represent a shift of paradigm as they envision the adoption of 5G to interconnect a backend infrastructure capable of accommodating the unique needs of INMP applications in terms of QoS and QoE. Indeed, 5G offers an unprecedented level of flexibility to fulfill service-specific requirements in terms of bitrate, latency, and reliability, achieved through novel techniques for radio transmission and novel architectural approaches, such as SDN and NFV. The above mentioned solutions, together with MEC paradigm exploitation, enable a deeper level of network slicing able to deliver better network KPIs so enhancing the users' QoE. Addressing the trade-off between QoS and QoE entails progressing our understanding on how musicians interact remotely in immersive audio settings, which includes conducting psychoacoustic research on the definition of commonly agreed reliability metrics as well as on the validation of novel PLC methods.

Notably, for INMP to exist it is paramount that current NMP systems are improved with the provision of the unmixed signals of each sound source at the receiver side. At present, NMP systems solely offer the receiver with the stereo mix of the sound signals coming from the remotely connected users. Such an improvement necessarily entails the advancement of the hardware and software components comprised in an NMP system.

It is worth noticing that our study presents some limitations. First, in this paper, we focused on the case of sound delivery via headphones since it is the most widespread and easy-to-deploy case among musicians compared to the use of a simple or complex surround sound system. Nevertheless, the architectures can be easily adapted to the case of a surround sound system composed of a given number of loudspeakers. Of course, this would entail the computation of the signals to be fed to each loudspeaker and in the case of the MEC-based architecture also of their downlink transmission. Second, we considered that the virtual position of the musicians was fixed for the whole duration of the musical session. This is the most common scenario, where musicians in an NMP do not significantly move on stage. Nevertheless, the architectures could be easily adapted to encompass also the scenario in which the position of the remote musicians changes dynamically and is streamed in real-time, for instance through the exploitation of a body tracking device (e.g., Azure Kinect DK) properly placed for indoor localization.

In future work we plan to implement the proposed architecture as well as assess them across QoS and QoE metrics. Finally, this work highlights the need of integrating spatial audio systems in NMP systems, as well as of progressing the development of both spatial audio algorithms and head tracking systems for the minimization of their latency contributions.

REFERENCES

[1] C. Rottondi, C. Chafe, C. Allocchio, and A. Sarti, "An overview on networked music performance technologies," *IEEE Access*, vol. 4, pp. 8823–8843, 2016.

[2] L. Comanducci, "Intelligent networked music performance experiences," in *Special Topics in Information Technology*. Cham, Switzerland: Springer, 2023, pp. 119–130.

[3] L. Comanducci et al., "Investigating networked music performances in pedagogical scenarios for the intermusic project," in *Proc. IEEE Conf. Open Innov. Assoc. (FRUCT)*, 2020, pp. 119–127.

[4] K. E. Onderdijk et al., "Livestream experiments: The role of COVID-19, agency, presence, and social context in facilitating social connectedness," *Front. Psychol.*, vol. 12, May 2021, Art. no. 647929.

[5] L. Turchet, C. Fischione, G. Essl, D. Keller, and M. Barthet, "Internet of Musical Things: Vision and challenges," *IEEE Access*, vol. 6, pp. 61994–62017, 2018.

[6] J. Cáceres and C. Chafe, "JackTrip: Under the hood of an engine for network audio," *J. New Music Res.*, vol. 39, no. 3, pp. 183–187, 2010.

[7] C. Drioli, C. Allocchio, and N. Buso, "Networked performances and natural interaction via LOLA: Low latency high quality A/V streaming system," in *Proc. Int. Conf. Inf. Technol. Perform. Arts, Media Access, Entertain.*, 2013, pp. 240–250.

[8] A. Carôt, C. Hoene, H. Busse, and C. Kuhr, "Results of the fast-music project—Five contributions to the domain of distributed music," *IEEE Access*, vol. 8, pp. 47925–47951, 2020.

[9] C. Werner and R. Kraneis, "UNISON: A novel system for ultra-low latency audio streaming over the Internet," in *Proc. IEEE 18th Annu. Consum. Commun. Netw. Conf.*, 2021, pp. 1–4.

[10] L. Turchet and C. Fischione, "Elk audio OS: An open source operating system for the Internet of Musical Things," *ACM Trans. Internet Things*, vol. 2, no. 2, pp. 1–18, 2021.

[11] X. Jiang et al., "Low-latency networking: Where latency lurks and how to tame it," *Proc. IEEE*, vol. 107, no. 2, pp. 280–306, Feb. 2019.

[12] R. Moscatelli, K. Stahel, R. Kraneis, and C. Werner, "Why real-time matters: Performance evaluation of recent ultra-low latency audio communication systems," in *Proc. IEEE 21st Consum. Commun. Netw. Conf.*, 2024, pp. 77–83.

[13] R. Hupke, D. Jan, N. Werner, and J. Peissig, "Latency and quality-of-experience analysis of a networked music performance framework for realistic interaction," in *Audio Engineering Society Convention*. New York, NY, USA: Audio Eng. Soc., 2022, pp. 1–10.

[14] K. Tsioutas and G. Xylomenos, "Assessing the effects of delay to NMP via audio analysis," *SN Comput. Sci.*, vol. 4, no. 2, pp. 1–13, 2023.

[15] C. Bartlette, D. Headlam, M. Bocko, and G. Velikic, "Effect of network latency on interactive musical performance," *Music Percept.*, vol. 24, no. 1, pp. 49–62, 2006.

[16] J. Dürre et al., "In-depth latency and reliability analysis of a networked music performance over public 5G infrastructure," in *Audio Engineering Society Convention*. New York, NY, USA: Audio Eng. Soc., 2022.

[17] M. Sacchetto, Y. Huang, A. Bianco, and C. Rottondi, "Using autoregressive models for real-time packet loss concealment in networked music performance applications," in *Proc. Int. Conf. Audio Mostly*, 2022, pp. 203–210.

[18] C. S. Oh, J. N. Bailenson, and G. F. Welch, "A systematic review of social presence: Definition, antecedents, and implications," *Front. Robot. AI*, vol. 5, p. 114, Oct. 2018.

[19] J. Paterson and H. Lee, *3D Audio*. Abingdon, U.K.: Routledge, 2021.

[20] L. Savioja and U. P. Svensson, "Overview of geometrical room acoustic modeling techniques," *J. Acoust. Soc. Amer.*, vol. 138, no. 2, pp. 708–730, 2015.

[21] P. Cairns et al., "Evaluation of metaverse music performance with BBC Maida Vale recording studios," *J. Audio Eng. Soc.*, vol. 71, no. 6, pp. 313–325, 2023.

[22] L. Gabrielli and S. Squartini, *Wireless Networked Music Performance*. Singapore: Springer, 2016.

[23] L. Turchet and P. Casari, "Latency and reliability analysis of a 5G-enabled Internet of Musical Things system," *IEEE Internet Things J.*, vol. 11, no. 1, pp. 1228–1240, Jan. 2024.

[24] A. Carôt, M. Dohler, S. Saunders, F. Sardis, R. Cornock, and N. Uniyal, "The world's first interactive 5G music concert: Professional quality networked music over a commodity network infrastructure," in *Proc. Sound Music Comput. Conf.*, 2020, pp. 407–412.

[25] L. Vignati et al., "Is music in the air? Evaluating 4G and 5G support for the Internet of Musical Things," *IEEE Access*, vol. 12, pp. 38081–38101, 2024.

[26] Z. Shu and T. Taleb, "A novel QoS framework for network slicing in 5G and beyond networks based on SDN and NFV," *IEEE Netw.*, vol. 34, no. 3, pp. 256–263, May/Jun. 2020.

[27] A. A. Barakabitze, A. Ahmad, R. Mijumbi, and A. Hines, "5G network slicing using SDN and NFV: A survey of taxonomy, architectures and future challenges," *Comput. Netw.*, vol. 167, Feb. 2020, Art. no. 106984.

[28] F. Cheli and S. Giordano, "Service parameters identification for adaptive networked music performance," in *Proc. Global Inf. Infrastruct. Netw. Symp.*, 2022, pp. 94–98.

[29] M. Centenaro, P. Casari, and L. Turchet, "Towards a 5G communication architecture for the Internet of Musical Things," in *Proc. IEEE Conf. Open Innov. Assoc. (FRUCT)*, 2020, pp. 38–45.

[30] D. A. Tedjopurnomo, Z. Bao, B. Zheng, F. M. Choudhury, and A. K. Qin, "A survey on modern deep neural network for traffic prediction: Trends, methods and challenges," *IEEE Trans. Knowl. Data Eng.*, vol. 34, no. 4, pp. 1544–1561, Apr. 2022.

[31] L. Turchet et al., "The Internet of Sounds: Convergent trends, insights and future directions," *IEEE Internet Things J.*, vol. 10, no. 13, pp. 11264–11292, Jul. 2023.

[32] J.-M. Jot, R. Audfray, M. Hertensteiner, and B. Schmidt, "Rendering spatial sound for interoperable experiences in the audio metaverse," in *Proc. Immers. 3D Audio, Archit. Automot. (I3DA)*, 2021, pp. 1–15.

[33] H. Møller, M. F. Sørensen, D. Hammershøi, and C. B. Jensen, "Head-related transfer functions of human subjects," *J. Audio Eng. Soc.*, vol. 43, no. 5, pp. 300–321, 1995.

[34] P. Majdak, P. Balazs, and B. Laback, "Multiple exponential sweep method for fast measurement of head-related transfer functions," *J. Audio Eng. Soc.*, vol. 55, no. 7/8, pp. 623–637, 2007.

[35] V. R. Algazi, R. O. Duda, D. M. Thompson, and C. Avendano, "The cipic HRTF database," in *Proc. IEEE Workshop Appl. Signal Process. Audio Acoust.*, 2001, pp. 99–102.

[36] N. Gupta, A. Barreto, M. Joshi, and J. C. Agudelo, "HRTF database at FIU DSP lab," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2010, pp. 169–172.

[37] M. Frank, F. Zotter, and A. Sontacchi, "Producing 3D audio in ambisonics," in *Proc. Audio Eng. Soc. Conf.*, 2015, pp. 1–8.

[38] M. Cuevas-Rodríguez et al., "3D Tune-In Toolkit: An open-source library for real-time binaural spatialisation," *PloS One*, vol. 14, no. 3, 2019, Art. no. e0211899.

[39] L. McCormack and A. Politis, “SPARTA & COMPASS: Real-time implementations of linear and parametric spatial audio reproduction and processing methods,” in *Proc. Audio Eng. Soc. Conf.*, 2019, pp. 1–13.

[40] M. Tomasetti and L. Turchet, “Playing with others using headphones: musicians prefer binaural audio with head tracking over stereo,” *IEEE Trans. Human-Mach. Syst.*, vol. 53, no. 3, pp. 501–511, Jun. 2023.

[41] A. I. Mezza, M. Amerena, A. Bernardini, and A. Sarti, “Hybrid packet loss concealment for real-time networked music applications,” *IEEE Open J. Signal Process.*, vol. 5, pp. 266–273, 2024.

[42] P. Verma, A. Mezza, C. Chafe, and C. Rottondi, “A deep learning approach for low-latency packet loss concealment of audio signals in networked music performance applications,” in *Proc. 27th Conf. Open Innov. Assoc. (FRUCT)*, 2020, pp. 268–275.

[43] Y. Li and C. Shahabi, “A brief overview of machine learning methods for short-term traffic forecasting and future directions,” *Sigspatial Spec.*, vol. 10, no. 1, pp. 3–9, 2018.

[44] W. Jiang and J. Luo, “Graph neural network for traffic forecasting: A survey,” *Expert Syst. Appl.*, vol. 207, Nov. 2022, Art. no. 117921.

[45] M. Abbasi, A. Shahraki, and A. Taherkordi, “Deep learning for network traffic monitoring and analysis (NTMA): A survey,” *Comput. Commun.*, vol. 170, pp. 19–41, Mar. 2021.

[46] F. Carpio, W. Bziuk, and A. Jukan, “Scaling migrations and replications of virtual network functions based on network traffic forecasting,” *Comput. Netw.*, vol. 203, Feb. 2022, Art. no. 108582.

[47] E. Lakiotakis, C. Liaskos, and X. Dimitropoulos, “Improving networked music performance systems using application-network collaboration,” *Concurr. Comput., Pract. Exp.*, vol. 31, no. 24, 2019, Art. no. e4730.

[48] “5G white paper,” NGMN Alliance, Frankfurt, Germany, White Paper, 2015.

[49] *Minimum Requirements Related to Technical Performance for IMT-2020 Radio Interface(s)*, ITU-Rec. M.2410-0, Int. Telecommun. Union, Geneva, Switzerland, 2017.

[50] K. Trichias, E. Kosmatos, I. Mesogiti, C. de Majo, and M. Giuffrida. “5G PPP trials results 2022—Key Performance Indicators measured in advanced 5G trial sites.” Jun. 2023. [Online]. Available: https://doi.org/10.5281/zenodo.7961946

[51] D. S. Brungart, B. D. Simpson, and A. J. Kordik, “The detectability of headtracker latency in virtual audio displays,” in *Proc. Int. Conf. Audit. Disp.*, 2005, pp. 37–42.

[52] A. Lindau, “The perception of system latency in dynamic binaural synthesis,” in *Proc. 35th DAGA Int. Conf. Acoust.*, 2009, pp. 1063–1066.

[53] L. Turchet and M. Tomasetti, “Immersive networked music performance systems: identifying latency factor,” in *Proc. Int. Conf. Immers. 3D Audio, Archit. Automot.*, 2023, pp. 1–6.

[54] W. Zhang, P. N. Samarasinghe, H. Chen, and T. D. Abhayapala, “Surround by sound: A review of spatial audio recording and reproduction,” *Appl. Sci.*, vol. 7, no. 5, p. 532, 2017.

[55] C. Rinaldi, F. Franchi, A. Marotta, F. Graziosi, and C. Centofanti, “On the exploitation of 5G multi-access edge computing for spatial audio in cultural heritage applications,” *IEEE Access*, vol. 9, pp. 155197–155206, 2021.

[56] A. F. Khalifeh, A.-K. Al-Tamimi, and K. A. Darabkh, “Perceptual evaluation of audio quality under lossy networks,” in *Proc. Int. Conf. Wireless Commun., Signal Process. Netw. (WiSPNET)*, 2017, pp. 939–943.

[57] M. Fink, M. Holters, and U. Zölzer, “Comparison of various predictors for audio extrapolation,” in *Proc. Int. Conf. Digit. Audio Eff.*, 2013, pp. 1–7.

[58] “Multi-access edge computing (MEC): Framework and reference architecture,” Eur. Telecommun. Stand. Inst., Sophia Antipolis, France, document GS MEC 003, Mar. 2022.

[59] Y. Mao, C. You, J. Zhang, K. Huang, and K. B. Letaief, “Mobile edge computing: Survey and research outlook,” 2017, *arXiv:1701.01090*.

[60] “Multi-access edge computing (MEC); Study on inter-MEC systems and MEC-Cloud systems coordination,” Eur. Telecommun. Stand. Inst., Sophia Antipolis, France, document GR MEC 035, Jun. 2021.

[61] “Multi-access edge computing (MEC); MEC management; Part 2: Application lifecycle, rules and requirements management,” Eur. Telecommun. Stand. Inst., Sophia Antipolis, France, document GS MEC 010-2, Jun. 2023.

[62] S. Bishop, M. Fairbairn, M. Norrish, P. Sewell, M. Smith, and K. Wansbrough, “Rigorous specification and conformance testing techniques for network protocols, as applied to TCP, UDP, and sockets,” in *Proc. Conf. Appl., Technol., Archit., Protocols Comput. Commun.*, 2005, pp. 265–276.

[63] G. Cluzel, K. Georgiou, Y. Moy, and C. Zeller, “Layered formal verification of a TCP stack,” in *Proc. IEEE Secure Develop. Conf. (SecDev)*, 2021, pp. 86–93.

[64] N. Patriciello, S. Lagen, B. Bojovic, and L. Giupponi, “An E2E simulator for 5G NR networks,” *Simul. Model. Pract. Theory*, vol. 96, Nov. 2019, Art. no. 101933.

LUCA TURCHET (Senior Member, IEEE) received the master’s degree (summa cum laude) in computer science from the University of Verona in 2006, the degrees in classical guitar and composition from the Music Conservatory of Verona in 2007 and 2009, respectively, received the Ph.D. degree in media technology from Aalborg University Copenhagen in 2013, and the degree in electronic music from the Royal College of Music of Stockholm in 2015. He is an Associate Professor with the Department of Information Engineering and Computer Science, University of Trento, Italy. His scientific, artistic, and entrepreneurial research has been supported by numerous grants from different funding agencies, including the European Commission, the European Institute of Innovation and Technology, the European Space Agency, the Italian Ministry of Foreign Affairs, and the Danish Research Council. He is the Co-Founder of the Music-Tech Company Elk Audio. He is the Chair of the IEEE Emerging Technology Initiative on the Internet of Sounds and the Founding President of the Internet of Sounds Research Network. He serves as an Associate Editor for IEEE ACCESS and the JOURNAL OF THE AUDIO ENGINEERING SOCIETY, and has been a Guest Editor for the IEEE COMMUNICATIONS MAGAZINE, the *Personal and Ubiquitous Computing*, the *Journal of the Audio Engineering Society*, *Frontiers in VR*, and *Digital Creativity*.

CLAUDIA RINALDI (Member, IEEE) received the bachelor’s degree in electronic music from the Conservatory of Music of L’Aquila, the Laurea degree (cum laude) in electronic engineering from the University of L’Aquila, Italy, in 2005, the master’s degree in trumpet from the Conservatory of Music of L’Aquila, and the Ph.D. degree in electronic engineering from the University of L’Aquila in 2009. She is a Researcher with the National Inter-University Consortium for Telecommunications (CNIT) and an Adjunct Professor with the Department of Information Engineering, Computer Science and Mathematics, University of L’Aquila. Her main research activities are focused on digital signal processing algorithms and in general on the use of technology in artistic fields. She is also concerned with design, modeling and optimization of communication algorithms with particular emphasis on the physical layer and software defined radio for the development of transmission systems responding to cognitive radios paradigms.

CARLO CENTOFANTI (Member, IEEE) received the M.Sc. degree in computer science engineering from the University of L’Aquila, where he is currently pursuing the Ph.D. degree in information and communication technology with the Department of Information Engineering, Computer Science and Mathematics. He actively contributed to ECSEL-RIA “AfarCloud” project designing and defining the system’s Software Architecture. He is involved in the EU MSCA “OPTIMIST” Project. He won several competition notices for fixed-term research fellow positions from the University of L’Aquila.

LUCA VIGNATI received the B.Sc. degree in computer and electronic engineering from the University of Pavia, Italy, and the M.Sc. degree in computer science and engineering from the Politecnico di Milano, Italy. He is currently pursuing the Ph.D. degree with the Department of Information Engineering and Computer Science, University of Trento. His main research interests include the Internet of Musical Things, audio over 5G networks, and embedded systems specific for audio applications.

CRISTINA ROTTONDI (Senior Member, IEEE) received the bachelor's and master's degrees (cum laude) in telecommunications engineering and the Ph.D. degree in information engineering from the Politecnico di Milano in 2008, 2010, and 2014 respectively. She is an Associate Professor with the Department of Electronics and Telecommunications with the Politecnico di Torino. From 2015 to 2018, she had a research appointment with the Dalle Molle Institute for Artificial Intelligence, Lugano, Switzerland. She is the co-author of more than 100 scientific publications in international journals and conferences. Her research interests include optical networks planning and networked music performance. She served as an Associate Editor for IEEE ACCESS from 2016 to 2020 and is currently an Associate Editor of the IEEE/OSA JOURNAL OF OPTICAL COMMUNICATIONS AND NETWORKING. She is co-recipient of the 2020 Charles Kao Award, of the Three Best Paper Awards in FRUCT-IWIS 2020, DRCN 2017, and GreenCom 2014, and of the One Excellent Paper Award in ICUFN2017.

Open Access funding provided by 'Università degli Studi di Trento' within the CRUI CARE Agreement