# Experimental Evaluation of Open-Source-based VoNR and OTT Voice Services in 5G SA Network Deployments

Elena-Ramona Modroiu, Joyce Mwangama, Marius Corici, and Thomas Magedanz

Original scientific article

***Abstract*—The Fifth Generation (5G) mobile network was developed to provide greater capacity, lower latency, and higher throughput, thereby enhancing multimedia service delivery. In the context of private 5G networks, open-source software is a key enabler of innovation and widespread adoption. This study introduces the first open-source prototype implementation of Voice over New Radio (VoNR) and evaluates its performance in two 5G Standalone (SA) private network deployment models: distributed and all-in-one (”in-a-box”). Leveraging an experimental setup composed entirely of open-source tools, the paper offers a comparative analysis of VoNR and Over-the-Top (OTT) voice services, examining their respective advantages and limitations. The findings contribute to the optimization of voice and multimedia services in next-generation (6G) networks and underscore the role of open-source platforms in achieving scalable, high-quality voice integration.**

***Index Terms*—VoNR, VoLTE, N5, IMS, 5G Stand-Alone, campus networks, KPI, Open Source.**

## I. Introduction

INTEGRATING voice services into 5G networks remains a significant challenge. Since 4G/Long Term Evolution (LTE), multimedia services, including voice, have been delivered through the IP Multimedia Subsystem (IMS), which integrates Voice over IP (VoIP) technology with Quality of Service (QoS) mechanisms. IMS-based technologies are also expected to underpin multimedia service delivery in future 6G networks [1].

As mobile networks evolve from 4G to 5G Non-Standalone (NSA) and ultimately to 5G Standalone (SA), mobile network operators often rely on fallback to Voice over LTE (VoLTE) for voice service integration. However, this approach interrupts ongoing data sessions when voice calls are initiated, as the user equipment (UE) switches from 5G to LTE. Additionally, inter-network switching has a detrimental effect on battery life. Ideally, voice calls should be handled natively over the 5G network using VoNR to fully leverage 5G’s enhanced performance capabilities.

Although VoLTE and VoNR technologies both rely on the same IMS framework, their interfaces to the 5G Core network are specified differently by 3GPP. VoNR employs the N5 interface [2], which is based on a RESTful API specification, whereas VoLTE uses the Diameter protocol [3] to communicate the QoS requirements of multimedia sessions to the mobile core network.

Similarly, existing literature on multimedia voice services in 5G networks has primarily focused on 5G NSA deployments, with an emphasis on data services and reliance on proprietary solutions for voice integration. Limited research has explored the integration of VoNR in 5G SA networks using open-source tools, and even fewer studies have evaluated its performance across diverse deployment scenarios.

This paper makes a significant contribution to the integration of voice services into 5G SA networks using open-source tools:

1) We present the first open-source VoNR prototype, deployed within a 5G SA testbed at Technische Universität Berlin. By leveraging open-source tools, the solution addresses key aspects such as cost-effectiveness, flexibility, and innovation potential.
2) The prototype was demonstrated live, showcasing its feasibility and real-world applicability. [1]
3) We provide a detailed performance evaluation of the VoNR prototype across both distributed and all-in-one (”in-a-box”) private 5G deployment scenarios:
   - While the initial conference paper [4] offered early insights, it did not fully address questions regarding the performance and scalability of VoNR across deployment types. This extended study offers a more comprehensive evaluation, introducing a new experimental setup: the private 5G in-a-box architecture.
   - We compare VoNR with OTT voice solutions developed using the same open-source platforms, delivering a holistic analysis of their relative advantages and limitations in both deployment configurations. This extended evaluation contributes to a better understanding of multimedia service

Manuscript received February 4, 2025; revised March 14, 2025. Date of publication May 12, 2025. Date of current version May 12, 2025.

E.-R. Modroiu is with the Technische Universität Berlin, Germany (e-mail: elena-ramona.modroiu@tu-berlin.de).

J. Mwangama is with the University of Cape Town, South Africa (e-mail: joyce.mwangama@uct.ac.za).

M. Corici and T. Magedanz are with the Technische Universität Berlin, Germany and Fokus Fraunhofer Institute, Berlin, Germany (e-mails: marius-iulian.corici@fokus.fraunhofer.de, thomas.magedanz@tu-berlin.de).

The paper was presented in part at the International Conference on Software, Telecommunications and Computer Networks (SoftCOM) 2024.

Digital Object Identifier (DOI): 10.24138/jcomss-2025-0007

[1] www.open6gnet.org/osmnt2024-conference/

integration strategies in the context of future 6G networks.

The remainder of this paper is structured as follows: Section II reviews related work and highlights the limitations of existing solutions. Section III details the architecture of the open-source VoNR prototype and the experimental setup. Section IV presents the performance evaluation and comparative analysis. Finally, Section V discusses broader implications and outlines directions for future research.

## II. Related Work

Mobile network technologies have evolved steadily over the past decades, with each successive generation introduced approximately every ten years, bringing substantial improvements in key performance indicators (KPIs) such as reduced latency, increased throughput, enhanced capacity, and improved user experience.

Advancements in technologies such as Software-Defined Networking (SDN) and Network Function Virtualization (NFV) [5] have transformed mobile networks. Traditionally, network functions were tightly coupled with dedicated hardware, limiting scalability and flexibility. Today, these functions are implemented as software on general-purpose hardware, enabling remote operation and centralized management. The emergence of open-source software has further accelerated this shift, fostering innovation, collaboration, flexibility, and cost efficiency. Notably, the fourth and fifth generations of mobile networks have significantly benefited from these trends in network softwarization and virtualization.

The commercial roll-out of 5G networks presented significant challenges [6]. Due to cost considerations, deployment began gradually, initially in 5G NSA mode, where 5G antennas were added to existing 4G infrastructure. This approach enabled higher downlink throughput and slightly reduced latency [7], while voice services continued to rely on VoLTE technology. In contrast, 5G SA deployments feature both a 5G New Radio (NR) access network and a 5G core network, enabling voice services through the VoNR standard. VoNR offers superior call quality by leveraging 5G's higher data rates, lower latency, and, importantly, its built-in QoS mechanisms.

In [8], a 5G NSA prototype was presented, along with performance measurements of its components implemented using the OpenAirInterface and srsRAN projects. Similarly, [9] described a cost-effective implementation of a 5G SA testbed, utilizing OpenAirInterface for both the radio access network and the core network.

Most existing studies have focused on deploying 5G NSA or SA networks using open-source technologies, with an emphasis on internet connectivity and data services. For example, [10] investigated the use of software-defined radios (SDRs) to enable voice services over a 5G NSA network integrated with an IMS core supporting VoLTE. However, that study did not demonstrate VoNR functionality. In contrast, our work addresses this limitation by integrating multimedia services — particularly voice — into a 5G SA private network. We evaluated the performance of our prototype while working closely with open-source project contributors to extend the Kamailio SIP Server [11], a key IMS control plane component, with a service-based interface to enable native multimedia services within the 5G private network.

In [12], an IMS-compliant conferencing system was developed using open-source tools, with detailed discussion of its design and associated challenges. The study emphasized the importance of QoS and the integration of multimedia elements, such as voice, video, and instant messaging, into a unified, access-agnostic infrastructure. While that work focused specifically on conferencing systems, our research builds on these concepts by integrating the IMS platform with the 5G network, leveraging its QoS mechanisms to establish a robust foundation for multimedia service delivery.

A comparative analysis of VoNR and 4G fallback solutions presented in [13] focused on call setup times and device-level performance, highlighting VoNR's advantage in reducing latency. Similarly, [14] described the design and evaluation of a 5G SA core network for private deployments, integrating VoNR alongside applications such as the Internet of Things (IoT) and Cellular Vehicle-to-Everything (C-V2X). However, these studies relied on proprietary solutions, whereas our work evaluates VoNR performance in private 5G SA networks using open-source implementations.

In our previous work, we reported on the performance of a private 5G SA network [15] built using open-source projects such as srsRAN, OpenAirInterface 5G RAN [16], Open5GS, free5GC [17], and commercial off-the-shelf (COTS) user equipment (UE). We extended this setup by incorporating additional open-source components, including the Kamailio SIP Server and PyHSS, to enable 5G-native voice services, specifically VoNR. Initial performance results of the VoNR prototype were presented in the Softcom'24 conference paper [4]. We further examined the prototype implementation and investigated the impact of network congestion using a simulated environment, with iperf used to generate traffic [18].

In this extended study, we evaluate two deployment scenarios for a voice-enabled 5G network under uncongested conditions: an all-in-one ("in-a-box") setup and a distributed setup. We analyze the similarities and challenges of each scenario, assessing the performance of voice services in both cases. Additionally, we integrate an OTT voice service using the same open-source platforms to better understand the advantages and limitations of OTT solutions compared to natively integrated voice services in 5G networks.

## III. Integrating Voice Services to 5G SA Networks

The architecture of a minimal 5G SA private network, including voice services, is illustrated in Fig. 1. The main architectural components are UE, the Radio Access Network (5G RAN), the 5G Core Network (5G CN), and the IMS core network.

Each UE receives an IP address from the 5G CN, which then routes user traffic to external networks that provide IP-based services. The 5G CN comprises multiple network functions divided into two primary planes: the user plane, responsible for routing end-user traffic, and the control plane, which manages and controls communication sessions between UEs and the network. The control plane performs several critical functions, including session management, mobility management, authentication, authorization, and enforcement of data usage and QoS policies. The Policy and Charging Function (PCF), a control plane component within the 5G CN, manages QoS for multimedia sessions such as voice and video. It plays a key role in setting up these sessions when requested by the IMS core network.

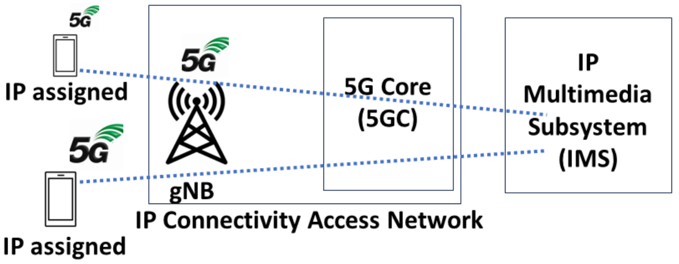


Fig. 1. Main Components of a 5G SA Network including Voice Services.

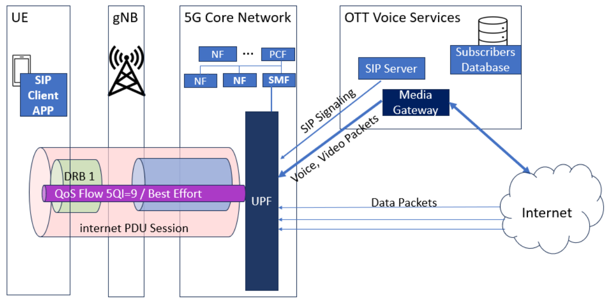


Fig. 2. Prototype Architecture: OTT Voice over 5G Network [18].

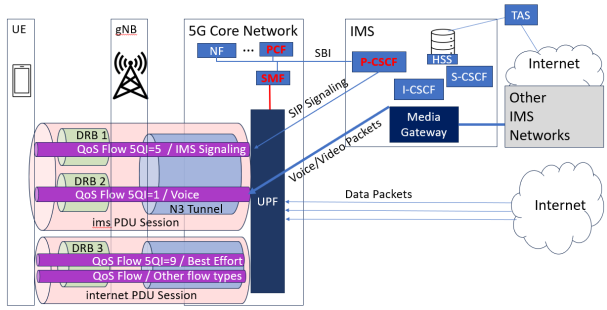


Fig. 3. Prototype Architecture: VoNR for 5G SA Network [18] [4].

### *A. Architectural Considerations for VoNR and OTT Voice in 5G SA Networks*

We investigated two approaches for integrating voice services into 5G SA networks. Once UEs are assigned IP addresses by the 5G CN and establish data connectivity, they can communicate over IP and access voice services delivered via the Internet or other IP-based networks.

When voice sessions using the Session Initiation Protocol (SIP) are established over the default data access point name (APN *internet*), the corresponding user traffic is treated with the same priority as regular data traffic. This approach is referred to as an OTT voice service. Its architecture integrates seamlessly within a 5G SA network environment, as illustrated in Fig. 2.

An alternative method for integrating voice services into a 5G network is to implement 3GPP standards-based VoNR, which builds on the IMS platform using the SIP protocol. This approach enables the use of QoS mechanisms in both the 5G RAN and the 5G CN. In this case, UEs use a dedicated APN, referred to as the *ims* access point, to transmit voice session traffic. IMS voice traffic is tagged with the appropriate 5QI identifiers, 5QI=1 for voice, 5QI=2 for video, and 5QI=5 for SIP signaling, and assigned a higher priority than best-effort data traffic (5QI=9). The architecture is illustrated in Fig. 3.

### *B. Distributed versus In-a-Box Deployment Scenarios*

For each of the two voice integration methods in 5G SA networks, we evaluated two deployment scenarios for placing network functions. In the first scenario, all components, including the 5G RAN, 5G CN, and IMS, are deployed on a single physical machine (in-a-box). In the second, distributed scenario, the 5G RAN is deployed locally, while the 5G CN and IMS run on a cloud-based server. As a result, we implemented four prototypes of a voice-enabled 5G SA network, using the same open-source platforms to ensure a consistent and fair performance comparison.

*1) VoNR Prototype Realization:* Constructing a 5G SA network with VoNR support on a single local machine offers a highly cost-effective solution for research and educational purposes. The system architecture comprises a next generation Node B (gNB) base station running on the local machine connected to a USRP B210 radio unit via USB interface. This primary component manages radio communications with user equipment utilizing the 5G New Radio (NR) air interface. The same machine also hosts the complete 5G CN and IMS infrastructure, implemented using Open5GS, Kamailio, and PyHSS. This approach enables rapid deployment while maintaining a realistic over-the-air testbed environment. The testbed is highly portable, making it particularly suitable for experiments involving nomadic 6G networks. This minimal implementation is illustrated in Fig. 4.

We also constructed a distributed 5G testbed, as illustrated in Fig. 5, utilizing cloud services to host the centralized 5G CN and IMS platform. The cloud-based server configuration provides 4 vCPU cores, 8 GB of RAM, and 80 GB of NVMe SSD storage, running on a KVM virtualization platform. The gNB base station operates on a local machine, with the USRP B210 connected via a USB interface. The backhaul connection to the 5G CN is established over an unreliable third-party transport network—the public Internet. Leveraging cloud services makes the setup highly cost-effective and allows for greater resource availability for network functions. However, this approach introduces additional latency due to the remote 5G CN, with a measured round-trip time of 0.025 s. This trade-off is a key focus of our investigation.

*2) OTT Voice Prototype Realization:* VoNR represents the native approach for integrating voice services into

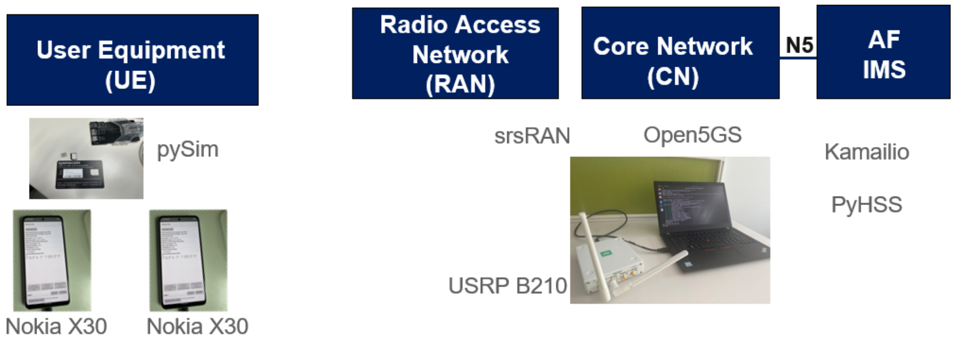


Fig. 4. VoNR Prototype: In-a-Box Deployment.

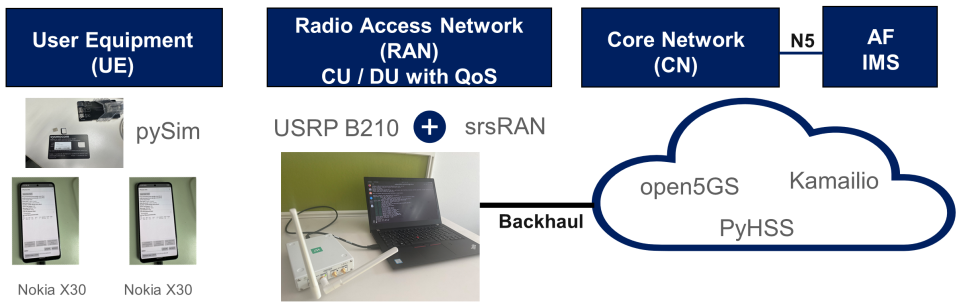


Fig. 5. VoNR Prototype: Cloud-based Deployment [4].

5G networks, eliminating the need for additional software applications to place calls. In contrast, integrating OTT voice services requires supplementary software, commonly referred to as SIP softphones, to enable call functionality. In this case, the IMS platform is not required; instead, the Kamailio SIP server, along with a MySQL database backend, provides sufficient functionality. Fig. 6 and Fig. 7 illustrate the two OTT voice prototypes deployed in a 5G SA network.

### C. UE and SIM Card Customization

The UE used in this study consisted of commercial off-the-shelf devices, along with programmable IP Multimedia Services Identity Modules (ISIMs) of type sysmoISIM-SJA2 [19]. The UE must be VoNR-compliant [20]. As of the time of writing, the availability of such COTS UEs is limited, as most public mobile networks do not yet support VoNR services. The ISIM cards are ETSI/3GPP standards-compliant and can be programmed using the open-source tool pySim [21], as demonstrated in the command example below.

```
./pySim-prog.py -p0 -t sysmoISIM-SJA2
   -a 32627241 -x 001 -y 01
   -i 001010000011000
   -s 8988211000000110000
   -o 398153093661279FB1FC74BE07059FEF
   -k 1D8B2562B992549F20D0F42113EAA6FB
   -n TUBerlin-5G
```

Named *TUBerlin-5G*, the private 5G network is identified by Mobile Country Code (MCC) 001 and Mobile Network Code (MNC) 01. Each ISIM card is uniquely identified by its Integrated Circuit Card Identifier (ICCID), specified using the -s parameter. Subscriber details include the International Mobile Subscriber Identity (IMSI) (e.g., 001010000011000), the subscriber key, and the operator key used for authentication.

In the lab, we also tested the use of special network identifiers—MCC 999 and a placeholder MNC (xy)—which are reserved for development and testing purposes in telecommunication networks. Using MCC 999 helps avoid interference between multiple testbeds, though it requires additional customization when programming ISIM cards. However, for the experiments described in this paper, we used the standard values MCC 001 and MNC 01, as most COTS UEs do not support the 999/xy network identifiers.

The Nokia X30 smartphones were configured to operate in 5G SA mode, with APN set to *internet* for data services and *ims* for multimedia services.

### D. 5G Radio Access Network

To build the gNB base stations, we utilized the open-source project *srsRAN*, paired with a USRP B210 radio unit (RU). The 5G RAN connects mobile devices to the network via a radio link, specifically using the New Radio (NR) interface introduced in 3GPP Release 15.

The local machine running the gNB had the following technical specifications:

```
Lenovo ThinkPad TP14 Gen1,
Linux 6.5.0-35-generic
AMD(R) Ryzen 7 Pro 4750U 16x1.7 GHz
32 GB RAM, 512 GB SSD
```

We customized the configuration parameters to suit our testbed, including IP addresses, RU settings, cell configuration, and QoS parameters. The gNB was then started using the

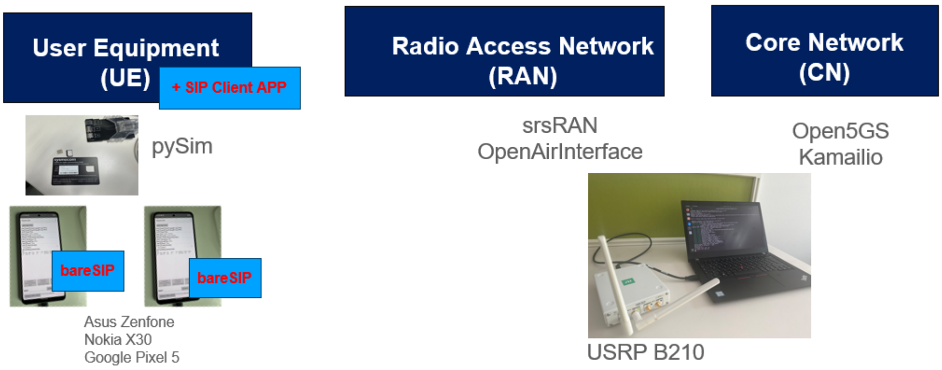


Fig. 6. OTT Voice over 5G Network: In-a-Box Deployment.

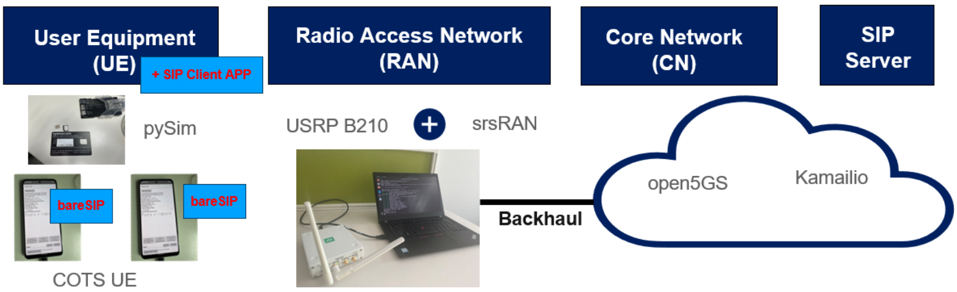


Fig. 7. OTT Voice over 5G Network: Cloud-based Deployment.

specified command, operating at a bandwidth of 20 MHz on 3.75 GHz, as this is the spectrum allocated for private 5G networks in Germany.

```
sudo gnb
-c /path-to/gnb-rf-b200-tdd-n78-20mhz.yml
-c /path-to/qos.yml
```

### *E. 5G Core Network*

We used *Open5GS*, an open-source project [22], to instantiate the 5G CN. Notably, it implements the N5 interface toward the IMS core network, which is essential for VoNR support. After installing and updating the configuration files to match our testbed, such as the IP addresses of the network functions (NFs) and the identifiers of the private 5G network, *Open5GS* was successfully deployed.

### *F. IMS Platform versus SIP-based VoIP Platform*

IMS is a standards-based framework that provides multimedia communication services over IP networks using SIP. The IMS core network constitutes the control layer of the architecture, alongside the transport and application layers. Its primary components include the Proxy, Interrogating, and Serving Call Session Control Functions (P-CSCF, I-CSCF, and S-CSCF), as well as the Home Subscriber Server (HSS).

The HSS stores subscriber profiles and service data, provides authentication vectors for security, and manages subscriber location information. The open-source project *PyHSS* [23] was used to implement the HSS in our setup. The use of *PyHSS* is advantageous, as it supports both 4G and 5G operations through a unified architecture, making it an ideal solution for operators transitioning from 4G to 5G networks by eliminating the need for separate HSS instances. Furthermore, *PyHSS* facilitates the evolution of existing IMS platforms by offering backward-compatible support for CSCF functions, allowing for selective interface updates rather than requiring comprehensive modifications to meet 5G standards.

The open-source *Kamailio* SIP server [11] was used to implement the P-CSCF, I-CSCF, and S-CSCF components, responsible for routing SIP messages between caller and callee. The P-CSCF serves as the entry point for SIP messages originating from IMS subscribers. In our setup, it was instantiated using the *Kamailio* SIP server, which was configured to run a web service enabling communication with the 5G CN over the N5 interface [24], as specified by 3GPP [25]. The P-CSCF acts as a service consumer of the PCF, sending requests to create, update, or delete application session contexts, or to subscribe to event notifications. It also listens for replies and notifications from the PCF related to subscribed events.

After configuring the IMS components according to our testbed requirements, the *pyHSS*, *kamailio-p-cscf*, *kamailio-i-cscf*, and *kamailio-s-cscf* services were successfully deployed.

As an alternative to the IMS platform, OTT voice services rely on a SIP-based VoIP platform. This setup typically includes a SIP server, ideally supported by a database backend. In our testbed, we deployed the *Kamailio* SIP server with a MySQL backend. While the command used to start the server is similar to that of the IMS-based setup, the *Kamailio* configuration file is adapted specifically for this deployment. The *Kamailio* SIP server offers a high degree of flexibility through its configuration file, enabling extensive customization for a wide range of use case scenarios.

### G. Voice Service Access: VoNR versus OTT Voice

*Registration.* In both VoNR and OTT voice setups, the end user must register for the voice service before being able to place calls.

For OTT voice, the user must install an OTT SIP phone application (user agent, or UA) and configure the SIP account with parameters such as the username, password, and SIP server IP address. The registration process for VoIP calls involves the following steps:

1) The UA (e.g., SIP phone) sends a `REGISTER` request to the SIP server—instantiated using the *Kamailio* SIP server—to indicate its current IP address and contact information.
2) The SIP server responds with a `401 Unauthorized` message, challenging the UA to provide authentication credentials.
3) The UA resends the `REGISTER` request, this time including the required credentials (username and password).
4) The SIP server authenticates the UA and, if successful, completes the registration by replying with a `200 OK` response.

For a VoNR setup, the native dialer application of the UE is used and configured to utilize IMS services by setting the *ims* APN on the device. Registration with the IMS platform occurs automatically by sending a `REGISTER` request to the P-CSCF, instantiated using the *Kamailio* SIP server. This process is similar to the SIP registration procedure for the OTT voice setup, except that it uses Authentication and Key Agreement (AKA) for authentication. AKA is a challenge-response mechanism based on symmetric cryptography, which authenticates users and distributes keys. It operates on the ISIM card of the UE.

Upon successful authentication, a PDU session is established for the *ims* APN, including a default bearer for IMS signaling services with a 5QI value of 5.

*Calling.* After successful subscriber authentication, voice calls can be initiated.

For OTT voice, a sequence of SIP messages is exchanged to establish a call. The signaling flow proceeds as follows:

1) The caller's UA sends an `INVITE` request to initiate the call.
2) The SIP server responds with a `100 TRYING` message to indicate it is processing the request.
3) The callee's UA replies with a `180 RINGING` response, indicating that the call is ringing.
4) During this signaling exchange, the caller and callee agree on a common media codec to be used during the media session.
5) When the callee answers the call, their UA sends a `200 OK` response to confirm call establishment.
6) The caller's UA acknowledges this with an `ACK` message.

At this point, voice communication begins between the caller and callee. The session ends when either party sends a `BYE` request to terminate the call, and the other party responds with a `200 OK` message confirming the termination.

For a VoNR setup, a sequence of SIP messages is also exchanged to establish the IMS session and initiate the VoNR call. Since the QoS mechanism of the 5G CN is utilized in this case, a few additional SIP messages are involved. The complete message flow for a VoNR call is illustrated in Appendix A.

As with OTT voice, both UEs must agree on the media codec to be used. Additionally, the IMS platform instructs the 5G CN to reserve a dedicated bearer for voice traffic (5QI = 1). The callee's UE does not begin ringing until the 5G CN confirms the successful reservation of these dedicated resources. Therefore, the SIP message *183 Session in Progress* is used as a provisional response by the terminating UE, including SDP details, such as voice codec capabilities, to inform the originating UE.

The IMS platform informs the PCF of the 5G CN that a flow is required for conversational voice. The PCF then creates a Packet Detection Rule (PDR) for the flow, with QoS parameters corresponding to a 5QI value of 1 (conversational voice). This PDR triggers the 5G CN network function known as the User Plane Function (UPF) to reserve the appropriate resources and establish the necessary data radio bearers.

The originating UE sends a *PRACK* message, acknowledging the provisional response and confirming the final codec selected for the call. The terminating UE sends a 200 OK in response, accepting the final codec, too. The codec for the call is chosen, while the NR resources still need to be reserved.

A SIP *UPDATE* message is sent by the originating UE to inform the terminating UE that radio resources have been successfully allocated on its side. Upon receiving this message, the terminating UE responds with a *200 OK*, confirming that the necessary resources are also reserved on its end.

Only once both parties have successfully reserved the required radio resources with the appropriate QoS does the terminating UE begin to ring, sending a *180 RINGING* message to the originating UE. When the callee answers the call, their UE sends a *200 OK* response to the caller.

The originating UE then acknowledges this with an *ACK* message, at which point the call is fully established. Voice communication proceeds over Real-time Transport Protocol (RTP) streams exchanged between the caller and callee. When either party ends the call, their UE sends a *BYE* message, prompting the IMS core network to release the radio resources reserved for the terminated session.

The complete sequence of SIP messages, captured using Wireshark during a VoNR call, is provided in [18].

## IV. Evaluation

3GPP defines a set of key performance indicators (KPIs) for various components of the 5G system [26]–[28]. In this section, we highlight a subset of these KPIs relevant to VoNR and IMS-based voice service evaluation.

The performance of the radio network can be assessed using KPIs such as VoNR call drop rate, handover success rate, and Single Radio Voice Call Continuity (SRVCC) success rate. The health of the 5G CN is indicated by metrics including VoNR attach success rate, VoNR paging success rate, and VoNR PDU session modification success rate. The performance of the IMS core network can be evaluated by monitoring various metrics collected at the P-CSCF or the telephony application server (TAS), with typical healthy values presented in Table I.

TABLE I
IMS CORE NETWORK KPIS [4], [26]–[28].

| *KPI* | *Description* | *Requirement* |
|---|---|---|
| CST | Call setup time (s) | $< 2.5\ s$ |
| RTP-packets-lost | Voice packets lost (%) | $< 1$ % |
| RTP-Jitter | Variation in delay (ms) | $< 30\ ms$ |
| RSR | Registration Success Ratio | $> 99$ % |
| CSSR | Call Setup Success Ratio | $> 99$ % |

In our evaluation, we did not measure the Radio Setup Success Rate (RSR) and Call Setup Success Rate (CSSR), as these metrics are more applicable to production-ready deployments and require more complex and costly setups. Given the constraints of a limited test environment with only a few UEs available, we focused on Call Setup Time (CST), RTP jitter, and RTP packet loss to objectively assess the technical performance of the voice service prototypes in two deployment scenarios: in-a-box and distributed.

CST is measured at the P-CSCF, in seconds, as the time elapsed between the receipt of the *SIP INVITE* from the caller and the transmission of the *180 RINGING* message back to the caller [28], as illustrated in Fig. 22. Under healthy network conditions, CST values are expected to remain below 2.5 s.

$$CST = Time_{180-RINGING} - Time_{INVITE} \tag{1}$$

For each of the four setups, we initiated $N = 22$ voice calls. For each call, network packets were captured in a `pcap` file using a packet sniffer, such as *TShark*, on the machine hosting the IMS platform or SIP server. Based on the captured traffic, we measured call setup time, RTP packet loss, and RTP jitter. Since each call generated two RTP flows (one in each direction), a total of 44 voice streams were analyzed per setup.

Following the spectrum allocation guidelines for private 5G networks in Germany, the gNB was configured to operate at a frequency of 3.75 GHz with a bandwidth of 20 MHz. For all measurements, the UEs remained within the coverage area of the same USRP B210 radio unit, positioned approximately three meters away.

### A. Call Setup Evaluation.

The clustered column chart presented in Fig. 8 shows the call setup times for the $N = 22$ calls placed over VoNR in cloud-based deployment. We observed the values fall in the interval 0.6 s to 1.2 s, the average call setup time being 0.92 s.

Fig. 9 presents the call setup times measured while placing $N = 22$ calls using OTT voice services in a cloud-based (distributed) deployment.

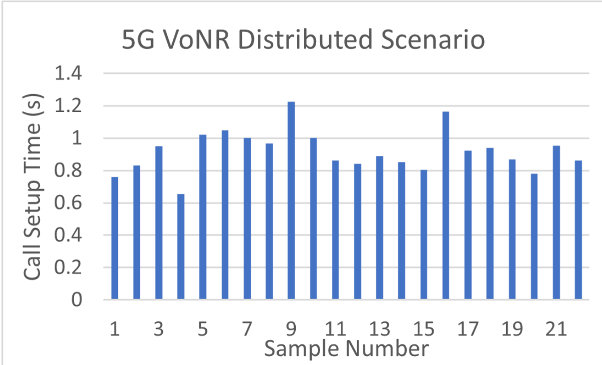


Fig. 8. Call Setup Time on Voice Traffic during VoNR Calls in Cloud-based Scenario [4].

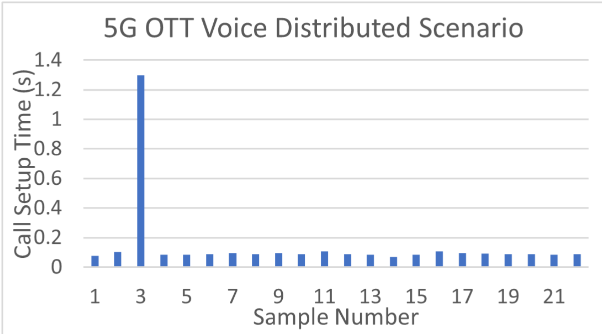


Fig. 9. Call Setup Time on voice traffic during OTT Voice calls / Cloud-based scenario [4].

In this scenario, the average call setup time was 0.09 s. Sample 3 was identified as an outlier; therefore, the average was computed excluding the minimum and maximum values to avoid skewing the result.

The CST value represents the total time required for the network to establish a call, including any radio resource reservation procedures. When comparing VoNR and OTT voice setups, we observe that VoNR requires approximately 0.83 s longer to complete call setup. Since OTT voice services operate without dedicated radio resource allocation, this additional time in the VoNR case can be attributed to the reservation of radio resources within the 5G network.

Fig. 10 presents the call setup times measured while placing $N = 22$ calls using OTT voice services in the in-a-box deployment. The average value was 0.35 s for VoNR call setup in the in-a-Box scenario.

Fig. 11 presents the call setup times measured while placing $N = 22$ calls using OTT voice services in the in-a-box deployment. The average CST in this scenario was 0.06 s.

TABLE II
CALL SETUP AVERAGE VALUES (S).

| *Scenario* | *VoNR* | *OTT Voice* |
|---|---|---|
| In-a-Box | 0.35 | 0.06 |
| Distributed | 0.92 | 0.09 |

The average values of the CST metric measured in the four tested scenarios are summarized in Table II.

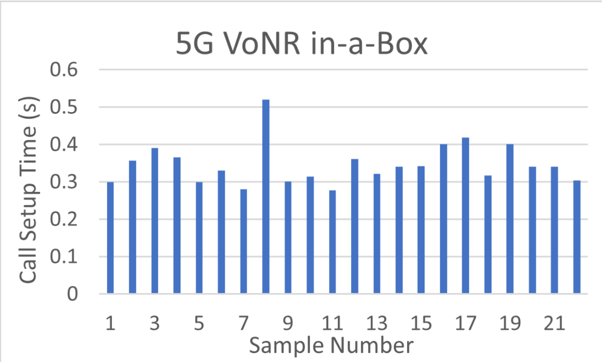


Fig. 10. Call Setup Time on voice traffic during VoNR calls / 5G in-a-Box scenario.

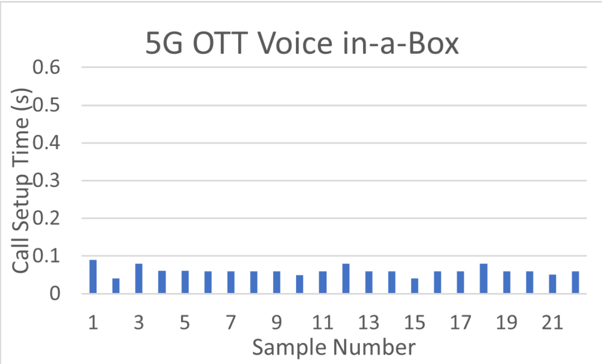


Fig. 11. Call Setup Time on voice traffic during OTT Voice calls / 5G in-a-Box scenario.

We observed that the cloud-based setup introduced varying delays for voice call setup, depending on the service type. For OTT voice calls, the cloud setup added approximately 0.03 s, closely matching the measured backhaul round-trip time of 0.025 s. In contrast, VoNR calls experienced a more substantial delay of 0.57 s in the cloud setup. For comparison, in the in-a-box scenario, VoNR calls required 0.35 s for setup, while OTT voice calls took only 0.06 s. This difference in setup times suggests that VoNR requires approximately 0.29 s for the full procedure of reserving dedicated radio resources in the in-a-box configuration.

An analysis of the SIP call setup flows for VoNR and OTT revealed that VoNR involves several additional SIP message exchanges between the two UEs, reflecting its more sophisticated mechanism for dedicated radio resource reservation. Notably, the receiving UE remains silent until explicit confirmation of resource allocation is received, ensuring reliable voice service quality and preventing premature alerting.

The distributed deployment architecture introduces additional latency in VoNR call establishment. Specifically, UE-to-UE signaling incurs a doubled round-trip time (approximately 0.05 s) compared to communication with a cloud server. When combined with six additional SIP message exchanges, this contributes around 0.3 s of delay. Moreover, the need for coordination between the remote 5G CN and the locally deployed gNB over the backhaul introduces further signaling overhead, resulting in a total increase of approximately 0.57 s in call setup time compared to the centralized (in-a-box) deployment.

### *B. Media Session Evaluation.*

The analysis of RTP streams established during the calls revealed that all call types across the four scenarios experienced jitter values under 30 ms, meeting the recommended threshold to avoid degradation in call quality.

Charts showing the maximum jitter (ms) and mean jitter (ms) are presented in Fig. 12, Fig. 13, Fig. 16, and Fig. 17. We observe that the voice streams of VoNR calls exhibit mean jitter values around 20 ms or 2 ms. In contrast, jitter values in the OTT scenarios vary more significantly across samples.

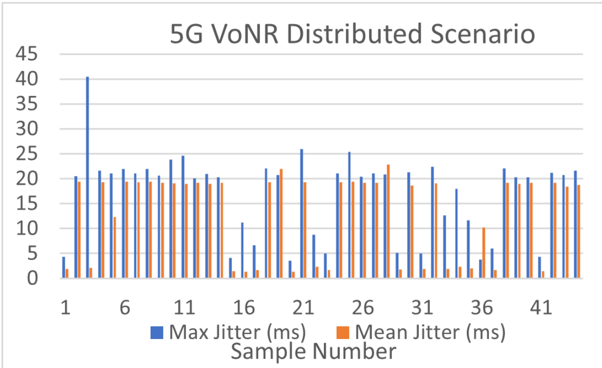


Fig. 12. Jitter on voice traffic during VoNR calls / Cloud-based Deployment [4].

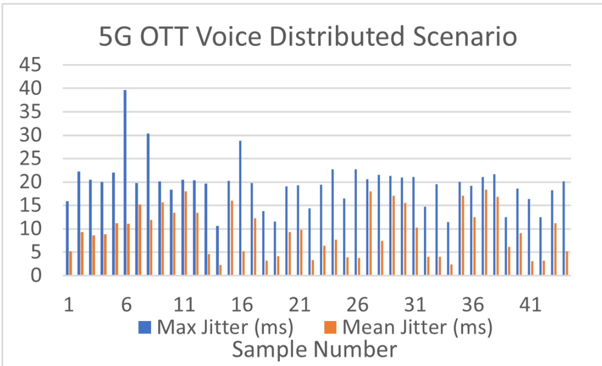


Fig. 13. Jitter on voice traffic during OTT calls / Cloud-based Deployment [4].

The mean jitter values showed improvement in the in-a-box scenario for OTT, with an average of 4.62 ms compared to 9.43 ms in the distributed deployment.

For VoNR calls, jitter values tended to cluster around either 20 ms or below 5 ms. This pattern was also observed in the in-a-box deployment, where jitter values across a series of calls were either consistently below 5 ms or around 20 ms, depending on the time of the experiment.

During one sequential test run involving 10 calls, the UEs were unexpectedly disconnected from the network and had to be manually reconnected. This behavior may indicate instability in the implementation of the QoS mechanism within the 5G RAN and 5G CN software stacks.

We acknowledge that jitter can be influenced by various factors, including hardware performance of the SDR, environmental conditions (e.g., congestion or interference), or suboptimal network configuration. However, during our experiments, external conditions remained relatively stable, while the fluctuations in observed jitter appeared abrupt. This suggests potential issues with QoS handling, most likely on the radio side, where bearer allocation and modification should occur in accordance with QoS requests from the 5G core network.

Nevertheless, all observed jitter values remained within the healthy threshold of 30 ms and did not affect the perceived voice call quality.

In the experimental measurements, the average packet loss was 0.043 % for VoNR calls and 0.039 % for OTT calls. Since the network was not congested, the observed packet loss is considered negligible. The charts illustrating packet loss in each of the tested scenarios are shown in Fig. 14, Fig.15, Fig.18, and Fig.19.

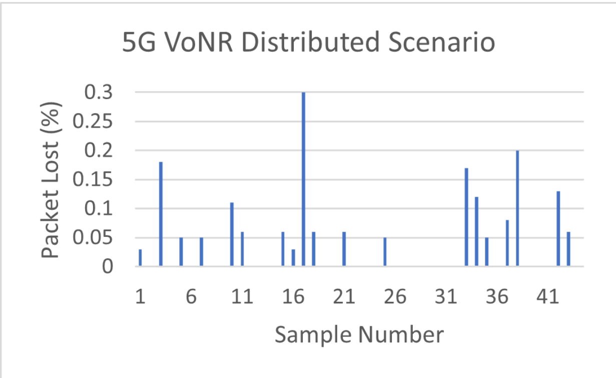


Fig. 14. Packet loss on voice traffic during VoNR calls / Distributed Scenario.

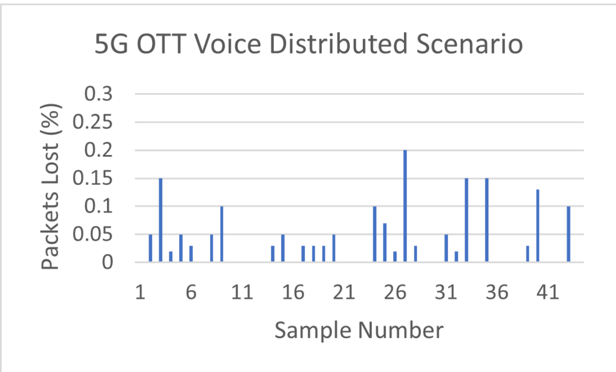


Fig. 15. Packet loss on voice traffic during OTT calls / Distributed Scenario.

## V. Conclusion

This paper presented the architecture and integration of VoNR services into a 5G SA private network using open-source software, including *srsRAN*, *open5GS*, *Kamailio*, and *PyHSS*. It also described the architecture of OTT voice services added to a private 5G SA network. To support end-to-end voice communication over 5G, four deployment scenarios were implemented and analyzed, covering both

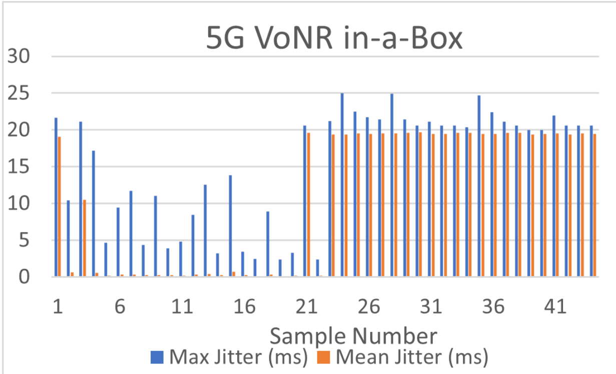


Fig. 16. Jitter on voice traffic during VoNR calls / In-a-box Deployment.

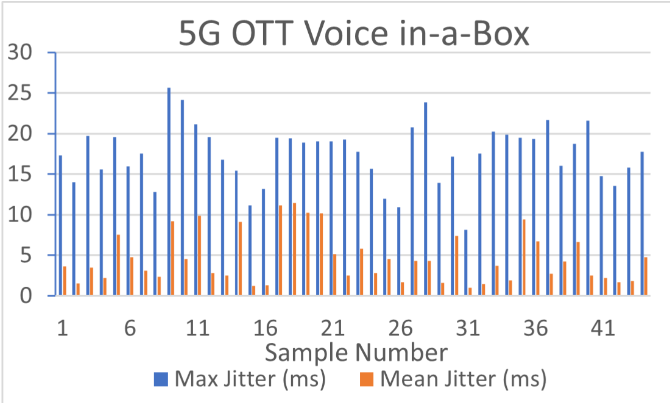


Fig. 17. Jitter on voice traffic during OTT calls / In-a-box Deployment.

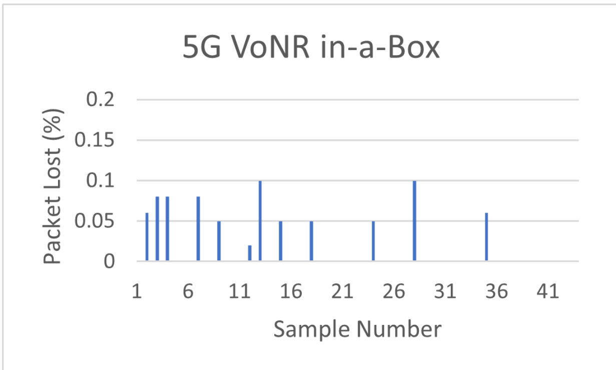


Fig. 18. Packet loss on voice traffic during VoNR calls / In-a-Box Scenario.

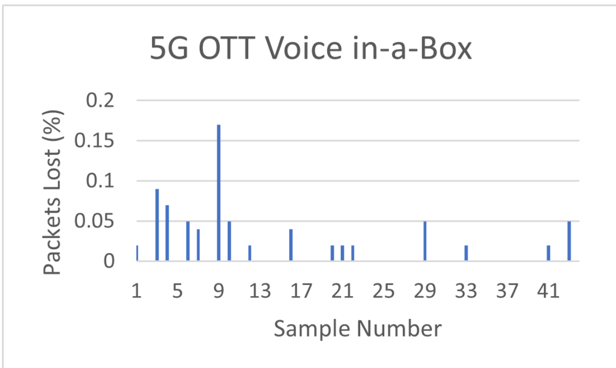


Fig. 19. Packet loss on voice traffic during OTT calls / In-a-Box Scenario.

monolithic and distributed configurations for VoNR and OTT voice technologies.

The paper evaluated the performance of the four prototypes by analyzing key metrics such as call setup time, jitter, and packet loss. The results highlighted the additional delay introduced by VoNR due to the allocation of dedicated radio resources through its QoS mechanisms. While the RTP stream analysis showed similar performance across all scenarios, VoNR calls exhibited more stable jitter, with values consistently falling within acceptable thresholds. The results also indicated signs of instability in the QoS implementation of the open-source 5G RAN and core network software. As expected, packet loss was negligible, as the experimental 5G private networks operated in a congestion-free environment.

Developing an open-source VoNR prototype enables detailed investigation of the mechanisms underpinning multimedia service integration, which is essential for the advancement of next-generation 6G networks. Future work will involve evaluating both VoNR and OTT voice prototypes in more realistic environments that emulate commercial network conditions, allowing for the identification of potential deployment and operational challenges. Initial findings under congested conditions have already been reported in our earlier work [18].

Further research should explore the comparative advantages of VoNR's QoS mechanisms versus OTT voice services. Such an investigation may inform the design of future 6G multimedia services that combine VoNR's guaranteed performance with the flexibility and advanced features of OTT voice, including rich communication services, enhanced call management, and seamless service integration.

## ACKNOWLEDGMENT

The authors acknowledge the financial support by the German Federal Ministry for Education and Research (BMBF) within the project "Open6GHub" [2]grant no. 16KISK003K.

We thank Kamailio Project's developers who provided insight and expertise that greatly contributed to this research.

[2]www.open6ghub.de/en/

## APPENDIX A
## DIAGRAMS FOR SERVICE REGISTRATION AND CALL SETUPS

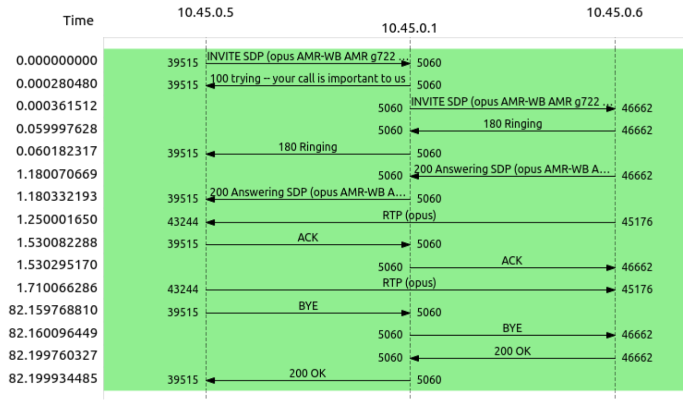


Fig. 20. OTT Voice Call Setup Flow shown in Wireshark.

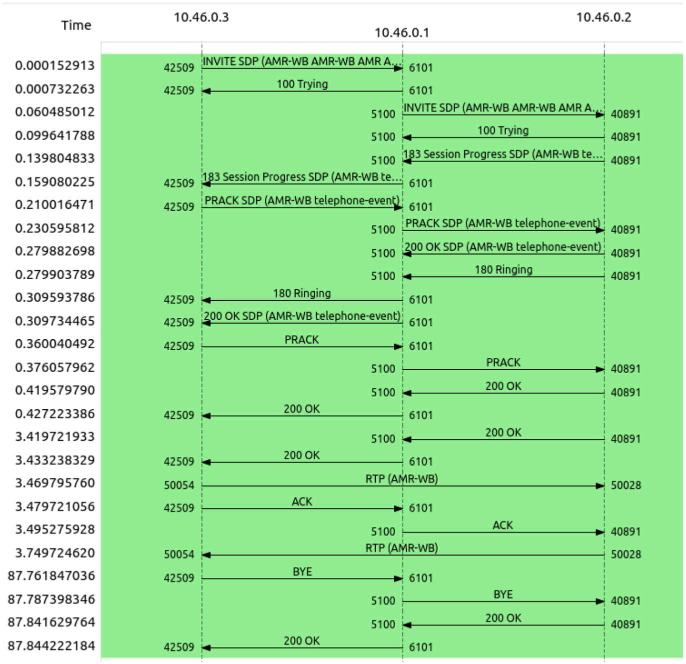


Fig. 21. VoNR Call Setup Flow shown in Wireshark.

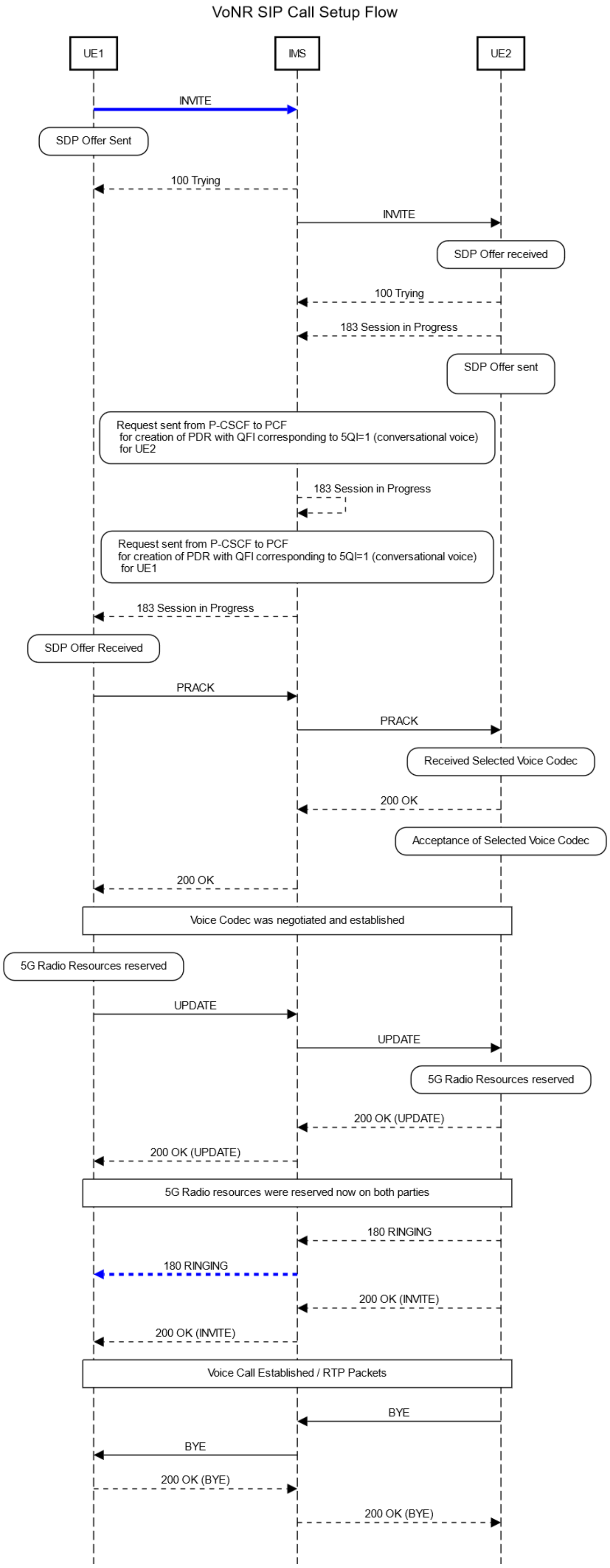


Fig. 22. VoNR Call Setup Flow [4].

## REFERENCES


[1] ericsson-whitepaper 2022, "Voice and communication services in 4G and 5G networks," https://www.ericsson.com/en/reports-and-papers/white-papers/voice-and-communication-services-in-4g-and-5g-networks, Accessed: 2024-05-25.

[2] 3rd Generation Partnership Project (3GPP), "Technical specification group services and system aspects; system architecture for the 5g system (5gs); stage 2," 3GPP, Tech. Rep. TS 23.501, 2024. [Online]. Available: https://www.3gpp.org/ftp/Specs/archive/23_series/23.501/

[3] ——, "Technical specification group core network and terminals; policy and charging control over rx reference point; stage 3," 3GPP, Tech. Rep. TS 29.214, 2024. [Online]. Available: https://www.3gpp.org/ftp/Specs/archive/29_series/29.214/

[4] E.-R. Modroiu, J. Mwangama, M. Corici, and T. Magedanz, "Integrating open-source vonr in 5g sa private networks: An experimental evaluation," in *2024 International Conference on Software, Telecommunications and Computer Networks (SoftCOM)*, 2024, pp. 1–6.

[5] V.-G. Nguyen, A. Brunstrom, K.-J. Grinnemo, and J. Taheri, "Sdn/nfv-based mobile packet core network architectures: A survey," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 3, pp. 1567–1602, 2017.

[6] K. Blind and C. Niebel, "5g roll-out failures addressed by innovation policies in the eu," *Technological Forecasting and Social Change*, vol. 180, p. 121673, 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0040162522002050

[7] K. Kousias, M. Rajiullah, G. Caso, O. Alay, A. Brunstorm, L. De Nardis, M. Neri, U. Ali, and M.-G. Di Benedetto, "Coverage and performance analysis of 5g non-standalone deployments," in *Proceedings of the 16th ACM Workshop on Wireless Network Testbeds, Experimental Evaluation & CHaracterization*, ser. WiNTECH '22. New York, NY, USA: Association for Computing Machinery, 2022, p. 61–68. [Online]. Available: https://doi.org/10.1145/3556564.3558233

[8] R. Mihai, R. Craciunescu, A. Martian, F. Y. Li, C. Patachia, and M.-C. Vochin, "Open-source enabled beyond 5g private mobile networks: From concept to prototype," in *2022 25th International Symposium on Wireless Personal Multimedia Communications (WPMC)*, 2022, pp. 181–186.

[9] F. John, J. Schuljak, L. B. Vosteen, B. Sievers, A. Hanemann, and H. Hellbrück, "A reference deployment of a minimal open-source private industry and campus 5g standalone (sa) system," in *2022 IEEE 10th International Conference on Information, Communication and Networks (ICICN)*, 2022, pp. 1–9.

[10] J. J. Abularach Arnez, W. Medeiros Silva, R. K. Gomes Do Reis, L. A. Da Silva, and M. G. Lima Damasceno, "Sdr 5g nsa mobile network and an ims core to provide voice over ip lte service," in *2022 IEEE-APS Topical Conference on Antennas and Propagation in Wireless Communications (APWC)*, 2022, pp. 071–076.

[11] K. G. Repository, "Kamailio: The Open Source SIP Server for large VoIP and real-time communication platforms," https://github.com/kamailio, Accessed: 2024-05-26.

[12] A. Amirante, A. Buono, T. Castaldi, L. Miniero, and S. P. Romano, "Centralized conferencing in the ip multimedia subsystem: from theory to practice," *Journal of Communications Software and Systems*, vol. 4, p. 80, 03 2008.

[13] M. F. De Aguiar, J. K. S. Carvalho, V. De A. Rodrigues, A. F. Chabi, J. V. Da S. Campos, L. R. Lopes, and J. O. De Sousa, "Performance analysis of 5g voice solutions via eps fallback and vonr," in *2023 IEEE Latin-American Conference on Communications (LATINCOM)*, 2023, pp. 1–6.

[14] A. H. S and A. K. R, "Designing, dimensioning and testing sdn-nfv private cloud architecture based realisation of 5gc with iiot, mcs, cv2x & vonr service," in *2022 IEEE Wireless Antenna and Microwave Symposium (WAMS)*, 2022, pp. 1–5.

[15] M. Chepkoech, E.-R. Modroiu, J. Mwangama, M. Corici, and T. Magedanz, "Evaluation of oss-enabled openran compliant 5g standalone campus networks," in *2023 International Conference on Electrical, Computer and Energy Technologies (ICECET)*, 2023, pp. 1–7.

[16] OpenAirInterface, "Openairinterface 5g radio access network (ran)," https://openairinterface.org/oai-5g-ran-project/, 2025, accessed: 2025-01-28.

[17] Free5GC Project, "Free5GC," https://github.com/free5gc/free5gc, Accessed: 2024-05-26.

[18] E.-R. Modroiu, J. Mwangama, B. M. Corici, Marius, and T. Magedanz, "Performance of vonr and ott voice in congested and uncongested

5g networks," in *Proceedings of the IEEE Virtual Conference on Communications (IEEE VCC 2024)*, 2024.
[19] pySim Products, "Programmable SIM Cards: Sysmocom," https://sysmocom.de/products/sim/sysmousim/index.html, Accessed: 2024-05-25.
[20] GSM Association, "NG.114 - IMS Profile for Voice, Video and Messaging over 5GS," 2020.
[21] pySim Repository, "pySim - SIM cards programming tool," https://gitea.osmocom.org/sim-card/pysim, Accessed: 2024-05-25.
[22] O. G. Repository, " Open5GS ," https://github.com/open5gs/open5gs, Accessed: 2025-05-26.
[23] P. G. Repository, " PyHSS - Diameter Home Subscriber Server ," https://github.com/nickvsnetworking/pyhss, Accessed: 2024-05-25.
[24] 3GPP, "Npcf PolicyAuthorization Open API," 2024.
[25] 3GPP , " 5G System; Access and Mobility Policy Authorization Service; Stage 3 (TS 29.514 Release 17) ," 2022 .
[26] 3GPP, "Umts; lte; telecommunication management; key performance indicators (kpi) for the ip multimedia subsystem (ims)," 3GPP, Tech. Rep. 3GPP TS 32.454 version 18.0.0 Release 18, 2024.
[27] GSM Association, "Definition of Quality of Service parameters and their computation, Version 9.0, 07 June 2018," https://www.gsma.com/newsroom/wp-content/uploads//IR.42-v9.0.pdf, Accessed: 2024-05-26.
[28] 3GPP, "Digital cellular telecommunications system (Phase 2+) (GSM); Universal Mobile Telecommunications System (UMTS); LTE; Telecommunication management; Performance Management (PM); Performance measurements; IP Multimedia Subsystem (IMS)," 3GPP, Tech. Rep. 3GPP TS 32.409 version 18.0.0 Release 18, 2024.

**Elena-Ramona Modroiu** is a researcher at Technische Universität Berlin (DE); her research work covers the field of 5G mobile networks, end-to-end setups with a focus on core networks, and the evolution from 5G to 6G mobile networks. In 2003, she completed Master in Computers Science at Politehnica University Bucharest (RO) and continued with post-graduation studies at Universidad Politechnica Valencia (ES). She has been actively involved in VoIP/SIP co-founding the Kamailio open source project, the SIP server at the core of IMS solutions for 4G/5G open-source based deployments.

**Joyce Mwangama** holds the position of an associate professor and researcher in the Department of Electrical Engineering at the University of Cape Town (UCT), in Cape Town, South Africa. She obtained her PhD and MSc in Electrical Engineering, along with a BSc in Electrical and Computer Engineering, all from the University of Cape Town. Her prominent research interests centre on communication networks, next-generation networks, and digital health, where she leads efforts in setting up experimental next-generation mobile testbeds. Additionally, Joyce provides her expertise and guidance to the Independent Communications Authority of South Africa 5G Forum, contributing to the research and development of 5G/6G technologies.

**Marius Corici** (Dr. Eng.) obtained his Diploma Engineering degree at the Politehnica University of Bucharest and his PhD at Technische Universität Berlin. He is Deputy Head of the Software-Based Networks Division at Fraunhofer FOKUS Institute, leading the research and development teams for the Open5GCore, Open6GCore and OpenLANES toolkits, acting as a Research Pathfinder and contributing to the design and specification of beyond-5G features and 6G architectures. He is also researcher at Technische Universität Berlin.

**Thomas Magedanz** (PhD) has been a professor at the Technische Universität Berlin, Germany, leading the Chair of Next-Generation Networks since 2004. He is also director of the business unit Software-based Networks at Fraunhofer FOKUS Institute in Berlin. His research focuses on software-based networks for various vertical industries, with a particular emphasis on building public and non-public campus networks. Over the past two decades, he has been instrumental in creating open software toolkits for building vendor-independent open testbeds for next-generation mobile networks, such as OpenIMSCore, OpenEPC, OpenMTC, OpenBaton, and Open5GCore. His current interest is in the evolution from 5G to 6G and the development of the Organic 6G Core Network.