

Real-World Integration and Evaluation of Open-Source 5G Core with Commercial RAN

Guoying Zu, Joshua Ofori Boateng, Varun S. Advani, Taimoor Ul Islam, Vincent Lee, Sarath Babu, Md Nadim, Daji Qiao, Mohamed Y. Selim, Hongwei Zhang

Department of Electrical and Computer Engineering, Iowa State University, Ames, Iowa, USA
{gyzu, jboateng, vsadvani, tislam, vincet, sarath4, nadim, daji, myoussef, hongwei}@iastate.edu

Abstract—Open-source platforms are reshaping cellular networks by decoupling proprietary stacks and enabling rapid innovation across radio access networks (RANs) and cellular cores. Production-quality open-source 5G cores have become available for years, yet there lacks rigorous, publicly-available studies on the feasibility and results of integrating open-source cores with commercial RANs. In this work, we present the first field-validated integration of Open5GS, a fully open-source 5G core, with a commercial Ericsson RAN in the ARA wireless living lab, spanning a diameter of 30 km in central Iowa. We detail the integration workflow, practical challenges, and lessons learned. We perform extensive real-world field measurements and show that Open5GS offers reliable performance by achieving more than 200 Mbps per UE in 90% of our field measurements, comfortably exceeding the enhanced mobile broadband (eMBB) requirements. Our results demonstrate that open-source cores can achieve carrier-grade performance and robustness with commercial RANs, thereby accelerating vendor-neutral, rapid innovation while offering affordable platforms for 5G adoption in private networks, rural communities, and so on.

I. INTRODUCTION

The evolution of 5G-and-beyond networks is increasingly being shaped by virtualization and service-based architectures (SBAs), which enhance scalability, flexibility, and service agility while empowering open-source innovations. Open-source core networks (OCNs) offer cost-effective, customizable alternatives to proprietary solutions; they also offer transparency and programmability, allowing innovators to explore cutting-edge core architectures and network functions (NFs). Such capabilities make OCNs beneficial for mission-critical applications, supporting both private deployment decoupled from vendors and standalone operations that have stringent security requirements. Integrating OCNs with commercial-grade RANs can further amplify the impact by enabling real-world evaluation and adoption of innovations in cellular cores, which are often lacking in simulation- and experimental SDR-based studies.

Several studies have evaluated individual OCN platforms such as Open5GS [1], OAI 5GC [2], Aether SD-Core [3], and Free5GC [4], focusing on their feature sets and maturity. Barbosa et al. [5] reported that Open5GS achieves lower control-plane latency, OAI 5GC offers higher throughput, and Free5GC is more resource efficient. Other works [6], [7] highlighted deployment challenges on general-purpose hardware and the need for extensive validation. Interoperability studies [8] using open-source gNBs such as srsRAN [9] and

OAI [10] showed minimal performance differences. Most of these studies rely on SDR-based gNBs in small-scale testbeds, and lack real-world evaluations with commercial RANs. For example, Håkegård et al. [11] analyzed the integration of Open5GS with srsRAN and showed the limitations of SDR in coverage and latency. Testbeds such as POWDER, COSMOS, and AERPAW [12] have deployed OCN platforms with open-source RANs only.

To fill the aforementioned gaps, we investigate the integration of Open5GS with state-of-the-art Ericsson massive MIMO gNBs in the ARA wireless living lab [13]. Compared with OAI 5GC (more tightly coupled to its own RAN) and Free5GC (lightweight but less validated at scale), Open5GS has been shown to offer mature interoperability with heterogeneous RANs, achieve lower control-plane latency, provide practical management tools, and, more importantly, was the only open-source 5G core that could be reliably integrated with Ericsson massive MIMO gNBs at the time of writing. A more detailed comparison between the aforementioned OCNs can be found in [5], [6]. The main contributions are as follows: (1) First-of-its-kind integration of Open5GS with Ericsson massive MIMO RAN, providing insights into real-world deployment scenarios; (2) A detailed integration blueprint that extends beyond the official Open5GS documentation, highlighting critical configuration challenges and necessary adjustment for successful integration; (3) Extensive performance evaluation, comparing Open5GS and Ericsson cores, demonstrating the reliability and performance of Open5GS; and (4) Performance assessment of simulated and field-deployed UEs with Open5GS, highlighting the importance of not relying exclusively on simulation-based evaluations of open-source core networks.

Our integration of Open5GS OCN with commercial Ericsson RANs provides a flexible, extensible, and cost-effective platform for developing and evaluating novel 5G-and-beyond systems under real-world wireless conditions with live traffic. Beyond its expected research impact, this integrated platform can significantly benefit rural and underserved regions, where low user-density and limited budgets make full-scale commercial 5G-and-beyond deployments impractical. By leveraging the integration of an OCN with a commercial RAN, these regions can build sustainable, locally managed 5G-and-beyond networks without compromising functionality.

The remainder of this paper is organized as follows. Sec-

tion II presents the system architecture and deployment. Section III provides the detailed procedure and lessons learned for integrating Open5GS core with Ericsson gNBs. Section IV summarizes the performance evaluation results of the Ericsson gNBs with Ericsson core and Open5GS core. Finally, the paper concludes in Section V.

II. SYSTEM ARCHITECTURE

In this section, we first present the key components and the overall architecture of a 5G standalone (SA) system. We then explain the details of a particular implementation of such a 5G system in the ARA wireless living lab [13], which is based on Ericsson gNBs [14] and Quectel UEs [15], with Ericsson's native 5G core and the open-source Open5GS core available.

A. Overview of 5G SA System

5G systems can be deployed in two configurations: SA and non-standalone (NSA). NSA leverages the existing 4G LTE infrastructure, using the LTE core to anchor control signaling while introducing 5G new radio (NR) for enhanced data throughput. SA utilizes a dedicated 5G core to deliver full 5G capabilities such as low latency, network slicing, and scalability. Specifically, the SA architecture introduces a SBA that decouples the control plane (CP) and user plane (UP), allowing flexible deployment and scaling of NFs. It comprises the RAN—including the gNodeB (gNB) and user equipment (UE)—and the core network (CN). In this subsection, we focus on the key concepts that are relevant to the integration of an open-source core network with a commercial RAN.

As shown in Fig. 1, the core network consists of a set of modularized NFs such as access and mobility management function (AMF), session management function (SMF), and user plane function (UPF), which communicate over standardized reference points such as **N2**, **N3**, and **N6** [16]. Key protocols for establishing and maintaining connectivity between the UE and the data network include (i) next generation application protocol (NGAP) [17], which is used to establish the connection between the gNB and AMF over **N2**, and (ii) non-access stratum (NAS) [18], which is tunneled within NGAP to support UE authentication, registration, and session management. At the RAN side, the gNB interfaces with the core network over **N2** for signaling, and over **N3** for data transfer to the UPF. The radio resource control (RRC) protocol manages radio resources between the gNB and UE.

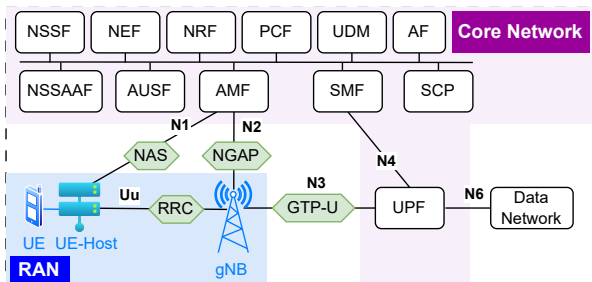


Fig. 1: Generic 5G SA system architecture. [16]

To evaluate open-source core and commercial RAN integration, we focus on three standardized procedures [17]: Registration, Service Request, and PDU Session Establishment. They handle UE authentication and context setup, reactivation of idle UEs, and establishment of data tunnels via SMF and UPF respectively. Fig. 2 lists the NAS messages of these procedures. **Registration latency** is measured from Registration request to Registration completion; **Service request latency** from Service request to Service acceptance; and **PDU session establishment latency** from PDU session establishment request to acceptance.

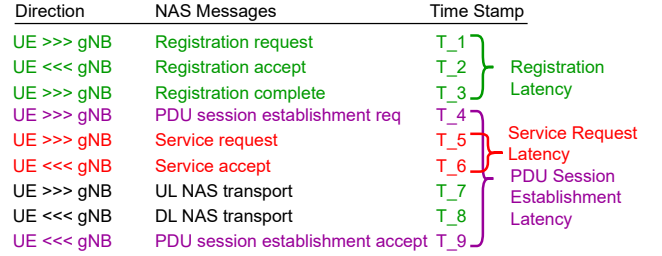


Fig. 2: NAS messages used in a 5G SA system.

Although 3GPP rigorously defines all protocols and procedures for 5G systems, the performance of such a system in real-world deployments varies, especially when combining heterogeneous components such as a commercial RAN and an open-source core network. Next, we will describe a particular real-world deployment of a 5G SA system in the ARA wireless living lab.

B. 5G SA in ARA Wireless Living Lab

ARA [13] is an at-scale real-world testbed for advanced wireless research, deployed across the Iowa State University (ISU) campus, the City of Ames (where ISU resides), and surrounding research and producer farms as well as rural communities in central Iowa. One of the important experimental platforms featured on ARA is a 5G SA system based on Ericsson RAN technology.

The 5G SA system in ARA consists of four Ericsson gNBs (each with three sectors), a 5G core network, a data center, and multiple UE sites. Each sector of the gNB is equipped with a mid-band AIR 6419 radio and a millimeter wave AIR 5322 radio. ARA has deployed more than 30 UE sites with Quectel RG530 radios across residential areas and farm fields. Table I summarizes the key features and capabilities of ARA's Ericsson RAN.

TABLE I: Features & Capabilities of Ericsson gNBs in ARA

Radio Model	Operating band	Bandwidth	Capacity	Range
AIR 6419	3.45–3.55 GHz	100 MHz	1 Gbps	8.5+ km
AIR 5322	27.5–27.9 GHz	400 MHz	2 Gbps	500 m

Fig. 3 shows the pictures of an example Ericsson gNB and an example Quectel UE site at ISU research farms. In the initial configuration of ARA's 5G SA system, gNBs and UEs

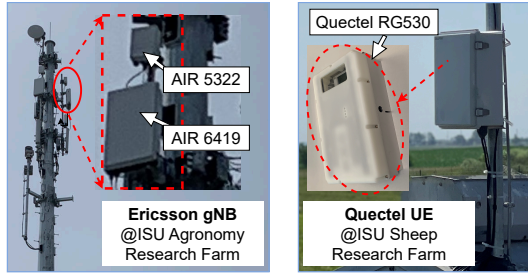


Fig. 3: Example sites in the ARA 5G deployment.

were served by the Ericsson’s native core network (E-5GC). As shown in Fig. 4a, with E-5GC, each Ericsson gNB (blue block) is served by a geographically split core network that is connected by a global network of 100 Gbps fiber links and switches. The control plane resides in an Ericsson lab in California (purple block), while the user plane sits in the ARA data center on the ISU campus (green block). Signaling in the control plane between the gNB and E-5GC travels through the Ericsson 6675 router, the ARA switches, the ITS router, and the firewall before reaching the data network and eventually the control plane located in California. Meanwhile, the user plane traffic is routed via the Ericsson 6676 router and ARA switches to the UPF and local packet gateway (LPG) hosted in the ARA data center, and then forwarded through the ITS router and the firewall to the data network.

Recently, ARA has transitioned its 5G SA system from E-5GC to an open-source core network to support more advanced experimentation. Open5GS [1] has been selected as the open-source core network for ARA, due to its complete functionality and stable performance. Researchers can access the ARA’s 5G SA system remotely through the ARA web portal [19] that provides containerized core network instances for various experiments with real-world gNBs and UEs. For example, users can deploy and test custom enhancements to Open5GS, validate different core network functions, and benchmark performance using real-world commercial gNBs and UEs. To support these experiments, ARA provides Moshell-based APIs to configure Ericsson gNBs, and UE profiles also can be adjusted to match the core network.

Compared with E-5GC, Open5GS co-locates control plane and user plane on the same server in the ARA data center (green block in Fig. 4b), which connects to the gNBs via a local network of 100 Gbps fiber links and ARA switches within the ISU network. Clearly, such deployment can significantly reduce the connection establishment latency. In the next section, we will describe in detail how the transition from E-5GC to Open5GS was completed in ARA, and share the lessons learned from this real-world practice.

III. REAL-WORLD EXPERIENCE AND LESSONS FROM OPEN5GS INTEGRATION WITH ERICSSON GNB

A. Transition from Ericsson Core to Open5GS

The transition from the Ericsson core to Open5GS involves five key aspects: core server preparation, Open5GS deploy-

ment and configuration, Ericsson gNB configuration, Quetel UE configuration, and transition confirmation.

Core Server Preparation. We begin the transitioning process by provisioning a Dell R750 white-box server running Ubuntu 22.04 or later for the core network. The server is equipped with an Intel Xeon Gold 6342 CPU, 395 GB of RAM, and 100 Gbps QSFP ports. It is important to choose network interface cards (NICs) with a capacity rated above your anticipated peak network load. The server is connected to the Ericsson gNB through a network of routers and switches using high-speed fiber patch cables. Finally, we install packet-capturing tools, such as `tcpdump`, on the Open5GS server to enable efficient troubleshooting and network analysis.

Open5GS Deployment and Configuration. Open5GS can be installed from source or via a package manager, depending on the operating system. Source installing is required for operating systems such as CentOS, Fedora, and Mac OSX, while Debian/Ubuntu OS requires the package manager. The deployment of Open5GS begins with installing MongoDB to manage UE subscriber information, followed by installing Open5GS. Details of Open5GS installation using the package manager can be found in the Open5GS quickstart guide [20]. For beginners, installing Open5GS WebUI helps manage subscriber data interactively, while advanced users may prefer the command line tool [21].

To configure the Open5GS core to connect to the Ericsson gNB, the default NGAP server IP address in the `amf.yaml` file must be set to the interface on the core server that connects to the gNB. Similarly, in the `upf.yaml` file, the default GTP-U IP address needs to be changed to the same interface to establish a successful N3 connection. To ensure successful authentication and registration, the public land mobile network (PLMN) ID and tracking area code (TAC) parameters in the `amf.yaml`, and `nrf.yaml` files should be updated to match the same at the gNB. Additionally, the single network slice selection assistance information (S-NSSAI) must be set in the `nssf.yaml`. Finally, the subscriber identity module (SIM) subscriber data must be added to the unified data repository (UDR) MongoDB backend via the WebUI or the command line tool.

Ericsson gNB Configuration. The connection between the gNB and the core network is established via the N2 interface. To initiate this connection, the gNB must be pre-configured with the NGAP server IP address set in the `amf.yaml` file using MOSHELL, a managed object shell provided by Ericsson. Additionally, the PLMN ID and TAC parameter settings on the gNB must match those in Open5GS to ensure successful control and user plane signalling between the gNB and the core network.

Quetel UE Configuration. At the UE side, we configure the Quetel RG530 radios using AT commands [22] through the `minicom` serial terminal [23]. Specifically, we disable automatic modem configuration binary (MBN) selection and manually activate the `ROW_Commercial` profile to en-

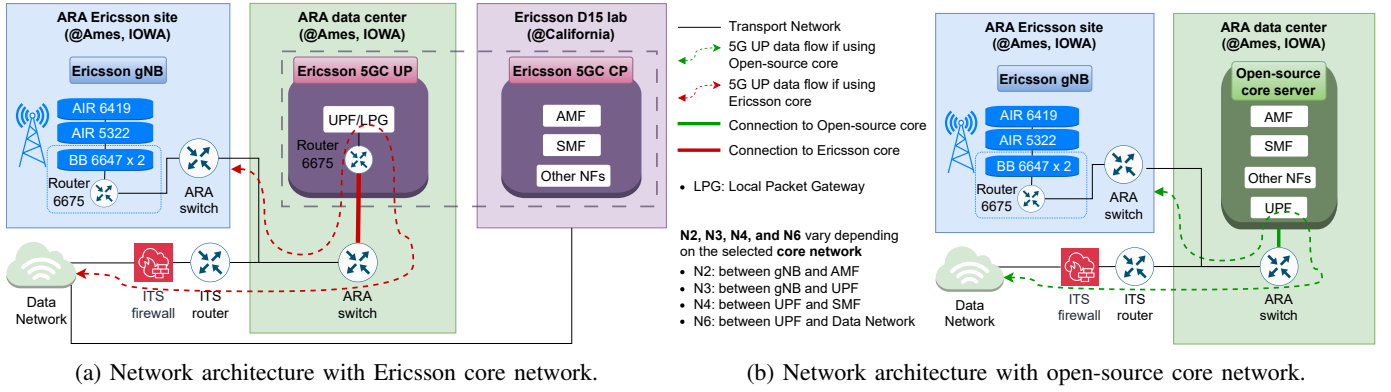


Fig. 4: 5G SA in ARA: Ericsson gNB with Ericsson core network vs. open-source core network.

sure compatibility with Open5GS. The IP version is set to IPv4/IPv6 and the Access Point Name (APN) is set to match the configurations specified in the Open5GS core database.

Integration Confirmation. Once both Open5GS and gNBs are configured properly, the gNB initiates the NGAP connection. A successful NGAP connection is indicated by the NG Setup Success message from the AMF to the gNB (visible in Open5GS logs), followed by the NG Setup Complete message from the gNB to the AMF (captured in pcap logs). Next, the Quectel UE triggers a PDU session establishment request and the subsequent receipt of the PDU Session Establishment Accept message confirms a successful PDU session establishment. From the control plane perspective, this validates a complete integration of Open5GS with the Ericsson gNB. Finally, *iPerf3* is used to assess throughput performance.

B. Lessons Learned

While the procedure above appears straightforward, we encountered several challenges during integration, as is common in real-world deployments. To our knowledge, this work is one of the first to integrate a commercial 5G RAN with an open-source 5G core and test it in a real-world setting. Below are the key lessons learned in this investigation.

Network Planning and Preparation: It is important to optimize the maximum transmission unit (MTU) on the Open5GS server interface to align with that of the transport network between the core and the gNB, to avoid packet fragmentation and consequent packet drops due to MTU mismatch. Inconsistent MTU settings may lead to NG Setup or PDU Session Setup failures, or degraded throughput. Capturing packet traces on the Open5GS server using tools such as *tcpdump* is also essential, as these logs—analyzed with *Wireshark*—can help diagnose NAS and NGAP signaling issues.

Open5GS Configuration: Network slicing should be explicitly configured with *sst* and *sd* values in both the AMF and NSSF configuration files to ensure correct slice selection and avoid NG SETUP FAILURE. The IP version must be configured consistently across the SMF, UPF and UE, as illustrated

in section III-A. Additionally, network address translation (NAT) and IP forwarding rules must be set on the open-source core network server to enable UE internet access—these rules do not persist after reboot and must be reapplied if issues like missing PDU Session Establishment Accept messages occur. We omit the secure edge proxy protocol (SEPP) NF in our 5G SA deployment as it is required only for inter-PLMN roaming, which is not relevant for isolated enterprise networks.

Ericsson gNB and Quectel UE Configuration: Before configuring the AMF IP on the gNB, it is crucial to verify network connectivity to AMF IP from the gNB. This ensures that the transport link between the gNB and AMF of the open-source core network is functioning correctly.

IV. REAL-WORLD PERFORMANCE CHARACTERIZATION

In what follows, we present performance evaluation of the Open5GS core integrated with a commercial 5G RAN. We begin by comparing the performance of Open5GS with the commercial Ericsson core. Next, we assess the performance of the Open5GS in terms of reliability. Finally, we analyze the performance differences between the simulated and real-world field UE registered with Open5GS, to reveal the importance of not relying on simulation-only evaluations.

A. Ericsson core vs. Open5GS

1) *Experimental Setup #1:* As shown in Fig. 5, we use Ericsson gNB at the ISU Curtiss Research Farm in this experiment and a Quectel UE in the nearby farm field, which is about 390 meters from the gNB, with an angular offset of about 40 degrees from the gNB’s antenna boresight. In each experiment, the field UE tries to register with the core network, request the service, establish the connection, and then run an *iPerf3* client to measure the end-to-end downlink throughput from an *iPerf3* server running on an ARA data center server. This experiment is repeated 12 times for both E-5GC and Open5GS, and the latencies and throughput measurements are recorded.

2) *Control Plane Comparison:* Open5GS exhibits lower latencies compared to the E-5GC as shown in Fig. 6, mainly due to the co-location of its control and user planes at

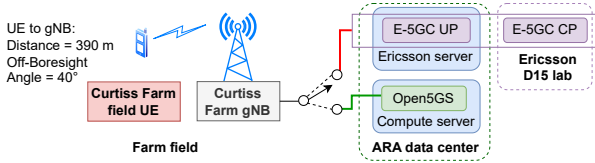


Fig. 5: Experimental Setup #1.

the ARA data center, as illustrated in Fig. 4, resulting in fewer network hops and reduced latencies, particularly for procedures requiring multiple interactions between the control plane and the user plane. In contrast, the E-5GC control plane resides remotely in California, while its user plane is located at the ARA data center. As shown in Fig. 6, Open5GS achieves average latency reductions of 19.1 ms for registration and 63.0 ms for service request procedures. However, PDU session establishment latency is only 1.7 ms lower, likely due to inefficiencies in Open5GS NFs or N6 interface setup, which can be investigated and improved further.

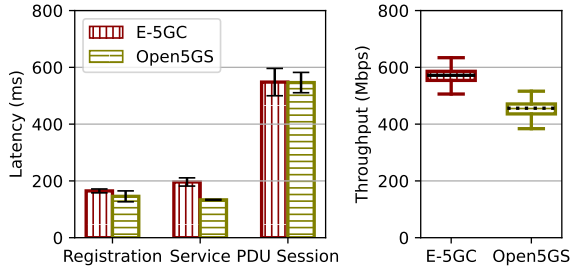


Fig. 6: Performance comparison of Open5GS with Ericsson core. (Left) Latency error bars with the 95% confidence interval. (Right) Throughput distributions box plots span the 25th–75th percentile range.

3) *User Plane Comparison:* As shown in Fig. 6, Open5GS achieves a high throughput of 445.6 Mbps, only 20.1% lower than what E-5GC enables. This far exceeds the performance requirements for eMBB services [24], which recommends user data rates to be around 100 Mbps for downlink. The throughput gap between E-5GC and Open5GS is due to the following factors: (i) Inefficient UPF algorithms in Open5GS, which can be further optimized for high-performance data transmission; (ii) Suboptimal handling of GTP sessions, particularly when out-of-order packet sequences accumulate beyond a critical threshold.

B. Open5GS system performance

This study compares the performance of a simulated UE with a commercial UE deployed in field, both connected to the same Open5GS core.

1) *Experimental Setup #2:* We use UERANSIM [25] as a simulated UE connecting to the Open5GS as shown in Fig. 7. The simulated UE is running on a compute server in ARA data center. Latencies and throughput are measured 50 times for both the simulated UE and a Quectel UE located in the field, 40 meters from the gNB and aligned with the antenna boresight (0-degree offset). We choose a different gNB and

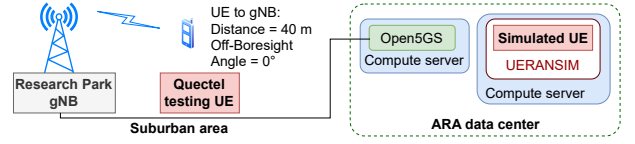


Fig. 7: Experimental Setup #2.

field UE for this setup (suburban) to evaluate performance in a different environment condition.

2) *Ericsson RAN Performance with Open5GS:* We evaluate the latencies of Registration, Service Request, and PDU Session Establishment, along with long-term throughput using Quectel UE to assess Open5GS performance. As shown in Fig. 8, Registration and Service Request latencies are both centered around 140 ms, while PDU Session Establishment is more variable, averaging around 750 ms. This difference is expected, as Registration is executed repeatedly by the UE in our experiments, omitting steps like Authentication and Security Command, involving fewer NFs and less signaling. Similarly, Service Request requires less interaction among NFs, resulting in lower and more stable latency. In contrast, PDU Session Establishment involves more NFs and extensive signaling procedures, making it higher and more sensitive to varying network conditions.

The CDF plot of throughput in Fig. 8 shows that the field UE consistently achieves high data rates. Over 90% of the measurements exceed 200 Mbps, and approximately 50% exceed 300 Mbps, demonstrating Open5GS's ability to maintain stable, high data rates in real-world deployments.

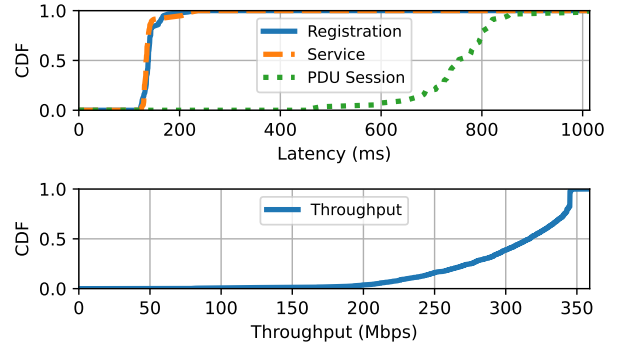


Fig. 8: Latency (top) and throughput (bottom) performance of ARA Ericsson RAN connected with Open5GS core.

3) *Field UE vs. Simulated UE:* We also conduct the same experiments using UERANSIM [25], an open-source 5G UE/RAN simulator that implements the NAS layer and parts of the RRC layer for basic connection setup and release. Instead of a physical layer, UERANSIM tunnels NAS/RRC messages via UDP sockets to the gNB. However, it lacks radio channel characteristics, limiting its realism in end-to-end evaluations.

Fig. 9 highlights significant differences in performance between the field and simulated UEs, stemming from the afore-

mentioned limitations of the simulated RAN. The simulated UE achieves substantially higher throughput (978.63 Mbps) compared to the field UE (299.1 Mbps), likely due to the absence of wireless channel perturbations in the simulated environment. Similarly, the simulated UEs exhibit much lower Registration latency (14.08 ms vs. 142.04 ms) and significantly reduced PDU Session Establishment latency (208.08 ms vs. 752.09 ms). Notably, the Service Request latency is only available for the field UE (140.53 ms), as this step is typically bypassed in the simulated setup. These disparities emphasize the limitations of simulation-only evaluations, which may underestimate signaling delays and overestimate throughput. Hence, incorporating real-world measurements is critical for a comprehensive assessment of open-source core network performance.

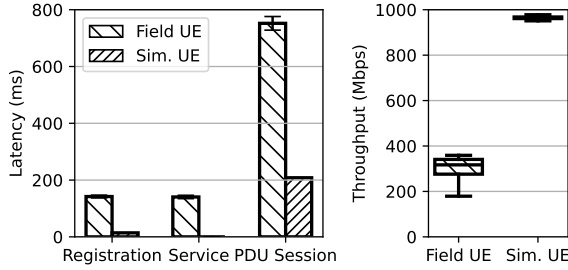


Fig. 9: Comparison of latency and throughput between field Quectel UE and simulated UE. (Left): Latency error bars with the 95% confidence interval. (Right): Throughput distributions box plots span the 25th–75th percentile range.

Further, the average throughput observed on the field UE in Setup #1 is 450 Mbps, while that of Setup #2 is only 300 Mbps, despite Setup #2 experiencing better radio conditions during the experiment. This discrepancy may be attributed to suboptimal or inconsistent MTU settings along the end-to-end GTP-U tunnel path. As discussed in Section III-B, MTU mismatches can lead to packet fragmentation, retransmissions, and tunneling inefficiencies, ultimately degrading throughput performance. Future work may investigate these discrepancies and contribute useful insights for developing more robust and reliable open-source core networks.

V. CONCLUDING REMARKS

We have demonstrated the first field-scale integration of the Open5GS open-source core with a commercial Ericsson massive MIMO RAN in ARA [13]. Extensive field measurements demonstrate the feasibility and potential of open-source core solutions like Open5GS as a practical path towards flexible, cost-effective 5G-and-beyond cellular networks, particularly for rural broadband where reducing the costs by a factor of 10 as compared to urban counterpart is important [13]. Interesting future directions include tightening the performance gap between open-source and commercial cores, characterizing additional 3GPP protocol procedures, such as handovers, session management, and testing different QoS scenarios with multiple UEs, as well as comparing emerging releases of

open-source core implementations. Besides systems research insights gained from this study, the resulting testbed of integrated Open5GS and Ericsson systems enables research on network slicing, edge computing, and energy-efficient operations, and it empowers the broad research community to validate ideas on a live, carrier-grade environment, accelerating open-source innovations from labs to fields.

VI. ACKNOWLEDGMENTS

This work is supported in part by the NSF awards 2130889, 2112606, 2212573, 2229654, and 2232461, NIFA award 2021-67021-33775, and the PAWR Industry Consortium. We also thank Ericsson for their support on this work.

REFERENCES

- [1] Open5GS, “A C-language open source implementation for 5G core and EPC,” <https://github.com/open5gs/open5gs>.
- [2] “OpenAirInterface5G core network,” <https://gitlab.eurecom.fr/oai/cn5g/oai-cn5g-fed>, 2024.
- [3] “SD-Core—Aether Project,” <https://aetherproject.org/sd-core/>.
- [4] Free5GC, “An open source 5G core network based on 3GPP Rel15,” <https://github.com/free5gc/free5gc>.
- [5] M. Barbosa *et al.*, “Open-source 5G core platforms: A low-cost solution and performance evaluation,” *arXiv:2412.21162*, Dec. 2024.
- [6] R. Reddy *et al.*, “Open source 5G core network implementations: A qualitative and quantitative analysis,” in *2023 IEEE International Black Sea Conference on Communications and Networking (BlackSeaCom)*, Nov. 2023, pp. 253–258.
- [7] Phan *et al.*, “Building a 5G core network testbed: Open-source solutions, lessons learned, and research directions,” in *2024 ICOIN*, Jul. 2024, pp. 641–646.
- [8] Amini *et al.*, “A comparative analysis of open-source software in an E2E 5G standalone platform,” in *2024 IEEE WCNC*, Jul. 2024.
- [9] “srsRAN Project—Open Source RAN,” <https://www.srslte.com/>.
- [10] “OpenAirInterface—5G software alliance for democratising wireless innovation,” <https://openairinterface.org/>.
- [11] J. E. Håkegård *et al.*, “Performance evaluation of an open source implementation of a 5G standalone platform,” *IEEE Access*, vol. 12, pp. 25 809–25 819, Feb. 2024.
- [12] PAWR, “PAWR—Platforms for Advanced Wireless Research,” <https://www.advancedwireless.org>, 2025.
- [13] T. U. Islam *et al.*, “Design and implementation of ARA wireless living lab for rural broadband and applications,” *Computer Networks*, vol. 263, May 2025.
- [14] Ericsson, “Ericsson AIR 6419—radio system,” <https://www.ericsson.com/en/ran>.
- [15] Quectel, “RG530F Series—5G Sub-6 GHz module,” <https://www.quectel.com/product/5g-rg530f-series/>.
- [16] “3GPP TS 23.501: 5G System Architecture,” 3GPP, Tech. Rep., 2022.
- [17] “3GPP TS 38.413: NGAP,” 3GPP, Tech. Rep., 2022.
- [18] “3GPP TS 24.301: NAS protocol for EPS,” 3GPP, Tech. Rep., 2022.
- [19] ARA Project Team, “ARA wireless portal: Access to the ARA wireless living lab,” <https://portal.arawireless.org/>.
- [20] S. Lee, “Open5GS quickstart guide,” <https://open5gs.org/open5gs/docs/guide/01-quickstart/>.
- [21] Open5GS, “Database command line tool,” <https://github.com/open5gs/open5gs/blob/main/misc/db/open5gs-dbcctl>.
- [22] Quectel Wireless Solutions, *AT Commands Manual*, https://www.quectel.com/download/quectel_rg50xqrm5xxq_series_at_commands_manual_v1-2/.
- [23] “minicom—Linux manual page,” <https://man7.org/linux/man-pages/man1/minicom.1.html>.
- [24] ITU, “Minimum Requirements for IMT-2020 Radio Interface(s),” ITU-R, Tech. Rep. M.2410-0, 2017.
- [25] A. Gungor, “UERANSIM: 5G core and RAN simulation platform,” <https://github.com/aligungr/UERANSIM>, 2024.