Influence of Core Decentralization on the Performance of a Private 5G Network

Ednaldo Júnior[1], Higor Dantas[1], Francisco Silva[1], Thiago Moraes[3], Ruan Gomes[2], João Brunet[1]

[1]Unidade Acadêmica de Sistemas e Computação (UFCG) – Paraíba – Brazil

[2]Unidade Acadêmica de Informática (IFPB) – Paraíba – Brasil

[3]IBM – Rio de Janeiro – Brasil

ednaldo.junior@lsd.ufcg.edu.br, higor.dantas@ccc.ufcg.edu.br

francisco.pereira.silva.filho@ccc.ufcg.edu.br, tgmoraes@br.ibm.com

ruan.gomes@ifpb.edu.br, joao.arthur@computacao.ufcg.edu.br

Abstract. *With the increasing relevance of private 5G networks and the development of open-source solutions, the performance evaluation of this architecture under various conditions is becoming increasingly necessary. In this context, the IBM Flextronics project at the Universidade Federal de Campina Grande (UFCG) implemented a private 5G network to conduct experiments. Given the need for testing across the infrastructures of different buildings and networks, this work proposes to evaluate the performance of a private 5G network using an Open5GS core, with one gNodeB installed on-premises in a building in close proximity to the core, and another gNodeB deployed remotely in other building at the university. The goal is to assess how decentralization of the core influences performance in terms of bitrate and latency.*

1. Introduction

Over time, mobile communication systems progressed through third- and fourth-generation (LTE) technologies, and, starting in late 2018, fifth-generation (5G) networks have become increasingly prominent and important in both research and commercial domains [Nester 2025]. Generally, the technologies employed by companies are either "black boxes", with their operational details concealed, or monolithic, featuring components integrated into a single proprietary solution. To shift this paradigm, organizations such as the O-RAN Alliance have published studies on open-source 5G architectures. These initiatives aim to broaden the debate among stakeholders and democratize research in the field, fostering collaborative and open-source evolution [Polese et al. 2023].

The 5G Core (5GC) is based on a Service-Based Architecture (SBA), which decomposes network functions (NFs) into independent microservices. This architecture facilitates the adoption of open-source implementations, such as OpenAirInterface (OAI), free5GC, and Open5GS, by allowing NFs to be modular and scalable within virtualized environments [Zieba et al. 2024]. Combined with the use of network function virtualization and containerization technologies, this architecture ensures unprecedented deployment flexibility, allowing control-plane and user-plane functions to operate across distributed cloud environments. This scenario not only fosters the development and adoption of open-source 5GC but also enables researchers and organizations to experiment with new functionalities and optimize network parameters in an agile and scalable fashion [Bonati and others 2020].

In this context, this paper describes the deployment of a private 5G network as part of the IBM 5G Flextronics innovation project, which aims to interconnect multiple commercial 5G Radio Units (RUs), located in various laboratories at the Universidade Federal de Campina Grande (UFCG), with an open-source core solution. The objective is to investigate the factors that influence the performance of the 5G network, encompassing aspects of the Radio Access Network (RAN), as well as the transport network used in the deployment, with a focus on evaluating the influence of core decentralization on the performance of the network. The results show that placing gNodeB in close proximity to the core network resulted in a significant increase in throughput, with downlink bitrates around 100 Mbps higher than those observed in the decentralized scenario. For the uplink, the local deployment achieved bitrates of about 20 Mbps higher. This local configuration further demonstrated lower latency, but with greater variability and jitter. In contrast, in the decentralized scenario, the overall performance metrics decreased substantially, but the system exhibited improved stability.

The remainder of this paper is organized as follows. Section 2 surveys related literature; Section 3 describes the testbed and the scenario considered in the experiment; Section 4 describes the experimental results; and Section 5 presents the conclusions and ideas for future work.

2. Related Works

Some studies in the literature have investigated comparative performance assessments of alternative 5G deployment architectures. In [Mukute et al. 2024], the authors compare the performance of a 5G network using Open5GS as the core in two deployment models: a conventional, non-virtualized setup on physical hardware and a containerized core orchestrated with Kubernetes. Their objective is to determine whether containerization influences performance indicators such as latency and throughput. The experiments evaluate UE registration and deregistration procedures in the control plane. The testbed comprises six servers hosting Open5GS for both scenarios and a traffic generator emulating multiple UEs sending requests, enabling measurement of both end-to-end processing time and the number of requests handled per time unit. The results show that, under moderate traffic, both deployments achieve comparable performance; however, beyond 300 devices, the Kubernetes-based setup experiences a 7% performance drop. While [Mukute et al. 2024] investigates how software infrastructure affects network performance, our work examines the influence of physical separation between radio and core in a private 5G testbed.

In [Merces et al. 2025], the performance of a private 5G network, consisting of a core based on Open5GS and a commercial RAN, is evaluated under different hardware configurations for the core. The findings indicate that in the scenario studied, network performance is more strongly correlated with CPU clock frequency than with the number of cores. In particular, configurations with higher CPU frequency yielded downlink throughput improvements exceeding 170%. This evidence supports the conclusion that, under almost ideal conditions, such as deployments with local fronthaul, the primary bottleneck is typically located in the 5G core. In contrast, when the gNodeB is geographically distant from the core, additional factors affect performance, especially the latency and variability introduced by the transport segment between the RAN and the core. In this case, the bottleneck becomes a joint effect of network and processing constraints, potentially causing higher throughput fluctuations and higher end-to-end latency. While [Merces et al. 2025] concentrates on how hardware characteristics influence 5G core performance, our work extends these results by examining how topology and physical placement of the core affect network behavior, comparing scenarios in which the gNodeB is co-located with the core and scenarios in which they reside in a distinct sub-network.

In [Singh et al. 2024], an analysis of the performance of a 5G network in an indoor environment is presented, using a testbed composed of an Amarisoft Callbox core, a gNodeB, UEs and a Remote Radio Read (RRH). Relevant metrics such as end-to-end latency, throughput, CPU utilization, and energy consumption were evaluated. The results indicate that latency can be significantly reduced by properly adjusting the scheduling parameters. It was also observed that the integration of an RRH introduces a slight increase in latency compared to a more centralized architecture, due to the additional communication and processing overhead. However, this distributed approach provides improvements in coverage and signal quality. Additionally, the study demonstrates that increasing bandwidth leads to higher data rates, resulting in increased CPU utilization at the gNodeB and higher energy consumption, both in the central unit and in the radio unit. Furthermore, it is observed that downlink traffic achieves higher throughput, whereas uplink traffic tends to require more computational resources. Although [Singh et al. 2024] aimed to perform a performance analysis of indoor 5G networks, this work explores a network performance scenario involving decentralization between the network core and the gNodeB. However, the findings of the study [Singh et al. 2024] are particularly relevant to the scenario addressed here, as they highlight the trade-off between latency and coverage, as well as the impacts of the functional distribution of the network on resource consumption and overall system performance.

The paper [Siegmund et al. 2026] investigates the performance of the User Plane Function (UPF) in 5G networks. Its main goal is to determine how well existing UPF implementations satisfy the QoS requirements of applications that concurrently require low latency, low jitter, and high throughput. To this end, the authors built a testbed capable of emulating realistic large-scale network deployments, and four UPF implementations were evaluated. The first implementation is free5GC, which is based on the Linux kernel. The second, eUPF, leverages eBPF/XDP for in-kernel packet processing, thereby minimizing overhead. The third, UPG-VPP, executes packet processing in the user space with direct access to the network interface card. The fourth is a hardware-accelerated design that integrates a programmable P4 switch with an FPGA that handles queueing and QoS enforcement. The hardware-accelerated UPF provides the highest performance, achieving microsecond-level latency, deterministic forwarding, and stable operation with up to 10000 concurrent PDU sessions. Among software-based implementations, UPG-VPP achieves the lowest latency among them and the highest throughput. eUPF offers a balanced trade-off, since it maintains high data rates and implements QoS shaping per session with more accurate rate control than free5GC. free5GC shows the lowest performance, with reduced throughput, higher and more variable latency, and bursty traffic, although it is the simplest to deploy. [Siegmund et al. 2026] provides a good foundation for the influence of UPF on network performance. In our paper, we used a software-based 5GC, but we evaluated the influence of other aspects that impact network performance, especially when the gNodeB is deployed remotely in relation to the core.

3. Study Case Description

In this section, a case study is presented to evaluate the performance of a private 5G network under two distinct scenarios: one in which the gNodeB is deployed close to the network core and another in which the gNodeB is deployed in a remote site and is connected through an external network infrastructure. This infrastructure was developed within the scope of a joint project between IBM Flextronics and the UFCG. The experimental environment includes a server responsible for running the network core based on Open5GS, as well as two gNodeBs (small cells) deployed in two different laboratories at the university.

The first gNodeB was installed in the Laboratory of Distributed Systems (LSD Lab), where the server that runs the network core is also located, and the second gNodeB was installed in the Software Practices Laboratory (SPLab). Although they are only about 60 meters apart, the buildings are integrated into distinct institutional networks. Consequently, the network topology was arranged so that gNodeB placed in the LSD Lab remained close to the server running the core, thus defining the local scenario. In contrast, the unit deployed in SPLab required the configuration of an IPSec-based VPN tunnel to interface with Open5GS, defining the remote scenario under investigation. For the experimental evaluation, a 5G-capable smartphone was used as user equipment (iPhone 13).

One distinctive aspect of this case study is that the 5G network was deployed in a real research environment, with components distributed between different laboratories and integrated into the existing university infrastructure. This setting provides an opportunity to observe the operation of a mobile network in conditions close to those found in real deployments, where physical, organizational, and infrastructure limitations may influence system behavior.

Figure 1 shows the project topology with the on-premises 5GC, with the internal topology of each building hidden for security reasons. It is important to note that Figure 1 shows 4 gNodeBs, however, only two were used in the tests described in this paper. The tunnel connection to the gNodeBs occurs between the edge device of each laboratory and the server named `UFCG-IBM-SERVER`. Once the gNodeBs can communicate with the server, they can register with the Network Orchestration System (NOS), which acts as the controller of the gNodeBs and is proprietary software developed by Mavenir. After that, they radiate the signal and connect the UEs to the 5G network.

3.1. Testbed

The testbed used in this work consists of a 5G network infrastructure composed of two small cells, model Mavenir E511. These devices support multiple frequency bands, including n 41 (2496-2690 MHz), n78 (3300-3800 MHz) and n79 (4400-5000 MHz), with band n78 being used in the experiments. The transmission power (EIRP) is 30 dBm. The equipment supports typical 5G NR channel bandwidths (up to 100 MHz) and implements a 2T2R (2 transmit and 2 receive antennas) Multiple-Input, Multiple-Output (MIMO) configuration.

The 5GC was implemented using the Open5GS software, running on a virtualized infrastructure on a dedicated physical server. The server used is a Dell PowerEdge R450,

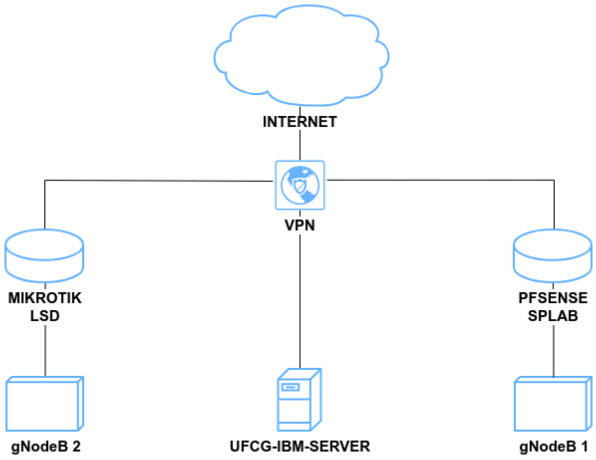

Figure 1. Topology diagram

equipped with an Intel Xeon Silver 4310 processor (2.10 GHz), 64GB of RAM, and 2TB of storage, operating with the CentOS 7 operating system. On this server, three virtual machines were instantiated: one for Open5GS (responsible for executing the 5GC), one for NOS, and one for the Small Cells Analytics Server (SCAS), responsible for collecting metrics and logs. In the case of Open5GS, its functions were implemented in Docker containers, covering both the control plane and the user plane. The Open5GS VM was created with Ubuntu 24.04, 4GB of RAM, and 40GB of storage; the NOS VM was configured with the CentOS 7 operating system, 20GB of RAM, and 300GB of storage; and the SCAS VM was also created with the CentOS 7 operating system, 12GB of RAM, and 500GB of storage.

Furthermore, a specific modification was required in the Open5GS source code[1] to ensure compatibility with the Mavenir small cell. Specifically, the maximum allowable size for the GPRS tunneling protocol (GTP) extension header was increased from 4 to 8 bytes, as the Mavenir equipment transmits an additional 4 bytes within the GTP header.

The UE used in the experiments was an Apple iPhone 13 smartphone, chosen for its compatibility with the frequency bands used, especially band n78. The device supports 5G operation and features hardware composed of the Apple A15 Bionic chip and 4GB of RAM. For the tests, the *applications iPerf 3 Wifi Speed testhttps*[2] and *Ping - network utilityhttps*[3] were used.

3.2. Experiment Scenario

Physically, the LSD Lab and SPLab buildings are located in the UFCG and are interconnected through the Information Technology Service (STI) network. Despite this, traffic is tunneled through VPN using IPSec to connect the private networks used in the configurations, in addition to encrypting the data. Both the server and the network gateways are physically located in the LSD Lab. Within this environment, there is a dedicated subnet designated as the “gNB network”, in which each gNodeB is assigned a unique IP address. The gNodeB deployed in SPLab (gNodeB 1 in Figure 1) also has an IP address, but this

---

[1] Updated source code available on https://github.com/Francisco-xiq/open5gs-gtp2-fix

[2] ://apps.apple.com/br/app/iPerf 3 Wifi Speed test/id1462260546

[3] ://apps.apple.com/br/app/ping-network-utility/id576773404

address belongs to a /30 private subnet, specifically configured to establish a point-to-point connection with the PFSense device, which functions as the edge router of SPLab.

There is a site-to-site VPN tunnel configured between the LSD and SPLab laboratories. The purpose of this tunnel is to provide connectivity between the LSD networks, where the other gNodeB and the N2 and N3 interfaces of the 5GC are configured, and the private network configured for the gNodeB located in the SPLab.

The service used to establish this tunnel was StrongSwan, which uses the IPSec and IKEv2 protocol stack, providing efficient data encryption and performing authentication via PSK (Pre-Shared Key). According to Figure 2, which illustrates the data flow from the UE connected to the gNodeB in SPLab, the data start from the UE to the gNodeB (1) and are then forwarded to the SPLab edge router (2), which is one of the VPN peers. The data is then encrypted (3) and transmitted to the other VPN peer (4), where they are decrypted and forwarded to the LSD edge router (5), and finally transmitted to the internet (6). If we consider the scenario of the UE connected to the gNodeB in the LSD Lab, flows (2), (3), and (4) do not exist, going only from the UE directly to the UFCG-IBM-SERVER.

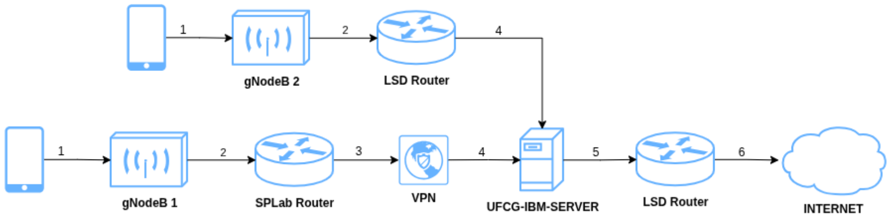

Figure 2. Data flow diagram

4. Results

This section presents the results of a series of conducted tests. In particular, 20 tests were executed for each scenario, comprising 10 upload and 10 download measurements. Figure 3 shows the performance disparity observed when the UE was attached to the remote gNodeB (SPLab) versus the local gNodeB (LSD Lab), based on a comparison of their mean throughput. A discrepancy is observed between the two environments. The average download throughput in the LSD Lab exceeds that of the SPLab gNodeB by more than 100 Mbps. A similar behavior is seen in the upload results, which present a difference greater than 20 Mbps between the two evaluated scenarios. Analysis of environments and collected measurements suggests that the IPsec tunnel is the primary factor causing performance degradation. No significant environmental differences were identified that could explain the variations in the wireless link between the UE and the gNodeB, and in both configurations the UE maintained line-of-sight to the gNodeB at a separation of less than 3 meters.

In the upload experiments, the local scenario exhibited greater stability. Its interquartile range (IQR) lies between 65 and 75 Mbps, and the corresponding boxplot appears to be very compact. In contrast, the uplink bitrate in the decentralized scenario shows markedly higher variability. The values fluctuate substantially, roughly between 25 and 60 Mbps, which implies that the median is a less reliable descriptor of the overall distribution. A comparable behavior is observed for the download bitrate. Although

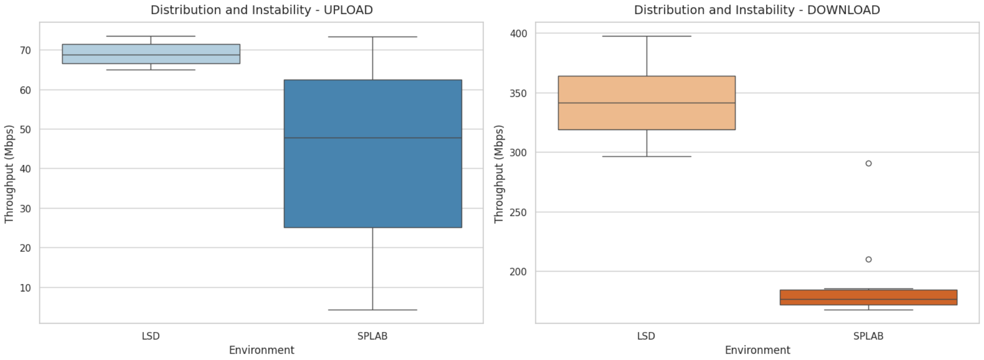

Figure 3. Comparison of bitrates in the two evaluated scenarios.

the boxplot for the measurements obtained in the SPLab (decentralized scenario) is also relatively compact, indicating improved stability compared to its upload results, it still reveals clearly inferior overall performance relative to the measurements from the tests conducted in the LSD Lab (local scenario).

To assess latency, a 15-minute ping test was performed between the iPhone and the Core network machine. The measurements collected revealed different behaviors in the two scenarios, as illustrated in Figure 4. Although the tests conducted in the LSD Lab (local scenario) show better performance in terms of minimum latency, with values as low as 2.848 ms, the network suffers from considerable jitter, with an estimated value of 5.436 ms and a standard deviation of 5.852 ms.

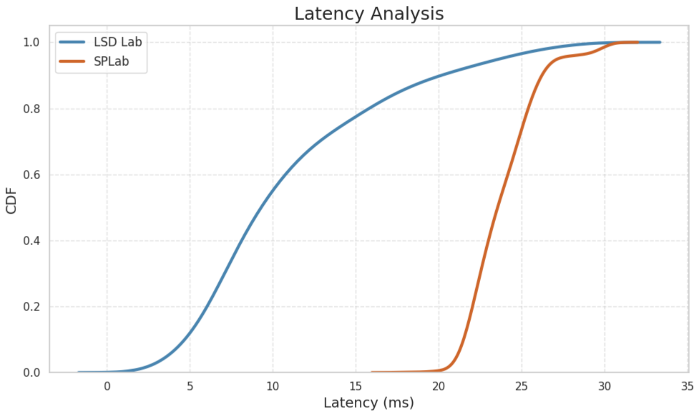

Figure 4. Comparison of network latency between the two evaluated scenarios.

In contrast, the tests conducted at SPLab (decentralized scenario) show a higher average latency of about 23.82 ms and a minimum latency of 17.5 ms. These results suggest that, in the local scenario, the network can provide faster responses, but with lower stability. In contrast, the tests performed in the decentralized scenario demonstrate a significantly higher latency but less variability, as indicated by its lower standard deviation of 1.989 ms.

5. Conclusion and Future Work

The experimental evaluation described in this paper demonstrated that physically separating the 5G core from the gNodeB and interconnecting them via an IPSec site-to-site

VPN significantly degrades network performance. When gNodeB is located in the remote SPLab (decentralized scenario), the additional encapsulation, encryption, and multi-hop forwarding introduced by the VPN tunnel increase latency by approximately 79.89% and reduce throughput by approximately 44.60% in downlink and 44.60% in uplink, compared to the local scenario. These results confirm that core decentralization, while offering architectural flexibility, can become a performance bottleneck in private 5G deployments unless the transport network is carefully engineered. Keeping the RAN and the core in close physical proximity, or optimizing the VPN link for low-latency, high-throughput operation, is therefore advisable, as well as the deployment of multiple UPFs.

In future work, additional experiments could involve alternative open-source core implementations (e.g. free5GC) and hardware acceleration for the user-plane function, as well as exploratory multi-UPF deployments.

Acknowledgments

This work was funded by FIT - Technology Institute/IBM (EMBRAPII PCEE-2310.0243). The authors also thank CNPq (307108/2025-2).

References

Bonati, L. and others, M. P. (2020). Open, programmable, and virtualized 5g networks: State-of-the-art and the road ahead. *Computer Networks*, 182:107516.

Merces, J., Silva, R., et al. (2025). Core 5g open source com ran comercial: Uma avaliação de desempenho em rede 5g privativa. In *XLIII Simpósio Brasileiro de Telecomunicações e Processamento de Sinais*, pages 1–5.

Mukute, T., Brito, M., et al. (2024). A brief performance comparison of bare-metal and kubernetes deployments for 5g core control plane network functions using open5gs. In *The 2024 Southern Africa Telecommunication Networks and Applications Conference (SATNAC 2024)*, pages 1–7.

Nester, R. (2025). 5g technology market size, share, trends analysis 2025-2035. `https://www.researchnester.com/reports/5g-technology-market/2257`. Accessed on September 21, 2025 at 6:50:27 PM.

Polese, M., Bonati, L., et al. (2023). Understanding o-ran: Architecture, interfaces, algorithms, security, and research challenges. *IEEE Communications Surveys & Tutorials*, 25(2):1376–1411.

Siegmund, F., Kundel, R., et al. (2026). User plane performance in beyond 5g networks: Comprehensive analysis and evaluation. *Computer Communications*, 247:108397.

Singh, B. C., Diaz, R., and Shetty, S. (2024). Evaluating the performance of 5g nr in indoor environments: An experimental study. *Procedia Computer Science*, 241:16–23. 19th International Conference on Future Networks and Communications/ 21th International Conference on Mobile Systems and Pervasive Computing/14th International Conference on Sustainable Energy Information Technology.

Zieba, M., Natkaniec, M., and Borylo, P. (2024). Cloud-enabled deployment of 5g core network with analytics features. *Applied Sciences*, 14(16).