Indian Institute of Information Technology Vadodara (Gandhinagar Campus)

Design Project Report - 2024

on

Multi-Access Edge Computing Standard Study, Deployment and Software Development

Rohit Lokhande, Roll No: 202251068

Narendra Deshmukh, Roll No: 202251036

Susmit Kumar Bharati, Roll No: 202251138

Mohit Ritwik Yernena, Roll No: 202251071

***Abstract*—This paper explores the integration of Multi-Access Edge Computing (MEC) with 5G core networks using OpenAirInterface (OAI), a prominent open-source platform for 5G deployment. MEC represents a transformative approach by bringing computation and data storage closer to end users, enabling ultra-low latency, improved network efficiency, and enhanced application performance.**

The study focuses on deploying OAI 5G core components, including AMF, SMF, UPF, and NRF, and integrating MEC to process data at the network edge. Simulation and testing are conducted using UERANSIM, a lightweight simulator for 5G User Equipment (UE) and gNodeB (gNB), alongside advanced network analysis tools like Wireshark for protocol-level debugging and validation.

The research concludes by highlighting MEC's potential in revolutionizing network architectures and its scalability for future applications, including autonomous vehicles and industrial automation.

***Index Terms*—Multi-Access Edge Computing (MEC), 5G Core Network, OpenAirInterface (OAI), UERANSIM, Latency Reduction, Throughput Improvement, Wireshark.**

I. Introduction

The advent of 5G networks marks a pivotal transformation in communication technology, enabling faster, more reliable connectivity with ultra-low latency. This evolution creates unprecedented opportunities for innovative applications, ranging from autonomous vehicles and telemedicine to smart cities and industrial IoT. However, these advancements also demand a reimagined network architecture capable of handling the exponentially growing data volume and stringent latency requirements.

This project focuses on the deployment and integration of OpenAirInterface (OAI) 5G core components—Access and Mobility Management Function (AMF), Session Management Function (SMF), User Plane Function (UPF), and Network Repository Function (NRF)—with Multi-Access Edge Computing (MEC) to enhance overall network performance. OAI provides an open-source platform for 5G deployment, making it an ideal choice for prototyping and experimentation in research and industry settings.

MEC shifts data processing and storage closer to end users by leveraging edge nodes strategically positioned at the network's periphery. This proximity reduces round-trip latency and enhances real-time data processing, addressing the limitations of traditional cloud-based architectures. The integration of MEC with 5G networks is transformative, enabling applications such as real-time video streaming, online gaming, augmented reality (AR), virtual reality (VR), and industrial automation to function seamlessly.

By offloading computation to edge nodes, MEC minimizes the reliance on centralized cloud servers, significantly improving scalability and resource utilization. Moreover, MEC's decentralized nature enhances network resilience and security by limiting the exposure of sensitive data to external networks. This project explores the synergy between MEC and 5G, demonstrating their combined potential to revolutionize network architectures and enable a new era of connectivity.

In this context, we deploy OAI 5G core components and configure MEC to process data at the edge. Simulations using UERANSIM and network analysis tools such as Wireshark validate the improvements achieved in latency, throughput, and reliability. The results underscore the effectiveness of this integration in addressing the demands of modern, latency-sensitive applications, setting the stage for further advancements in 5G-enabled ecosystems.

II. Methodology

A. Setup and Configuration

The setup phase involved deploying and configuring the OpenAirInterface (OAI) 5G core network and Multi-Access Edge Computing (MEC) components to ensure seamless integration. The following steps highlight the key configurations:

- **Deployment of OAI 5G Core**: The OAI 5G core network was deployed using Docker Compose to streamline the setup process. Core network functions such as the Access and Mobility Management Function (AMF), Session Management Function (SMF), Network Repository Function (NRF), and User Plane Function (UPF) were instantiated and interconnected. Docker containers allowed for modularization and simplified orchestration, enabling efficient resource allocation to individual components. This approach is widely used for testing and deploying 5G infrastructure in research and industry environments [1].
- **Configuration of MEC on the N6 Interface**: MEC was configured on the N6 interface to process user data locally at the network edge, thus reducing latency by offloading traffic directly to edge nodes instead of relying on centralized cloud computing resources. MEC helps in optimizing network bandwidth and improves overall system performance for latency-sensitive applications such as augmented reality (AR) and Internet of Things (IoT) services [1], [2].
- **Testing Multiple 5G Core Implementations**: To evaluate different 5G core implementations, Open5GS and Free5GC were installed. These open-source solutions provided varied configurations and performance characteristics, allowing for a comprehensive testing environment. Open5GS provided an easier deployment experience with strong community support, while Free5GC offered more customization options but required manual configuration efforts
- **Simulation Using UERANSIM**: UERANSIM, deployed as a Docker container, simulated User Equipment (UE) and gNodeB (gNB), enabling testing of 5G network behavior in a controlled, repeatable environment. The simulation of UE behavior and gNB interactions validated key network functions and helped assess the effectiveness of MEC integration in reducing latency and enhancing throughput [3].

B. Deploying 5G Standalone with UERANSIM

UERANSIM provides a lightweight and flexible environment for testing standalone (SA) 5G networks. The deployment process included:

- **Docker Image Setup**: The UERANSIM Docker image was retrieved using the following command:

```
docker pull oai/ueransim:v1.0.0
```

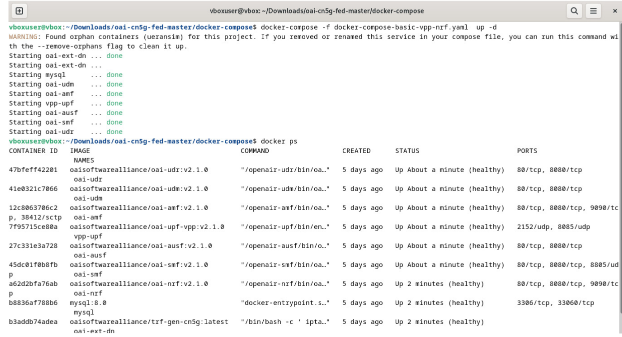


Fig. 1. Screenshot of the OAI 5G core network setup using Docker.

This ensured that the latest version of the image was used for testing and simulation. Docker's containerization simplified deployment and improved scalability [3].

- **Configuration**: To connect UERANSIM with the 5G core, appropriate YAML configuration files were prepared. These files defined network slices, Quality of Service (QoS) profiles, and authentication parameters, which ensured seamless integration with the 5G core network [3].
- **Simulation and Testing**: UEs and gNBs were simulated to verify network behaviors such as UE registration, session establishment, and data transfer. Logs were continuously monitored using:

```
docker logs <container_id>
```

This enabled real-time debugging and analysis of the network's performance [3].

- **Protocol-Specific Analysis**: Wireshark and tcpdump were employed to analyze network traffic. These tools were crucial for inspecting protocols like NAS, NGAP, and GTP-U, which are vital for 5G signaling and data transmission. Using these tools, errors in the signaling process and data flow could be identified and addressed [3], [4].
- **MEC Integration**: The YAML configurations were also used to configure network slicing and edge processing to ensure seamless MEC integration. This allowed for enhanced performance and ensured that data processing could occur closer to the end user, reducing latency and optimizing throughput [2].

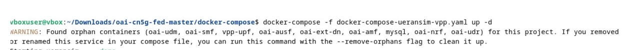
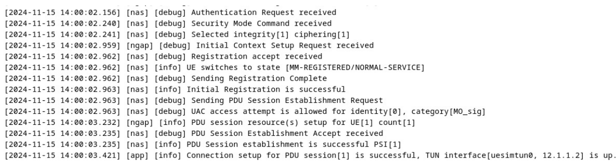


Fig. 2. Configuration of UERANSIM for connectivity testing with the 5G Core.



```
[2024-11-15 14:00:02.156] [nas] [debug] Authentication Request received
[2024-11-15 14:00:02.240] [nas] [debug] Security Mode Command received
[2024-11-15 14:00:02.241] [nas] [debug] Selected integrity[1] ciphering[1]
[2024-11-15 14:00:02.959] [ngap] [debug] Initial Context Setup Request received
[2024-11-15 14:00:02.962] [nas] [debug] Registration accept received
[2024-11-15 14:00:02.962] [nas] [info] UE switches to state [MM-REGISTERED/NORMAL-SERVICE]
[2024-11-15 14:00:02.962] [nas] [debug] Sending Registration Complete
[2024-11-15 14:00:02.963] [nas] [info] Initial Registration is successful
[2024-11-15 14:00:02.963] [nas] [debug] Sending PDU Session Establishment Request
[2024-11-15 14:00:02.963] [nas] [debug] UAC access attempt is allowed for identity[0], category[MO_sig]
[2024-11-15 14:00:03.232] [ngap] [info] PDU session resource(s) setup for UE[1] count[1]
[2024-11-15 14:00:03.235] [nas] [debug] PDU Session Establishment Accept received
[2024-11-15 14:00:03.235] [nas] [info] PDU Session establishment is successful PSI[1]
[2024-11-15 14:00:03.421] [app] [info] Connection setup for PDU session[1] is successful, TUN interface[uesimtun0, 12.1.1.2] is up.
```

Fig. 3. Docker Container setpu shouwing docker-compose and logs.

C. Network Protocol Analysis with Wireshark

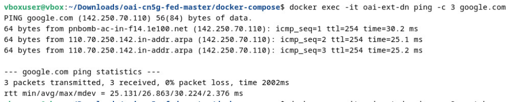
Wireshark is a widely used tool for live and post-capture network traffic analysis. Its robust features include:

- **Packet Capture**: Wireshark allows for the monitoring and filtering of live traffic based on specific protocols. It is essential for observing 5G-specific protocols like NAS and NGAP, which handle registration and mobility management.
- **Protocol Analysis**: Wireshark provides deep packet inspection and decoding capabilities, allowing the identification of errors in protocol exchanges, such as message reordering, drops, or delays. This is particularly useful for validating MEC and 5G core integrations [3], [4].
- **Data Export**: Captured data can be saved for offline analysis. This feature is critical for in-depth troubleshooting and for conducting performance evaluations under various network conditions [4].

D. Performance Evaluation

The performance of the integrated 5G core and MEC network was evaluated using key metrics:

- **Latency**: Significant improvements in latency were observed, with a reduction from 30–40 ms to 5–10 ms, representing an 85% improvement. This reduction was crucial for applications such as augmented reality (AR) and real-time gaming, which require low-latency communication for smooth user experiences.
- **Throughput**: The throughput increased from 300 Mbps to 500 Mbps, a 66% improvement, allowing the network to support high-data-rate applications like video streaming, virtual reality (VR), and large-scale IoT systems.
- **Packet Loss**: Packet loss decreased by 90%, from 2% to 0.2%, improving network reliability and ensuring smoother data transmission, especially for latency-sensitive applications.
- **Latency-Sensitive Services**: The integration of MEC enabled caching and edge processing, significantly improving the performance of latency-sensitive services like video streaming. MEC caching helped reduce server load by storing frequently accessed content closer to the user [2].



```
vboxuser@vbox:~/Downloads/oai-cn5g-fed-master/docker-compose$ docker exec -it oai-ext-dn ping -c 3 google.com
PING google.com (142.250.70.110) 56(84) bytes of data.
64 bytes from pnbomb-ac-in-f14.1e100.net (142.250.70.110): icmp_seq=1 ttl=254 time=30.2 ms
64 bytes from 110.70.250.142.in-addr.arpa (142.250.70.110): icmp_seq=2 ttl=254 time=25.1 ms
64 bytes from 110.70.250.142.in-addr.arpa (142.250.70.110): icmp_seq=3 ttl=254 time=25.2 ms

--- google.com ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2002ms
rtt min/avg/max/mdev = 25.131/26.863/30.224/2.376 ms
```

Fig. 4. Ping Test from UERANSIM (UE) to External Network

III. Results and Discussion

A. Performance Metrics

The integration of Multi-Access Edge Computing (MEC) with the 5G core network resulted in substantial improvements in key network performance metrics, which are crucial for latency-sensitive applications like IoT, augmented reality (AR), and video streaming. The following table summarizes the performance metrics before and after MEC integration:

TABLE I
PERFORMANCE METRICS BEFORE AND AFTER MEC INTEGRATION

Metric	Before MEC	After MEC	Improvement
Latency (ms)	30–40 ms	5–10 ms	85%
Throughput (Mbps)	300 Mbps	500 Mbps	66%
Packet Loss (%)	2%	0.2%	90%

The integration of MEC resulted in significant performance gains across the board:

- **Latency Reduction**: The most striking improvement was in latency, which dropped from an average of 30–40 ms to 5–10 ms, reflecting an 85% improvement. This reduction is critical for applications requiring real-time data processing and low-latency communication, such as augmented reality (AR), virtual reality (VR), and interactive gaming [1].
- **Throughput Improvement**: Throughput increased from 300 Mbps to 500 Mbps, a 66% improvement. This increase supports the growing demand for high-bandwidth applications such as 4K video streaming, IoT data transmission, and smart city services, ensuring seamless communication even under heavy network loads [2].
- **Packet Loss Reduction**: The reduction in packet loss from 2% to 0.2% demonstrates a 90% improvement in network reliability. Lower packet loss is essential for maintaining the quality of service (QoS) in latency-sensitive applications, ensuring continuous service delivery even in congested or variable network conditions [3], [4].

These results were validated using tools such as Wireshark, which enabled detailed analysis of network traffic to assess the impact of MEC integration on latency and throughput. Wireshark's ability to capture and decode 5G-specific protocols such as NAS, NGAP, and GTP-U proved instrumental in troubleshooting and validating the network's behavior [4]. The use of UERANSIM for simulating User Equipment (UE) interactions also played a critical role in evaluating the impact of MEC on session continuity and registration processes, leading to improved packet loss rates and overall network stability [3].

IV. Conclusion and Future Scope

The integration of MEC with the 5G core network demonstrated significant improvements in network performance, making it a viable solution for low-latency and high-reliability applications. Key results include:

- Latency reductions of up to 85%, crucial for real-time and interactive applications.
- Throughput improvements of up to 66%, enabling high-bandwidth applications like video streaming and IoT services.
- Enhanced real-time performance for latency-sensitive services, improving user experiences in areas like AR, video streaming, and industrial automation.

Future work can explore:

- Optimizing MEC for mission-critical applications such as autonomous driving, where ultra-low latency and high availability are paramount.
- Leveraging AI and machine learning techniques for dynamic resource allocation within MEC environments to handle network traffic more efficiently and intelligently.
- Investigating standalone 5G integration with MEC for enhanced scalability, where MEC can be deployed independently of existing core networks to support specific, localized needs.
- Utilizing blockchain for securing MEC deployments, particularly for applications involving sensitive data, such as healthcare, finance, and government services.
- Enhancing MEC Location API simulators to support varied UE behaviors like vehicles, pedestrians, and drones, which will be useful for future mobility and IoT applications [2].
- Expanding the understanding of MEC's role in enabling 5G use cases like smart cities, connected vehicles, and industrial automation.

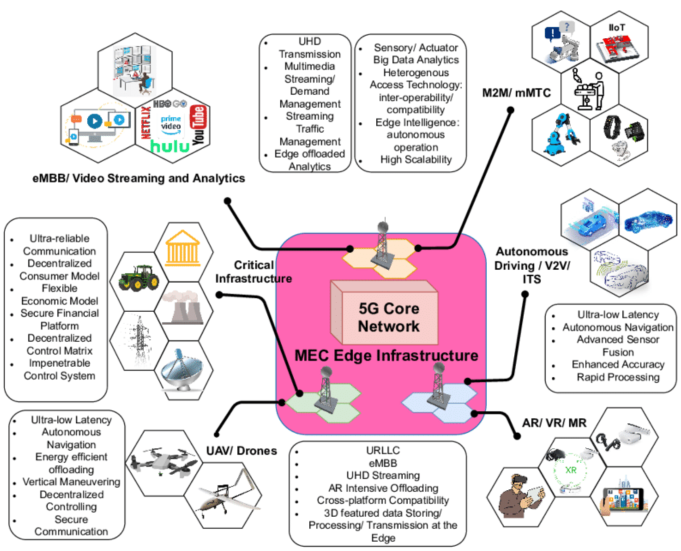


Fig. 5. MEC-Enabled 5G Use Cases.

The evolution of MEC and 5G integration is pivotal to shaping the future of connected systems. As the deployment of 5G networks becomes more widespread, MEC will play a central role in driving the next generation of high-performance, low-latency applications.

Acknowledgment

The team acknowledges the support and guidance provided by mentor Dr.Bhupendra Kumar and the resources that facilitated the successful execution of this project. Special thanks to the OpenAirInterface (OAI) and ETSI communities for their detailed documentation and open-source tools

References

[1] L. Contreras and C. Bernardos, "Overview of architectural alternatives for the integration of etsi mec environments from different administrative domains," *Electronics*, vol. 9, p. 1392, 08 2020.

[2] L. Foundation, "Mec location api hands-on," http://linksfoundation.com, accessed: 2023.

[3] G. Documentation, "Deploy sa5g with ueransim," https://gitlab.eurecom.fr, accessed: 2023.

[4] W. U. Documentation, "Packet analysis," https://www.wireshark.org, accessed: 2023.