Enabling NS-3 Simulations Integrated with Latest Versions of Open RAN Near-RT RICs

Andrey A. Miranda de Oliveira, João Albuquerque, Cleverson Nahum, Daniel Campos, Kleber Cardoso, Aldebaro Klautau, José Rezende

Abstract— **The evolution of 5G networks and the Open RAN architecture demands flexible and interoperable simulation tools for validating intelligent RAN control solutions. However, existing simulators often lack seamless integration with real-time RAN controllers, limiting their use in realistic closed-loop experiments. This paper presents the development of NORI (New Open RAN Interface), a modular and scalable solution to integrate the NS-3 simulator with the Near-RT RIC of the Open RAN architecture. NORI supports recent NS-3 versions and enables metric collection via the standardized E2 interface, without requiring modifications to the simulator core. It is compatible with the 5G-LENA module and the full Near-RT RIC from the O-RAN Software Community. Experimental results demonstrate the viability of NORI for realistic 5G simulation scenarios, enabling performance evaluation and metric reporting in disaggregated RAN environments.**

Keywords— **NS-3, Open RAN, Near-Real-Time RIC, E2 Interface, 5G-LENA, xApps, Network Simulation.**

I. Introduction

The advent of 5G networks has introduced a range of heterogeneous use cases—such as Enhanced Mobile Broadband (eMBB), Ultra-Reliable Low-Latency Communications (URLLC), and massive Machine-Type Communications (mMTC)—that demand flexible and efficient radio access network (RAN) architectures. Traditional RAN models, with tightly coupled hardware and software, fall short in meeting these evolving and dynamic requirements [1]. Even centralized approaches like Cloud-RAN (C-RAN) have demonstrated limitations, particularly in terms of interoperability and scalability.

In response to these challenges, the Open RAN (O-RAN) architecture promotes disaggregation, virtualization, and open interfaces, which enable interoperability between vendors and facilitate AI-driven network optimization through components such as the Near-Real-Time RAN Intelligent Controller (Near-RT RIC). These innovations allow for greater programmability and fine-grained control of the RAN. However, validating such complex and dynamic systems remains a significant challenge. Testing in real-world deployments often demands extensive infrastructure, incurs high costs, and introduces risks to service stability—especially when evaluating AI-based control algorithms that are still under training or tuning phases[2].

To address the need for cost-effective experimentation and validation, simulation tools have gained prominence as an alternative to physical testbeds. These tools allow researchers and developers to evaluate network behaviors and algorithmic performance under controlled, repeatable, and scalable conditions. Several frameworks have emerged with this goal, including ns-O-RAN, which integrates the NS-3 simulator with a simplified version of the Near-RT RIC[3]. While ns-O-RAN represents a significant step toward enabling closed-loop control experiments, it suffers from critical limitations: it is bound to outdated NS-3 versions, lacks modularity, and requires invasive modifications to the simulator's core, which hampers portability and future upgrades.

Building on the need for a more robust and scalable solution, this work proposes NORI (New Open RAN Interface), a module that connects NS-3 [1] with a compatible Near-RT RIC. NORI supports the NR 5G Lena [2] module, avoids core modifications to NS-3, and enables real execution of xApps through the standardized E2 interface. This design not only supports advanced channel modeling but also facilitates the development and validation of AI-driven optimization algorithms in realistic and flexible Open RAN scenarios.

II. O-RAN Overview and Near-RT RIC Architecture

The Open Radio Access Network (O-RAN) architecture represents a shift from traditional monolithic RAN implementations toward a disaggregated and interoperable network model, designed to support the evolving requirements of 5G and beyond. In the O-RAN architecture, the base station is decomposed into three logical units: the Radio Unit (O-RU), Distributed Unit (O-DU), and Central Unit (O-CU), each connected via standardized interfaces. However, the most significant innovation lies in the introduction of intelligent controllers—specifically, the Near-RT RIC—which enables dynamic and programmable control of RAN behavior at near-real-time scales, typically between 10 milliseconds and 1 second. The Near-RT RIC operates at the edge of the RAN and executes third-party applications, known as xApps, that can monitor and influence radio resource management functions such as handover, scheduling, and load balancing [2].

Andrey Oliveira, João Albuquerque, Cleverson Nahum, and Aldebaro Klautau are with LASSE - Telecommunications, Automation and Electronics Research and Development Center, Federal University of Pará (UFPA), Belém-PA, Brazil. E-mails: {andrey.oliveira, joao.barbosa.albuquerque}@itec.ufpa.br; {cleversonahum, aldebaro}@ufpa.br; Dante Campos and Kleber Cardoso are with Universidade Federal de Goiás (UFG), Goiânia, GO – Brazil. E-mails: {danielcampos, kleber}@inf.ufg.br; José F. de Rezende is with Systems Engineering and Computer Science Program, Federal University of Rio de Janeiro, Rio de Janeiro-RJ 21941-914, Brazil. E-mail: rezende@land.ufrj.br).

[1] https://www.nsnam.org/
[2] https://5g-lena.cttc.es/

These xApps interact with RAN nodes through the standardized E2 interface, which is responsible for both collecting telemetry and issuing control commands. The E2 interface is built on the E2 Application Protocol (E2AP), which provides a transport-agnostic messaging framework for communication between the RIC and the E2 nodes (e.g., O-DUs, O-CUs). To structure this interaction, E2 leverages Service Models (E2SMs) that define the semantics of exchanged messages for specific functionalities. Among the most relevant E2SMs are E2SM-KPM (Key Performance Measurement), used for the collection of fine-grained metrics such as throughput, SINR, and retransmissions, and E2SM-RC (RAN Control), which enables the RIC to send control commands back to the RAN based on policy decisions or AI-driven logic. These service models are essential for ensuring compatibility and interoperability across vendors. In this work, we focus on integrating a full Near-RT RIC implementation with the NS-3 simulator through the module called NORI. Our approach leverages both E2SM-KPM and E2SM-RC to enable a realistic and standards-compliant control loop between simulated RAN nodes and real xApps running on the Near-RT RIC[2].

III. Design and Implementation of the NORI Module in NS-3

This section presents the design and implementation of the NORI module, a solution developed to integrate NS-3 with the Near-RT RIC of the O-RAN architecture in a modular and scalable way. Building upon the limitations identified in the ns-O-RAN framework [3], NORI avoids changes to the NS-3 core, supports newer versions of the simulator, and replaces the mmWave module with the more versatile NR 5G Lena. It enables seamless communication with a full-featured Near-RT RIC and real xApps through the E2 interface, promoting a flexible and reusable simulation environment aligned with O-RAN standards.

A. NORI Module Architecture

The NORI module was implemented as an independent component within the NS-3 simulator, following a modular design that enables seamless and standardized communication with the Near-RT RIC of the O-RAN architecture. As shown in Fig.1, NORI bridges the simulated 5G NR environment—built using the NR module and the 5G-LENA [4] extension—with a real, production-grade Near-RT RIC, allowing the exchange of telemetry and control messages in accordance with O-RAN specifications. The development and testing of NORI were carried out using the Blueprint v1 environment provided by the OpenRAN@Brasil initiative[5], which includes a pre-configured RIC infrastructure based on the O-RAN Software Community (OSC) stack.

At the core of the NORI architecture is the integration of the E2 Termination interface directly into the simulated gNB nodes. This integration enables each simulated base station to initiate and maintain an E2 session with the RIC, following the E2AP procedures. The E2 Termination is embedded within each gNB and connects to the RIC via the e2sim_lib library, a component derived from the open-source e2sim simulator maintained by the O-RAN Software Community, as utilized in the work of Andrea Lacava et al. [6]. Originally designed to emulate E2 nodes in integration tests with real RICs, e2sim was adapted into a lightweight library (e2sim_lib) to support runtime E2 message encoding, decoding, and session handling within NS-3 simulations. This adaptation was crucial to avoid intrusive changes to NS-3 or its NR module while still enabling compliance with E2AP and support for relevant service models such as E2SM-KPM and E2SM-RC.

While the NS-3 NR module is responsible for simulating network behavior—including user equipment (UE) mobility, radio channels, and gNB transmission—NORI augments each simulated gNB with the capability to generate and transmit real-time performance metrics to the Near-RT RIC and, conversely, to receive control commands from xApps. The E2 Termination component embedded in each gNB handles E2 setup procedures, maintains Stream Control Transmission Protocol (SCTP) based sessions with the RIC, and supports subscription and reporting of standardized metrics. These include downlink throughput, retransmissions, SINR, modulation usage, and other radio-level indicators defined by the E2SM-KPM model. This architecture was designed with maintainability, ensuring all additions remain modular and compatible with future updates of NS-3 and the LENA module.

NORI enables a complete simulation to monitoring feedback loop between the NS-3 environment and a real Near-RT RIC instance. Through the integration with E2 Termination components and support for the E2SM-KPM service model, it allows simulated gNBs to report detailed performance metrics—such as throughput, SINR, and modulation usage—to xApps operating on the RIC. These xApps can process the telemetry in real time to implement monitoring, anomaly detection, or learning-based analytics based on realistic and dynamic simulation scenarios. By supporting large-scale topologies and realistic traffic models, NORI provides a powerful platform for the development and validation of AI-enabled RAN monitoring strategies in a reproducible and cost-effective manner. Furthermore, the non-intrusive architecture of NORI ensures compatibility with future NS-3 updates and promotes reusability across research projects.

B. Implementation of Module Components

To ensure that the NORI module remained modular and followed NS-3's development standards, it was necessary to reorganize the code originally developed by Lacava et al. [3] and implement new classes to handle functions that were previously embedded directly in the simulator core. This reorganization was designed to avoid intrusive modifications in both NS-3 and the NR 5G Lena module, ensuring greater compatibility with future versions. Fig. 2 presents the main classes created and the minimal updates required to integrate NORI into the simulation environment. This approach represents an advancement over the ns-O-RAN framework, which depended on specific versions of NS-3 and deep modifications to the mmWave module. By adopting a decoupled and modular architecture, NORI overcomes these limitations, improving maintainability, extensibility, and compatibility with the latest O-RAN Alliance specifications and ongoing NS-3 updates.

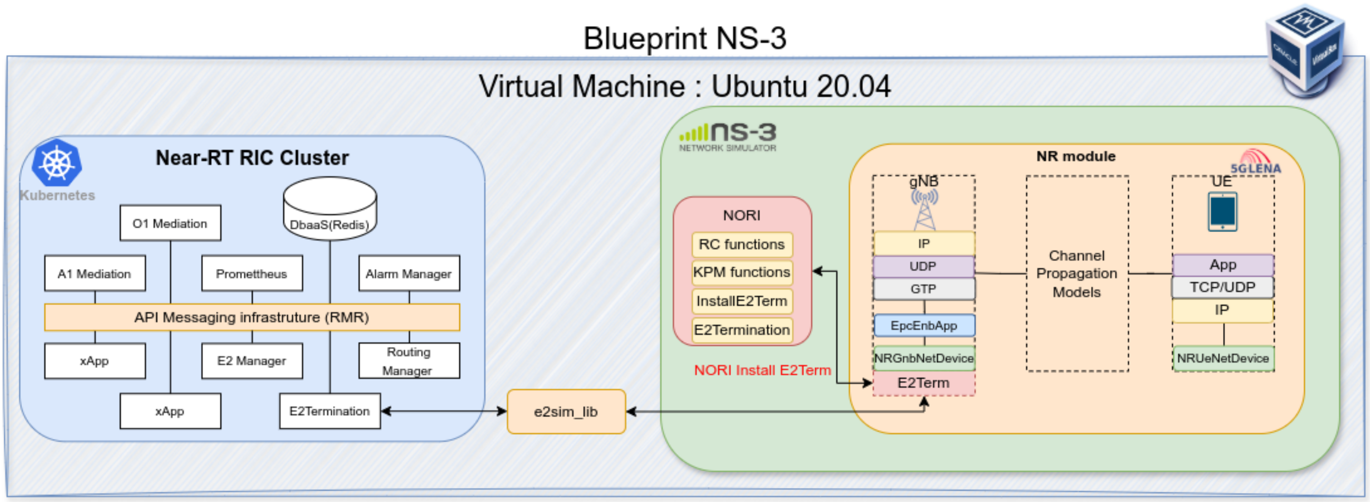

Fig. 1: Overview of the NORI architecture, showing its integration with the NR module and the Near-RT RIC [5].

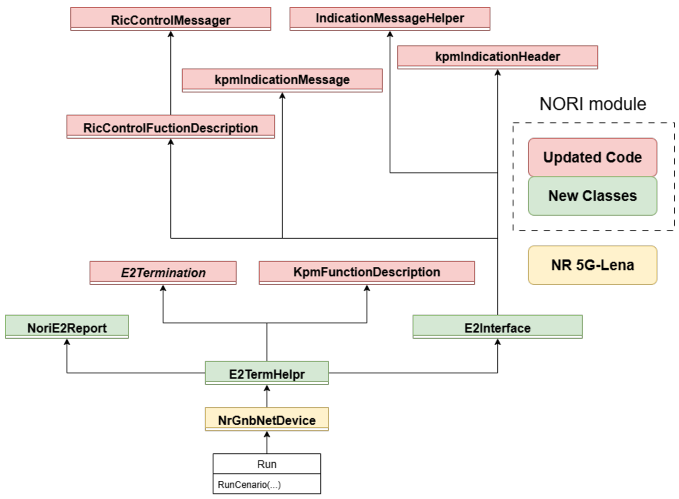

Fig. 2: Class overview of the NORI module and integration points within NS-3 [5].

1) NoriE2Report Class: The `NoriE2Report` class was created to act as a MAC layer statistics collector within the NORI module, enabling the extraction and analysis of performance metrics from the simulated network. This allows for real-time monitoring of packet transmissions, retransmissions, and data volume, which is essential for evaluating network behavior and supporting decision-making processes in the Near-RT RIC context. Moreover, `NoriE2Report` implements an SNR bin-specific logic, where each SNR value is stored in a specific bin to perform a simple estimation of all UEs' SNR probability density functions (PDFs).

This design makes it possible to access transmission-related information, segmenting data according to various traffic characteristics. The methods in the E2Interface class allow you to query the total number of PDUs sent to a specific user in a given cell, the number of initial transmissions and retransmissions, and the total transmitted volume. The class also enables tracking of modulation schemes used, such as QPSK, 16QAM, and 64QAM, as well as the total number of symbols transmitted. Prior to this implementation, most performance monitoring was tightly coupled to the NS-3 core code, making adaptations and maintenance difficult. By isolating the traffic monitoring functionality into a standalone class, this design improves code organization and ensures greater compatibility with future updates of NS-3.

2) NoriE2Interface Class: The `NoriE2Interface` class was developed to manage communication between the NORI module and the Open RAN architecture, enabling the exchange of information between the NS-3 simulator and the Near-RT RIC. Through the E2 interface, this class facilitates the transmission of network statistics, monitoring of signal quality, and the delivery of control commands, ensuring that the simulated network environment can interact with external control logic in a realistic manner.

As with `NoriE2Report`, one of the primary development challenges was implementing the necessary methods and functionalities without modifying the NS-3 or NR 5G Lena core code. The `NoriE2Interface` class was therefore structured in compliance with NS-3's object system. This class provides interfaces for handling RIC Subscription Requests, generating and sending RIC Indication messages, and registering various metrics such as transmitted PDUs and SINR readings. It supports end-to-end monitoring of user traffic and radio conditions, which are essential for enabling xApps to make decisions based on realistic and dynamic simulation data.

In addition to capturing performance statistics, the class integrates with NS-3's statistics calculators across the entire NR stack, from the Radio Resource Control (RRC) layer to the Physical (PHY) layer. It performs preprocessing of the collected data, which is then sent via the KPM service model for each of the disaggregated RAN entities, such as the Distributed Unit (DU) and Centralized Unit (CU), including both Control Plane (CP) and User Plane (UP) traffic. This integration provides a comprehensive set of metrics that can be consumed by xApps running on the Near-RT RIC, enabling more granular and realistic network monitoring. By encapsulating all logic related to E2 data reporting and interaction in a single, modular component, the NoriE2Interface enhances the flexibility and reusability of the NORI module, allowing it to be easily adapted to different simulation scenarios and control objectives.

- Modulation Order
- RLC Buffer Size
- Throughput
- Signal-to-Noise Ratio (SNR)
- Latency
- SINR (Signal-to-Interference-plus-Noise Ratio)
- Packet Delay

3) E2TermHelper Class: The `E2TermHelper` class serves as the topology helper of the NORI module and is responsible for simplifying the creation and configuration of simulation scenarios in NS-3. This class facilitates the installation of the E2 Termination component in the simulated RAN, allowing network elements to communicate with the Near-RT RIC without requiring manual and detailed configuration within the simulation code.

Before the implementation of `E2TermHelper`, the setup of the E2 interface had to be performed manually for each node in the simulation, which increased complexity and the likelihood of configuration errors. With this helper class, the process of installing and enabling the E2 interface across multiple nodes becomes automated, promoting greater modularity and reducing the effort required to create and replicate simulation environments.

The class provides a set of methods to install the E2 Termination on individual or multiple NetDevices, enable signal to Interference plus Noise Ratio (SINR) tracing, activate logging, and link the data collection mechanisms from various protocol layers such as RLC and PDCP. These capabilities ensure that the simulation produces rich and structured telemetry data for use by the Near-RT RIC.

In addition, `E2TermHelper` can subscribe to trace sources within the E2 interface functions, enabling detailed event logging that is particularly useful during debugging and performance validation. This centralized and standardized configuration logic allows researchers to create flexible and reproducible experiments that integrate seamlessly with real RICs and xApps.

By encapsulating the configuration logic in a reusable component, `E2TermHelper` not only improves developer productivity but also ensures consistency and scalability across different simulation setups involving the NORI module.

The e2sim library, originally developed by the O-RAN Software Community to emulate E2 nodes communicating with the Near-RT RIC, was adapted in the NORI module to integrate NS-3 with a real Near-RT RIC based on Release I. To ensure compatibility, the E2AP was updated to version v2.02.03, including modifications to ASN.1 structures and serialization methods to support new fields like Transaction ID and E2ConfigIE. These changes enable a standards-compliant E2 session between simulated nodes and the RIC. Figure 3 shows the updated e2sim_lib architecture, highlighting encoding/decoding layers, socket management, and message handlers. The E2 Setup Request message, crucial in this process, contains the Global E2 Node ID and RAN Function List, with added fields for transaction tracking and configuration parameters. These updates ensure full protocol compliance, allowing NS-3 to participate in realistic control loops with the Near-RT RIC for developing and testing real xApps and control policies.

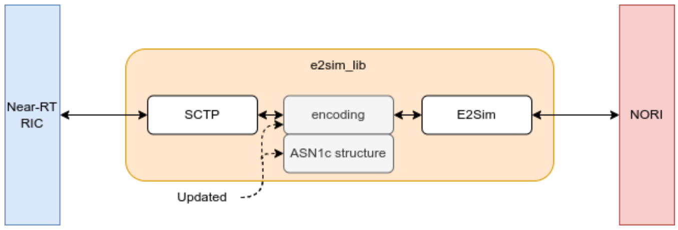

Fig. 3: Simplified architecture of the e2sim_lib showing updated components integrated into NORI [5].

IV. Results

This scenario simulates a simple 5G NR network focused on Quality of Service (QoS) for two traffic types: low-latency (eMBB) and voice (GBR). The topology consists of 1 gNB (base station) and 4 UEs (user equipment) arranged in a 3x3 grid, with 5 meters horizontal and vertical spacing between nodes. The scenario employs the UMi (Urban Micro) channel model, without considering shadowing effects. The system operates at a 4 GHz center frequency with a 5 MHz bandwidth and a total transmit power of 43 dBm. Numerology is 0, and there is a single component carrier (CC) with one bandwidth part (BWP). Beamforming is ideal and isotropic, with antenna arrays consisting of 1x1 elements at both gNB and UEs.

An xApp implemented in the Near-RT RIC control environment uses reinforcement learning (RL) algorithms to dynamically optimize resource allocation among the slices. This xApp collects real-time performance metrics from network nodes (gNBs) through the standardized E2 interface, using the E2SM-KPM service model. At this stage, the RL agent communicates directly with NS-3 without utilizing the E2SM-RC interface, which is planned to be implemented in future work.

Resource scheduling in the simulation supports RAN slicing, with two types of traffic differentiated by their Quality of Service Class Identifiers (QCI): low-latency, high-throughput traffic (eMBB) and voice traffic with guaranteed bit rate (GBR). The first two UEs are assigned to slice 1, which carries the low-latency traffic, while the last two UEs belong to slice 2, dedicated to voice traffic. Each slice is configured with its own priorities and QoS requirements, with slice 1 targeting a throughput of 4 Mbps and slice 2 targeting 1 Mbps, enabling a realistic performance evaluation under diverse traffic demands.

The NORI module enables the collection of detailed performance metrics, supporting up to 35 cell-level metrics and 25 UE-level metrics. Examples of these metrics include Downlink Throughput, Signal-to-Interference-plus-Noise Ratio (SINR), Transport Block Errors, Buffer Size, and Bearer Establishment Success. Among these, the average throughput per slice is a key input used by the RL agent implemented in the MobNet environment. Using this comprehensive set of observations,

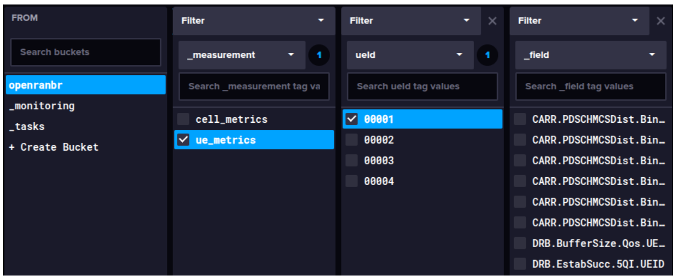

Fig. 4: Example of metrics available in InfluxDb

the agent calculates and applies resource allocation decisions by distributing Physical Resource Blocks (PRBs) across the slices, aiming to optimize resource utilization while meeting the specific QoS requirements of each slice. The Fig.4 exemplifies the set of metrics available in the InfluxDB database, illustrating the variety and granularity of data collected for analysis and decision-making.

Fig. 5: Average throughput over time for two network slices, with slice 1 maintaining higher rates than slice 2.

Fig.5 illustrates the average throughput observed for each slice over time. Initially, there is some variability in the results due to the reinforcement learning (RL) agent's exploration process during training. As the agent learns and adapts, the throughput stabilizes, reflecting a differentiated and stable allocation pattern that enforces the QoS requirements for both eMBB and GBR slices. Similarly, Fig.6 shows the average throughput per UE, grouped by slice membership. It is important to note that some UE metrics overlap in the figure due to the number of UEs per slice (two UEs each), which causes their throughput curves to be closely aligned. Overall, the results confirm that the xApp effectively maintains throughput fairness and service differentiation across users despite traffic variability and network dynamics.

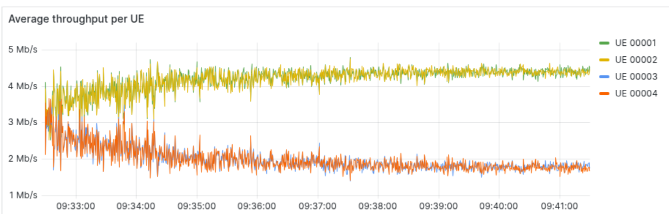

Fig. 6: Average throughput per UE over time, showing stable transmission rates for four users ranging between 2 Mb/s and 4.5 Mb/s.

V. CONCLUSION

The proposed NORI module demonstrated correct functionality and effectiveness in capturing real-time performance metrics from simulated gNBs, validating its integration with the Near-RT RIC via the E2 interface. Through its modular and non-intrusive design, the solution proved capable of accurately monitoring key performance indicators such as throughput, latency, and retransmissions, enabling realistic closed-loop control experiments. Furthermore, the architecture has shown to be scalable and adaptable for future use cases, allowing researchers to extend the simulation framework to evaluate new xApps, diverse traffic profiles, and advanced RAN control strategies in compliance with O-RAN standards. The NORI module is publicly available in an open repository, facilitating broader access and collaboration within the research community [7]. In future work, the E2SM-RC will be implemented to standardize communication between the xApp and e2sim, enabling more robust and compliant control interactions.

AKNOWLEDGEMENTS

This work was partially financed by Brasil 6G project (RNP/MCTI grant 01245.010604/2020-14); OpenRAN Brazil - Phase 2 project (MCTI grant Nº A01245.014203/2021-14); Universal (CNPq grant 405111/2021-5); Project Smart 5G Core And MUltiRAn Integration (SAMURAI) (MCTIC/CGI.br/FAPESP under Grant 2020/05127-2).

REFERENCES

[1] V. S. Pana, O. P. Babalola, and V. Balyan, "5G radio access networks: A survey," *Computer Communications and Networks*, vol. 14, p. 100170, 2022, [Online]. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2590005622000315

[2] M. Polese *et al.*, "Understanding o-ran: Architecture, interfaces, algorithms, security, and research challenges," *IEEE Communications Surveys & Tutorials*, vol. 25, no. 2, pp. 1376–1411, 2023. [Online]. Available: https://ieeexplore.ieee.org/document/10024837

[3] A. Lacava, M. Polese, R. Sivaraj, R. Soundrarajan, B. S. Bhati, T. Singh, T. Zugno, F. Cuomo, and T. Melodia, "Programmable and customized intelligence for traffic steering in 5G networks using open ran architectures," *IEEE Transactions on Mobile Computing*, vol. 23, no. 4, pp. 2882–2897, 2024.

[4] N. Patriciello, S. Lagen, B. Bojovic, and L. Giupponi, "An E2E simulator for 5G NR networks," vol. 96, p. 101933. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1569190X19300589

[5] LABORA-INF-UFG, "OpenRAN-BR-Blueprint," https://github.com/LABORA-INF-UFG/openran-br-blueprint?tab=readme-ov-file, [Online].

[6] A. Lacava *et al.*, "ns-o-ran: Simulating o-ran 5G systems in ns-3," in *Proc. of the 2023 Workshop on ns-3*, 2023, pp. 35–44, [Online]. [Online]. Available: http://arxiv.org/abs/2305.06906

[7] "lasseufpa/nori," original-date: 2024-12-04T13:13:56Z. [Online]. Available: https://github.com/lasseufpa/nori