An Artificial Intelligent Aided Unified Network for Secure Beyond 5G Long Term Evolution [GA: 101096456]

Deliverable 2.4

NANCY System Performance Assessment

Programme: HORIZON-JU-SNS-2022-STREAM-A-01-06

Start Date: 01 January 2023

Duration: 36 Months

Co-funded by the European Union

6GSNS

NANCY project has received funding from the Smart Networks and Services Joint Undertaking (SNS JU) under the European Union's Horizon Europe research and innovation programme under Grant Agreement No 101096456.

Document Control Page

Deliverable Name	NANCY System Performance Assessment
Deliverable Number	D2.4
Work Package	WP2 'Usage Scenario and B-RAN Modelling, Network Requirements, and Research Framework'
Associated Task	T2.4 'Performance Assessment via Modelling and Testbed Results'
Dissemination Level	Public
Due Date	31 December 2025 (M36)
Completion Date	29 December 2025
Submission Date	30 December 2025
Deliverable Lead Partner	INNO
Deliverable Author(s)	Stylianos Trevlakis (INNO), Vasileios Kouvakis (INNO), Eirini Gkarnetidou (INNO), Lamprini Mitsiou (INNO), Theodoros Tsiftsis (INNO), Anna Triantafyllou (UOWM), Dimitrios Pliatsios (UOWM)
Version	1.0

Document History

Version	Date	Change History	Author(s)	Organisation
0.1	10/10/2025	Initial version	Stylianos Trevlakis, Kouvakis Vasileios, Eirini Gkarnetidou, Theodoros Tsiftsis, Lambrini Mitsiou	INNO
0.2	20/10/2025	Finalized ToC & contributions in Section 2	Stylianos Trevlakis, Kouvakis Vasileios, Eirini Gkarnetidou, Theodoros Tsiftsis, Lambrini Mitsiou	INNO
0.3	5/11/2025	Contributions in Sections 2, 3, & 4	Stylianos Trevlakis, Kouvakis Vasileios, Eirini Gkarnetidou, Theodoros Tsiftsis, Lambrini Mitsiou	INNO
0.4	12/11/2025	Completion of Section 2	Stylianos Trevlakis, Kouvakis Vasileios, Eirini Gkarnetidou, Theodoros Tsiftsis, Lambrini Mitsiou	INNO
0.5	30/11/2025	Completion of Sections 3, 4, & 5	Stylianos Trevlakis, Kouvakis Vasileios, Eirini Gkarnetidou, Theodoros Tsiftsis, Lambrini Mitsiou	INNO
0.6	18/12/2025	Complete version provided for internal review	Stylianos Trevlakis, Kouvakis Vasileios, Eirini Gkarnetidou, Theodoros Tsiftsis, Lambrini Mitsiou	INNO
0.7	24/12/2025	Internal review	Blaz Bertalanic (IJS), Emanuele De Santis (CRAT)	IJS, CRAT
0.8	24/12/2025	Review comments addressed	Stylianos Trevlakis, Eirini Gkarnetidou, Lambrini Mitsiou	INNO

1.0	29 December 2025	Final version after quality revisions	Anna Triantafyllou, Dimitrios Pliatsios	UOWM

Internal Review History

Name	Organisation	Date
Blaz Bertalanic	IJS	24 December 2025
Emanuele De Santis	CRAT	24 December 2025

Quality Manager Revision

Name	Organisation	Date
Anna Triantafyllou, Dimitrios Pliatsios	UOWM	29 December 2025

Table of Contents

Table of Contents 4
List of Figures 6
List of Tables 7
List of Acronyms 8
Executive summary 9
1. Introduction 10
1.1. Purpose of this Deliverable 10
1.2. Relation to Other Deliverables 10
1.3. Structure of the Deliverable 10
2. Simulation Framework 12
2.1. Ray-tracing Simulations 12
2.1.1. Simulation platform and core capabilities 12
2.1.2. Geographical and environmental modeling 12
2.1.3. Network topology and base station deployment 13
2.1.4. User equipment initialization and mobility 15
2.1.5. Simulation dynamics and temporal evolution 15
2.1.6. Reconfigurable intelligent surfaces integration 16
2.1.7. Ray tracing and channel computation 16
2.1.8. Physical phenomena and realism validation 17
2.2. Collected Metrics and Preprocessing 17
3. Intelligent Network Management 20
3.1. Outage Probability Prediction 20
3.1.1. Overview and motivation 20
3.1.2. Input feature representation 21
3.1.3. Feature interaction and model processing 21
3.1.4. Output interpretation and performance considerations 22
3.2. Coverage Probability Prediction 23
3.3. SNN-based Optimal BS-UE Association 26
3.3.1. Background 26
3.3.2. Methodology 27
3.3.3. Results 32
4. Blockchain 36
4.1. B-RAN simulation framework 36
4.1.1. Configurable system parameters 36

4.1.2. Integration with Sionna simulation environment 37
4.2. Hyperledger fabric 40
5. End-to-End performance assessment 45
5.1. End-to-end System Model 45
5.2. Performance Assessment 45
5.2.1. Temporal dynamics 45
5.2.1. AI impact 47
5.2.2. Blockchain impact 49
5.2.3. Handover thresholds impact 50
5.2.4. Experimental results 52
6. Conclusions 54
Bibliography 56

List of Figures

Figure 1. Real-world map with defined user movement areas 14
Figure 2. Timeseries of RSS for an individual user 15
Figure 3. Simulation-specific system model 18
Figure 4. AI-specific system model 20
Figure 5. Handover prediction sensitivity as a function of time step size (Δt) and outage probability threshold 23
Figure 6. Handover prediction sensitivity as a function of time step size (Δt) and RSRQ threshold 25
Figure 7. 3D representation of the system model 27
Figure 8. UE-BS association 28
Figure 9. Coverage map 28
Figure 10. SNN architecture 30
Figure 11. Top-down and bottom-up approaches 31
Figure 12. Comparison of the achievable accuracy of the two approaches 33
Figure 13. Confusion matrix of the top-down approach 33
Figure 14. Confusion matrix of the bottom-up approach 34
Figure 15. Comparison of the achievable data rate of the two approaches 35
Figure 16. Data rate error of the "top-down" approach 35
Figure 17. Data rate error of the ”bottom-up” approach 35
Figure 18. Blockchain-specific system model 36
Figure 19. Latency performance under varying traffic and mining power conditions 38
Figure 20. Impact of traffic load on call drop rate across different mining power configurations 39
Figure 21. Latency performance comparison under varying traffic conditions 41
Figure 22. Call drop performance comparison under varying traffic conditions 42
Figure 23. Latency Performance Across Blockchain Implementations Under Varying Traffic Intensity 43
Figure 24. End-to-end system model 45
Figure 25. Timeseries of RSS for an individual user 46
Figure 26. Timeseries of RSS for an individual user 46
Figure 27. Call drop performance and saved calls as a function of AI processing delay 47
Figure 28. Effect of Blockchain delay on dropped call rates with and without AI integration 48
Figure 29. Impact of handover start threshold on system performance with and without Blockchain 49
Figure 30. Impact of Blockchain delay on call drops 50
Figure 31. Impact of call drop threshold on system performance with and without AI integration 51
Figure 32. Impact of call drop threshold on system performance with and without Blockchain authorization 52
Figure 33. Experimental results by OTE 53

List of Tables

Table 1. Region Index Mapping 14
Table 2. Collected Metrics 17

List of Acronyms

Acronym	Explanation
UE	User Equipment
BS	Base Station
RSRQ	Reference Signal Received Quality
AI	Artificial Intelligence
KPI	Key Performance Indicators
NTN	Non-Terrestrial Networks
SNN	Spiking Neural Network
UAV	Unmanned Aerial Vehicles
LIF	Leaky Integrate-and-Fire
XAI	Explainable Artificial Intelligence
GPU	Graphics Processing Unit
CPU	Central Processing Unit
RAM	Random access memory
RIS	Reconfigurable Intelligent Surfaces
RSS	Received Signal Strength
CQI	Channel Quality Indicator
SINR	Signal-to-Interference-plus-Noise Ratio
DL	Downlink Buffer
PRBs	Physical Resource Blocks
Tx	Transmitter
Rx	Receiver
YOLO	You Only Look Once
R-CNN	Region-based Convolutional Neural Network
SSD	Single Shot multibox Detector
RSU	Road Side Unit
B-RAN	Blockchain - Radio Access Network
Adam	Adaptive Learning Rate Optimization
AWS	Amazon Web Services
DLT	Distributed Ledger Technology

Executive summary

This deliverable provides an integrated simulation environment designed to analyze the theoretical B-RAN and AI models under realistic operational environments. This deliverable fills the major gap between the conceptualization of theoretical designs and the verification of those designs in practice, using the combination of GPU-accelerated ray tracing to simulate the physical world and distributed ledger technology, along with the capability to perform machine learning (ML) based inference.

The primary goal of this deliverable is to validate the performance of two complex, interconnected technologies, specifically AI-driven network optimization and blockchain-based access, within simulated environments that accurately represent real-world complexity and compare with experimental results. Unlike prior studies, the NANCY simulation framework is based on actual geographic data, accurate building geometries, realistic materials and physical properties, and human-like mobility patterns. This grounding in the physical world ensures that the performance metrics and insights generated by the simulations can be translated into meaningful results applicable to actual operating wireless networks.

The NANCY simulation framework is composed of three integrated components, with each one representing a unique and complementary aspect of the overall B-RAN architecture. The first component utilizes the NVIDIA Sionna ray tracing engine, a GPU-accelerated platform for generating high-fidelity wireless propagation models that include multi-path interference, reflection, diffraction, and RIS. The second component enables the operation of three different AI prediction models that utilize extracted features from the simulation data to predict network failure and to optimize UE to BS associations. In more detail, the three models, namely outage probability prediction, coverage probability prediction, and SNN-based optimal UE-BS association, demonstrate measurable performance improvements. The third component is the blockchain, which is modeled with actual deployments as well as the theoretical B-RAN framework, both of which evaluate the integration of blockchain in wireless networks in order to ensure security and accountability.

The evaluation through real-world simulations and experimental data presented in this deliverable provides several key insights related to the performance of the integrated system. The AI models demonstrate significant accuracy in predicting and optimizing network performance. The blockchain integration, both theoretical and experimental, demonstrates that it is possible to implement distributed mechanisms for wireless network management and that the latency of these mechanisms will vary predictably with respect to mining power and the rate at which transactions are received. Critically, the end-to-end evaluations demonstrate complex tradeoffs, such as the fact that increasing the AI model complexity decreases the number of handovers that can be completed in time. Thus, computational capabilities for executing AI and blockchain procedures emerge as a critical design parameter for the system.

Finally, this deliverable establishes the B-RAN simulation framework as a verified platform for evaluating future generations of network management architectures and provides evidence for the feasibility of integrating AI-driven and blockchain-based optimization in wireless networks, while providing insight into the operational limitations and design tradeoffs.

1. Introduction

1.1. Purpose of this Deliverable

Validation of complex cellular systems, such as those employing blockchain radio access networks (B-RAN), has traditionally been challenging due to the many variables that need to be accounted for, including wireless propagation, user mobility, interference, and resource constraints. The NANCY Project is addressing this problem through the development of a simulation tool to simulate complex cellular systems and their associated environments using artificial intelligence (AI) and blockchain technologies. For example, AI-based predictive intelligence is being developed to enhance the operation of the network, while blockchain-based distributed coordination is being developed to secure and coordinate user access to the network.

An essential objective of the NANCY Project is to demonstrate that it is feasible to deploy the B-RAN architecture to a variety of deployment scenarios. In order to achieve this goal, D2.4 presents a complete simulation tool that is based on ray tracing and integrates AI models as well as a theoretical blockchain framework to evaluate the performance of the B-RAN architecture under conditions that represent the complexity of actual cellular systems. NANCY utilized this simulation tool to conduct extensive evaluations of the performance of the B-RAN system and also tested various configurations to identify the trade-offs involved in designing the B-RAN system. It also provides an environment to evaluate reliability, throughput, latency, power consumption, and other critical performance parameters.

1.2. Relation to Other Deliverables

This deliverable builds upon previous research and findings within the NANCY project as follows:

- D2.2 “NANCY Experimental-Driven Modeling” developed a theoretical framework for a blockchain and established guidelines for designing distributed ledger systems optimized for use in wireless networks. D2.4 tests the theoretical frameworks in the simulation environment as well as through experimental measurements and compares them to each other in terms of their performance.
- D2.3 “NANCY Network Information Framework” was responsible for developing the AI predictive models (outage probability and coverage probability predictive models), which were based on feature extraction from experimental data. The predictive models are integrated in the simulation environment, and the performance of the entire system is evaluated.

By integrating the results of T2.2 “Experimental-driven B-RAN and attacks modelling” and T2.3 “Network information framework development” into an end-to-end framework, this deliverable presents a significant improvement over evaluating individual components separately. Unlike evaluating the AI models as applied to static datasets or evaluating the performance of blockchains in response to synthetic transaction patterns, this deliverable allows us to evaluate the operation of both of these components combined under real network conditions, while also illustrating emergent design guidelines and trade-offs.

1.3. Structure of the Deliverable

The rest of the document is structured as follows:

- **Section 2 – Simulation framework**: This section describes the ray-tracing simulation environment created using NVIDIA Sionna. It includes the geographic and environmental modelling that bounds the simulations in realistic conditions, the network topology and deployment strategy, the initial location and mobility profile of the user equipment, the temporal mechanism for the evolution of the environment, the inclusion of reconfigurable intelligent surfaces (RIS), the methods of ray-tracing and channel calculations, and the validation of the simulated physical phenomena. The main goal of this section is to describe the data gathering and processing mechanisms to extract key performance indicators (KPIs) that are used as features for the AI models of T2.3.
- **Section 3 – Intelligent network management**: This section delves into three AI models that constitute the intelligent network management processes of the B-RAN system. The outage probability predictive model learns to identify UEs that are at risk of losing service and is used to proactively initiate the handover process. The coverage probability predictive model assesses the spatial conditions of coverage and the likelihood that a UE will move to an area without coverage from the connected BS. The spiking neural network (SNN) model addresses the network wide optimization problem of identifying which BS to associate with a given UE to maximize overall network capacity while satisfying global constraints. The input features, processing method and quantitative performance results of each model are compared.
- **Section 4 – Blockchain**: This section describes the theoretical B-RAN framework and its integration with the Sionna simulation environment. It analyses how to select optimal system parameters to enable the blockchain to support the operation of wireless networks under varying operational scenarios. It also provides comparative results between the theoretical, emulation, and experimental deployments of the blockchain, showcase how the former can be used to model any deployment.
- **Section 5 – End-to-end performance assessment**: This section presents performance evaluations of the end-to-end system that incorporates ray-tracing simulations, AI models, and blockchain. It investigates the temporal dynamics of the system, quantifies the impact of delays in the AI processing workflows, studies how the delay associated with verifying blockchain transactions impacts the number of dropped calls and studies the interactions of multiple configuration parameters. The results presented in Section 5 demonstrate the benefits offered by the B-RAN architecture and demonstrates the practical constraints that must be considered.
- **Section 6 – Conclusions**: This section summarizes the findings of the assessment of all modules and provides recommendations for future B-RAN implementations supported by simulations and experimental results.

The work presented in this deliverable highlights the fact that simulation is not merely an intellectual exercise; it is an essential tool between innovative theoretical concepts and practical considerations tied to their deployment in real-world conditions. To ensure that the insights into performance garnered via simulation are applicable to operational networks, NANCY ensured a high level of fidelity of the simulation environment (i.e., its dependence on accurate geography, accurate material properties, realistic user behaviour, etc.).

2. Simulation Framework

This section provides a detailed analysis of the ray-tracing simulator that serves as the basis for integrating AI and Blockchain components, validating their functions, and evaluating their performance under varied and close-to-reality conditions.

2.1. Ray-tracing Simulations

2.1.1. Simulation platform and core capabilities

UE-BS simulation environment is implemented using NVIDIA Sionna. Sionna is an Open Source Library of GPU-accelerated code for simulating link-level wireless communications systems [1, 2]. Sionna has provided a substantial advance in wireless simulation technology by providing physical ray tracing capabilities to provide detailed propagation modeling to capture the inherent complexities of real-world wireless environments. Due to the high levels of computation involved in simulating electromagnetic wave propagation through complex environments at the speed needed to be useful, it was previously impossible to accomplish this type of task using traditional CPU-based simulation tools.

This simulation framework provides the capability to model electromagnetic wave propagation through complex environments realistically and includes the ability to take into account physical phenomena that have the greatest and most common impact on cellular network performance. These physical phenomena include multi-path effects, reflections from building surfaces and other structures, diffraction around obstacles that block the direct path of signal propagation and scattering from irregular surfaces and small objects in the environment. The ray tracing engine uses a physically based method to compute the propagation paths by launching rays from transmitters and tracing their interaction with the environment including multiple orders of reflections and diffractions [3]. The physically based method provides channel characteristics simulated using this framework to closely match those observed in actual deployments.

2.1.2. Geographical and environmental modeling

All the simulations in this study were based on models of real geographic maps that accurately depicted the physical nature of urban and suburban areas, and therefore included the authentic terrain features, building layouts, street configurations, and the placement of the infrastructure. The use of actual geographic data ensures that the simulated environment captures the complex and irregular aspects of the deployed cellular network, and not idealized or synthetic scenarios. In addition, the geographic maps included elevation, building footprint with accurate height, street layout, and open space, contributing to the realistic signal propagation characteristics.

In addition to the surface types signal's encounter in the different realistic deployment scenarios, the detailed physical attributes of these surface types were incorporated into the model. The examples include a variety of concrete building types (based on composition and density) and glass windows; each can both reflect radio waves and allow some of them to pass through. Also included metal components (e.g., steel frames and exterior walls), vegetative components (trees, foliage); and attenuate/scatter radio waves. Finally, the model includes ground surfaces such as asphalt and grass. Each of these surface types is represented in terms of their electromagnetic properties, i.e., the complex permittivity and conductivity of the surface, and describe the interaction between radio waves and the surface.

The material modeling system in Sionna offers the exact details of the electromagnetic properties for each type of surface in the environment. The permittivity defines the speed of the electromagnetic wave through the material as well as the portion of the signal that is reflected from the surface, whereas the conductivity is related to the absorption and attenuation of the signal. These variables are dependent on frequency, and the simulation takes that into account to be able to model different bands accurately. This material-aware propagation modeling is capable of producing the realistic signal fading, the reflection coefficients, and the multi-path propagation that is used in actual cellular deployments. The framework treats specular reflections from polished surfaces and diffuse scattering from rough surfaces equally well; thus, it is a complete model of the electromagnetic interactions.

2.1.3. Network topology and base station deployment

Across all scenarios, the simulation first generates multiple instances of user equipment (UE) on the actual map of the area, along with strategically placed base stations (BS). The map, as it can be seen in Figure 1, represents a real scenario in a London area, which was chosen as an example given its iconic location and high traffic scene. In this scenario, the green-shaded regions indicate areas accessible to pedestrians (UEs on foot) and the purple-shaded regions indicate roads accessible to vehicles (UEs in cars). Each area is also labelled with a number or letter, which corresponds to indices in Table 1. For example, road section 1 corresponds to the Charing Cross 1 section, and so on, allowing a clear mapping between the simulated UE locations and the real-world locations. The network topology is planned out to mirror the cellular network architectures of the real world, which means that we have taken into account factors such as coverage, the number of users, and the limitations of the existing infrastructure. Base stations are located at places where it makes sense in terms of typical cellular network deployment patterns that generally work by putting the antennas on the roofs, towers, or the places where the infrastructure that provides the best coverage and the line of sight to the service area is built.

Every base station is equipped with a set of transmission parameters that define the characteristics of its operations. Such parameters include the level of transmission power that is responsible for the extending of the coverage and the strength of the signal, antenna configuration, which indicates the type, the gain, and the radiation pattern of the antenna that is used for the transmission, frequency allocation, which tells the portion of the radio spectrum from where the base station is functioning and beamforming, which is used for advanced antenna systems. The antenna radiation patterns can be as simple as one for omnidirectional antennas that provide uniform coverage in all directions or directional antennas with specific sector coverage; however, they can also be complex multi-element arrays capable of forming adaptive beams.

It is worth mentioning that the base stations are being allocated certain fixed positions that are the same throughout the simulation as a representation of the cellular infrastructure that is permanent. The installation of the inter-site distance, coverage overlap, terrain elevation for the line-of-sight propagation, and expected traffic density to deploy resources where user demand is highest are the things that the placement strategy takes into account. The mobile users' dynamic actions can be compared with this static infrastructure serving as a stable reference framework.

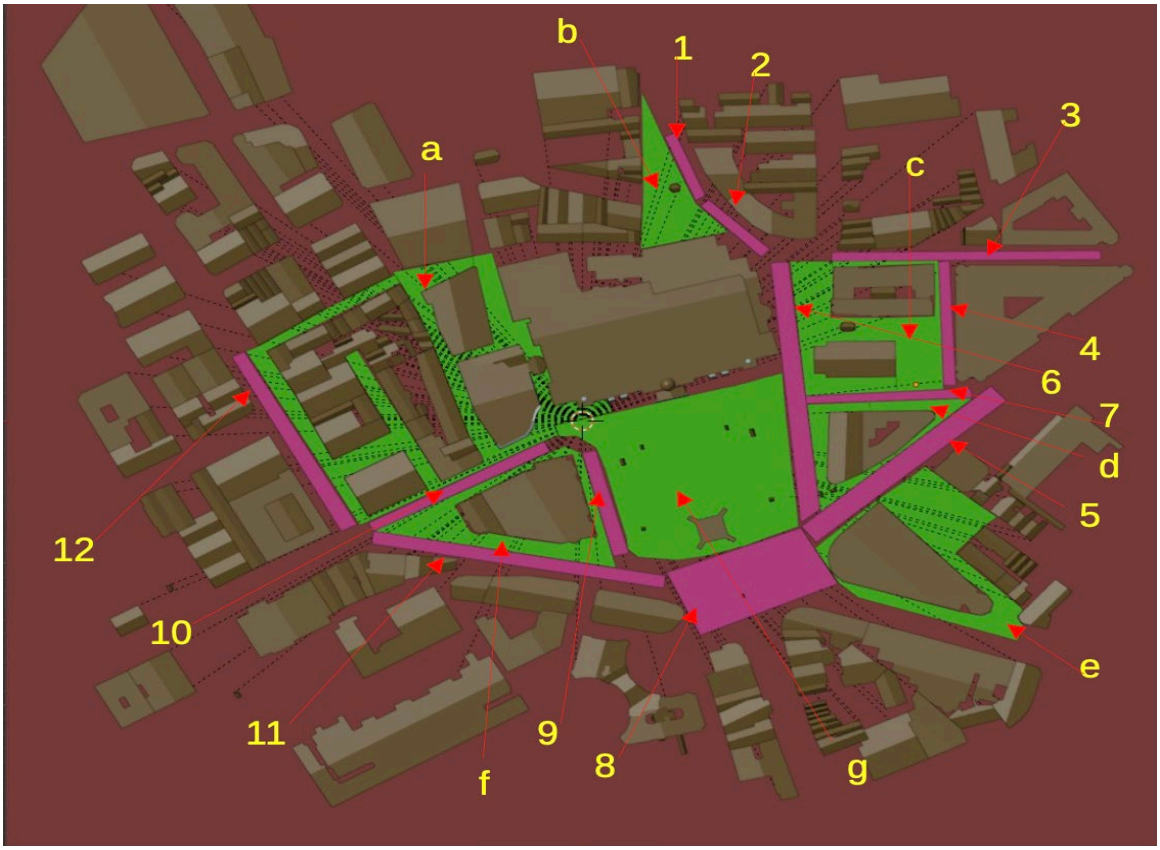

Figure 1. Real-world map with defined user movement areas

Table 1. Region Index Mapping

Index	Name of the area
1	Charing Cross 1
2	Charing Cross 2
3	William IV
4	Adelaide
5	Strand
6	Trafalqar East
7	Duncannon st.
8	King statue
9	Trafalqar West
10	Pall Mall E'
11	Cockspur st.
12	Haymarket st.
a	Haymarket p.
b	Orange st National Gallery.
c	St Martin Page Hotels.
d	Duncannon Strand Trafalqar p.
e	Franklin house.
f	Oceanic Canada.
g	Trafalgars sq.

2.1.4. User equipment initialization and mobility

On the map, each of the forty (40) UEs is given initial random coordinates that are directly related to the reality of the starting locations. These could be streets, buildings, or even parks that are typical places where users might be. Depending on the simulation scenario, the initial distribution of the UEs may also have different patterns. In such a way as to cover the service area, one could have a uniform random distribution of UEs, a clustered distribution like that of the user activity at the hotspot or structured patterns that go along the streets and pedestrian pathways. Initialization of this process guarantees that the simulation starts with a spatial distribution of the users that is not far from reality.

Besides positioning at the start, a random mobility pattern is also assigned to each UE that would realistically demonstrate the user's movement in the service area. The mobility models in this sense would be the cellular networks to be changing continuously as the users move, which means that new connections will be established, maintained, handed over, and terminated. The mobility patterns may simulate different behaviours of movement, for instance, random waypoint models where users move toward a destination that is randomly chosen, random walk models that can be used for less directed movement and the models which are more following the layout of the streets or buildings. The speed and direction of movement may change from one user to another and that can mean different transportation modes such as pedestrians, vehicles, or even stationary users.

2.1.5. Simulation dynamics and temporal evolution

At each discrete simulation step, time is updated and every UE performs its own mobility profile, which it has been assigned, hence their geographical location on the map is constantly being updated. The simulation is done in time steps that are small enough so that changes in channel conditions and connection states are not noticed, but are big enough to ensure the computational efficiency is still there. An example timeseries representation of the received RSS between a BS and a single UE is depicted in Figure 2. While moving, the system constantly checks which base stations are the closest to each UE and thus can be used as the next serving cells based on signal strength, interference levels, and connection policies.

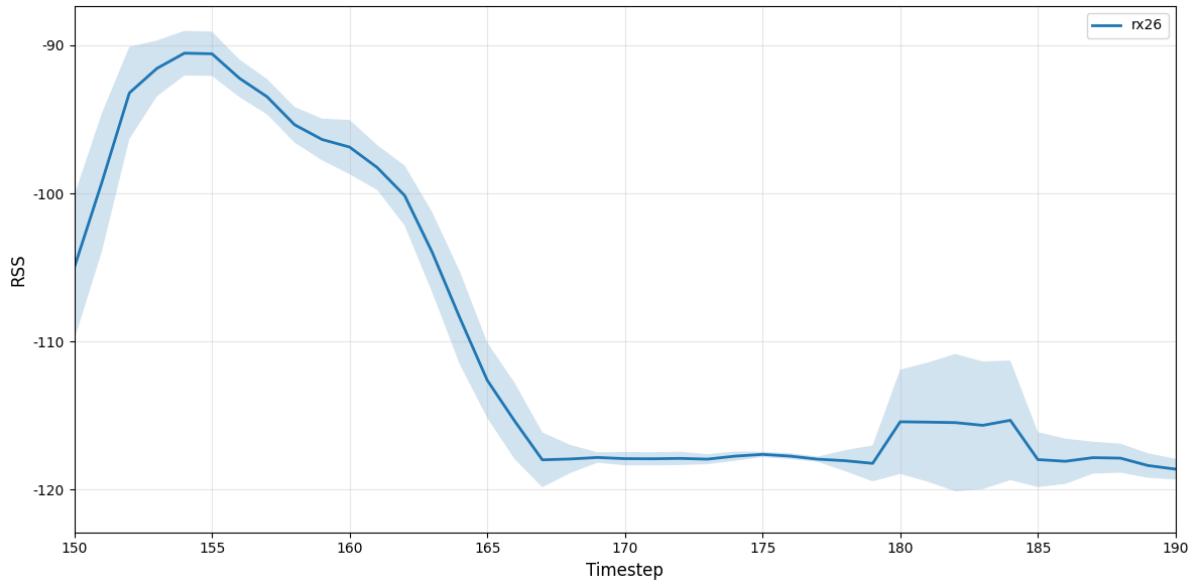

Figure 2. Timeseries of RSS for an individual user

Throughout the simulation, connection states are maintained for each UE, the network also records which base station is currently serving the user and the quality of that connection is monitored.

Connection management covers activities like the selection of the initial cell when the UE is turning on and entering the network for the first time, or when the UE moves out of coverage or ends its session. Those are the connection state changes triggered by signal quality metrics measurements, and they follow the protocols that are very close to the cellular network behaviour in reality.

2.1.6. Reconfigurable intelligent surfaces integration

Moreover, the Reconfigurable Intelligent Surfaces (RIS) are included in the environment for some simulation scenarios and are deployed at the locations that are most sensitive within the geographical map. RIS technology is a new concept for wireless communications, in which passive or semi-passive surfaces are controllable by electronics to change the propagation of electromagnetic waves [4]. The idea is to use the surfaces, which have the arrays of the tunable components that can change the phase and the amplitude of the signals that are reflected, thus allowing for a wise and bright wireless propagation environment.

As a result, the RIS elements become the passive reflecting surfaces that intelligently direct electromagnetic waves toward user equipment, thus improving signal propagation in case the direct paths are blocked or attenuated heavily [5]. The location of the RIS in the simulation is determined by such factors as finding out the places that have bad coverage because of obstacles, situating the surfaces in such a way as to create good reflection paths toward the places where there are a lot of users, and, lastly, by determining the best direction of the RIS pieces in order to get the maximum out of them [6]. Besides that, the phase changes that are made by the RIS elements can be adjusted to the level that they can actually combine the reflected signals that are coming from the receiver and can thus increase the strength and quality of the signal.

The simulation recreates the wireless behaviour of the RIS by considering each element as a reflector that is not only controllable but also has a specific response in terms of phase and amplitude. Consequently, the simulator determines the new direction of the reflected ray together with the signal features based on the actual setting of the RIS, hence, when the rays meet the RIS surfaces. This provides the possibilities for studying different RIS installation plans and regulation strategies, hence revealing the ability of these surfaces, under real circumstances, to elevate network performance.

2.1.7. Ray tracing and channel computation

Sionna's ray tracing engine determines all the significant propagation paths between the base stations and the user equipment by launching rays from the transmitter locations and following their interactions with the environment [7]. The engine finds different types of propagation paths, such as direct line-of-sight paths which have the highest signal strength when no obstruction is present, reflected paths from buildings and surfaces in which signals are reflected off materials before reaching the receiver, paths reflected by RIS elements that are purposely manufactured to provide coverage extension, and diffracted paths around obstacles which enable signals to reach areas in the shadow.

The ray tracing program relies on the principles of geometric optics to find the directions of rays and uses physical optics to calculate the field amplitudes and phases. The engine checks for each possible path if the ray intersects with the scene geometry, determines the reflection or transmission coefficients at each point of interaction based on the material properties and the angle of incidence, and adds the path loss and the phase changes along the propagation path. The model takes into account the different orders of reflections, as the rays can reflect off several surfaces before reaching the receiver, hence the model fully represents the multi-path propagation.

The simulator compiles the full multi-path channel response for each UE-BS link by summing up the contributions of the traced rays. The channel response is the complex channel coefficients that

represent both amplitude and phase of each propagation path, delay spread parameters that specify the time dispersion of the channel, and Doppler shift effects caused by the UE mobility that result in frequency spreading of the received signal. This complete ray tracing method accounts for the signals reflected from all surfaces and RIS elements, thus providing a very accurate model of the complicated wireless channel that takes into account the constructive and destructive interference of the different propagation paths.

2.1.8. Physical phenomena and realism validation

In the end, the simulation reproduces the physical phenomena that have a major influence on the performance of real-world cellular networks, thus the environment modelled is a true reflection of the behaviour of the actual network. The shadowing effects of buildings on the signal create large-scale variations as the UEs move from the line of sight to non-line-of-sight, thus coverage and connection stability are affected. Besides that, the simulation considers slow fading due to shadowing as well as fast fading caused by multi-path interference, which means the model is now able to capture the entire spectrum of channel variations in reality.

Signal improvements due to RIS installations are reflected by the supplementary propagation routes that are fabricated by these are the surfaces, which show how the passive infrastructure can be used to cover the difficult areas. To do that, the simulation does basically the same thing to the real world which is measuring the strength of the signal, the SINR, and the new coverage areas. Also, interference patterns from the neighbouring cells are modelled by the simulation of different base stations simultaneously, quite differently from the realistic situation where few stations co-exist in the same geographical area and interference scenarios can occur, limiting network capacity.

Such a high level of accuracy and physical fidelity means that the simulated network environment is almost the same as those actual cellular deployments, considering not only the propagation mechanisms but also interference dynamics and coverage challenges of the real networks. The benchmarking of simulation accuracy can be done through the comparison of key performance indicators between the simulated and the measured data from the existing deployments. This stable base is a source of trust that the result of the feature extraction and the AI model training using simulation data will have effectiveness in the real-world applications, thus will open the way for the implementation of the algorithms and optimization strategies that will improve the actual network performance.

2.2. Collected Metrics and Preprocessing

The above-described simulation creates a changing environment with UE-BS pairs for real-time connections that will be used for feature extraction and analysis later. Not only does the simulation record the connections that happen at each time frame, as can be seen from Table 2, but it also shows the changes of these relationships over time; therefore, it provides a very detailed dataset of how connections differ when users move through the network. The temporal aspect here is extremely important for comprehending the network dynamics, forecasting the connection states, and optimizing resource allocation strategies.

Table 2. Collected Metrics

Metric	Description
time_step	Time interval of the simulation.
Rx_id	Identity of the receiver.
Rx_x_pos	Position of receiver is x axis.
Rx_y_pos	Position of receiver in y axis.

Rx_z_pos	Position of receiver in z axis.
Rx_rss	Received signal strength at the receiver
Rx_data_rate	Quantifies the current data throughput of the connection.
Distance	Physical separation between the user equipment and the base station measured in spatial units
SINR	Ratio of desired signal power to combined interfering signal and noise power.
CQI	Downlink channel quality measurement.
Buffer	Indicates the downlink data buffer status at the base station
Packets	It quantifies the number of downlink packets being transmitted.
Requested PRBs	Number of physical resource blocks requested for transmission
Granted PRBs	Number of physical resource blocks actually allocated.

While the connections are set up and kept, the simulation can at any time get the connection parameters and thus, describe the link quality and capacity. Along with RSS values that indicate the total power received, SINR that is used for determining data rate, CQI that is used for setting-up data rate due to channel conditions, pathloss that is used for attenuation along the propagation distance, and throughput that is used for data transfer capacity, the parameters can also be obtained continuously in a time series. The regular observations generate time series data that depicts the changes in the metrics brought about by the mobility of the users, variations in interference, and environmental factors.

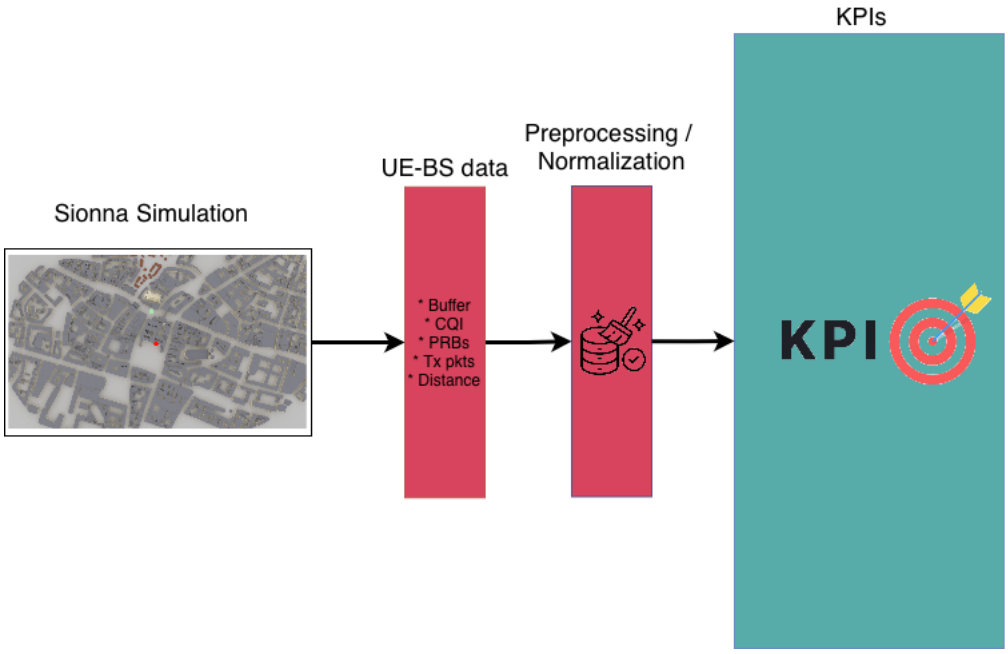

Figure 3. Simulation-specific system model

In addition to this, the simulation also always knows the network's higher-level events and states that will be required for network management and optimization. These are among others: handover events that describe the movement of a UE from one BS to another, radio link failures when the disconnection is caused by bad channel conditions, resource block allocation patterns that demonstrate the use of spectrum resources by users, and interference patterns that disclose the combining of different base stations' transmissions to form an interference. Such a comprehensive data gathering strategy makes

it possible for detailed network behaviour analysis and supports the training of machine learning models with the provision of ground truth labels.

The extracted features are allocated to specific AI prediction models based on their relevance. The RSRQ Prediction Model uses distance and SINR as inputs, while the outage Prediction Model uses DL Buffer, Tx Packets DL, DL CQI, Requested PRBs, and Granted PRBs as inputs. Finally, the Optimal Association Model uses distance, data rate, and SINR from multiple candidate base stations to determine the best UE-BS association. As presented in Figure 3, all extracted features undergo preprocessing to handle missing values, outliers, and noise artifacts. The features are normalized into a consistent scale to ensure machine learning models can process them fairly without bias toward parameters with larger numerical ranges. The cleaned and normalized feature vectors are then stored in a structured format, ready for model ingestion.

3. Intelligent Network Management

This section describes the three AI prediction models that operate on the features extracted from UE-BS simulations. Each model is detailed independently with its inputs, outputs, prediction mechanisms, and performance characteristics.

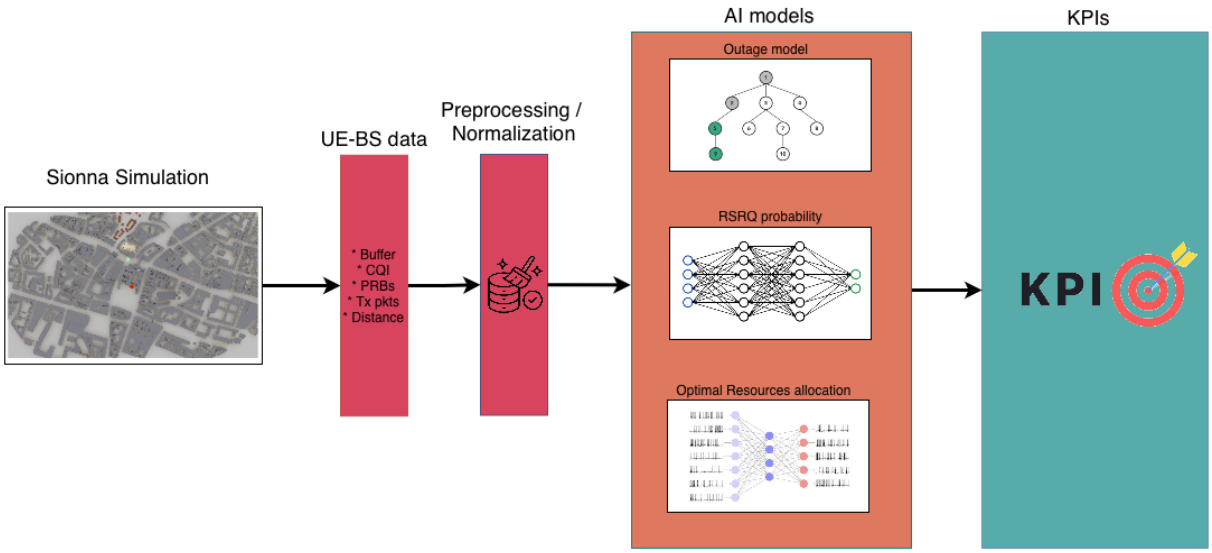

Figure 4. AI-specific system model

A high-level overview of the system model that is considered for the intelligent network management is illustrated in Figure 4. The data that are extracted from the simulations are provided as input to the AI mechanisms in order to perform inference and predict the future behaviour of the network. This provides an edge to the system, which can lead to taking pre-emptive actions that can save possibly dropped calls or sustain peak network performance over time.

Three distinct models have been developed and are evaluated in the following subsections. Specifically, the first two subsections are devoted to the outage and coverage probability prediction AI models developed in D2.3 “NANCY Network Information Framework”. Moreover, the BS-UE association problem is tackled by developing two SNN-enabled approaches based on ray-tracing simulations and assessing their performance in a highly complex scene that includes multiple UES, two BSs and a RIS.

3.1. Outage Probability Prediction

3.1.1. Overview and motivation

The Outage Prediction Model, which has been presented in detail in D2.3 “NANCY Network Information Framework”, based on the Colosseum O-RAN dataset [8], is the key machine learning component that was developed to anticipate transmission outages in cellular networks. Such a foresight capability is instrumental in resolving the arising problem of service continuity in the rapidly evolving wireless communication systems, which is essentially about the quality of mobile users' experience. The model works on features that are already pre-processed from the active UE-BS connections and, hence, makes probabilistic predictions that describe the risk of service disruption of each connection instance in the network.

Network outages are those performance degradations that challenge the operations of the network and the users through insufficient transmission capacity to meet the demand. Outages include, among

others, dropped connections where communication sessions that had been established are abruptly terminated, severe throughput reduction where data rates fall to levels that are unacceptable for the intended service, and increased latency that, as a result, makes the interactive applications unusable. Their occurrence is determined by the availability of resources, traffic load, channel quality, and network configuration parameters. The problem of accurate outage prediction mainly affects the network management objectives, thereby giving the network management a wider range of tools at their disposal to accomplish these goals, such as implementing preventive measures like load balancing across base stations, prioritization of critical traffic flows, and dynamic resource reallocation to connections at risk.

3.1.2. Input feature representation

The Outage Prediction Model accepts a carefully selected set of five input features that collectively characterize the transmission resources and data buffering status of a UE-BS connection. These features are chosen based on their relevance to the physical and logical mechanisms that lead to transmission outages, ensuring that the model has access to information that directly influences outage probability.

- Downlink buffer
- Transmitted packets downlink
- Downlink CQI
- Requested PRBs
- Granted PRBs

3.1.3. Feature interaction and model processing

Each one of these five input features, individually and collectively, influences the likelihood of transmission outages through their interaction and the combined effect on the performance of the connection. The nonlinearity of the relationship between the features is such that it is a complex one and the machine learning methods are needed to discover the patterns that separate stable connections from those in which outage conditions are close. For example, a connection with a very high buffer occupancy but on the other hand also high granted resources and a good channel quality may be able to empty the buffer without the occurrence of an outage while at the same time the same buffer level in combination with the poor quality of the channel and the lack of sufficient granted resources is a very strong indication that service disruption is about to happen.

The machine learning model, which has been trained, takes the five inputs, DL Buffer, Tx Packets DL, DL CQI, Requested PRBs, and Granted PRBs through the representations that it has internally learned, and hence it generates its predictions about the likelihood of the outage. The internal composition of the model has been established on the historical network data, where the strong relationship between the feature values and the subsequent outage events has been made evident. The model during training learns to recognize patterns in feature space that are connected with the probability of high or low outage; thus, it is able to find the complex decision boundaries that separate stable from at-risk connections.

Feature scaling and preprocessing steps are very important in making the model capable of learning from the features that have diverse scales and different statistical distributions. In some instances, the sizes of buffers and counts of packets may be very large, and in addition to that, they may be heavily tailed, whereas CQI values occupy a discrete range that is bounded, and PRB quantities are limited by the bandwidth of the system. If the features are not properly normalized, the ones that have a larger numerical range could dominate the learning process, and as a result, the model would not be able to

find the important patterns in the features that are of a smaller scale. The training stage is characterized by the model being exposed to a multitude of instances of connection situations, each with a label indicating whether or not they have experienced outages within a certain period, and thus the model changes its internal parameters in such a way that the prediction errors are minimized, and at the same time it generalizes to unseen examples.

3.1.4. Output interpretation and performance considerations

The model generates one numerical value for each UE-BS connection that represents the probability of transmission outage happening under the current network conditions. This probability number is a very brief description of the expected risk of outage given a particular combination of resource allocation and channel quality characterizing that connection at the time of prediction. In general, the output is a number between zero and one, where values close to zero correspond to stable connections with low outage risk, while values close to one correspond to connections that are highly likely to experience service disruption if the current conditions persist.

The probabilistic model outputs allow network management decisions to be made in different ways. Instead of binary predictions that simply classify connections as at-risk or stable, probability values allow risk-based prioritization to be implemented, i.e., network interventions are first directed towards connections that have the highest probability of outage. Making decisions from model outputs by threshold selection is a process that involves at least several factors, among which the main ones are low and high thresholds. A low threshold that triggers interventions for modest probability values will detect most potential outages, but at the same time, it may cause unnecessary intervention actions for many of the stable connections, while a high threshold lessens the number of false alarms, but some outages may not be detected; therefore, they cannot be prevented.

The Outage Prediction Model's ability to convey network management ideas is one of its essential features, along with dependability and accuracy. The quality of the prediction is evaluated by a number of metrics that indicate different aspects, e.g., precision, which refers to how many of the predicted events actually took place, and recall, which shows how many of the real events were successfully predicted. Calibration is also an important quality dimension for probabilistic predictions, where a well-calibrated model provides probability estimates that correspond to empirical frequencies. The model's performance is mainly dependent on the standard and representativeness of the training data, which should cover not only the normal operating conditions but also different load levels, channel qualities, and network configurations in deployment.

In Figure 5, we are examining the relationship between the number of predicted handovers and the outage probability threshold across six temporal prediction horizons within the Sionna simulation framework, where an AI model forecasts outage connections between users and base stations. The outage probability threshold represents the confidence level determining how certain the system must be before triggering a handover, while Δt values (from one to six) represent progressively longer time steps into the future, after the prediction. All scenarios follow similar trajectories, characterized by a sharp decline in predicted handovers as the threshold increases from 50 to 60 percent, followed by stabilization at lower values for higher thresholds. At lower thresholds between 50 and 55 percent, scenarios exhibit high predicted handover counts, with delta t values of five and six reaching approximately 17 handovers, while shorter horizons show 13 to 15, reflecting the model's tendency to trigger handovers based on modest confidence levels. As the threshold increases to 60 percent, all scenarios decline dramatically, converging toward approximately five to six predicted handovers and remaining relatively constant through higher values. At the highest thresholds of 90% and above, delta t values of four, five, and six continue declining to 3 and 4 handovers, indicating that longer temporal

horizons become increasingly conservative at extreme strictness levels. A particularly significant observation is the close convergence between delta t of one and six beyond the 60% threshold, demonstrating that the model maintains predictive accuracy even when forecasting multiple time steps ahead, as both immediate and distant predictions identify similar numbers of high-confidence outage events. This temporal stability suggests that underlying features provide reliable outage probability indicators across extended horizons, enabling proactive handover decisions well in advance without sacrificing reliability while allowing system designers to implement conservative strategies by setting higher thresholds to reduce unnecessary handovers.

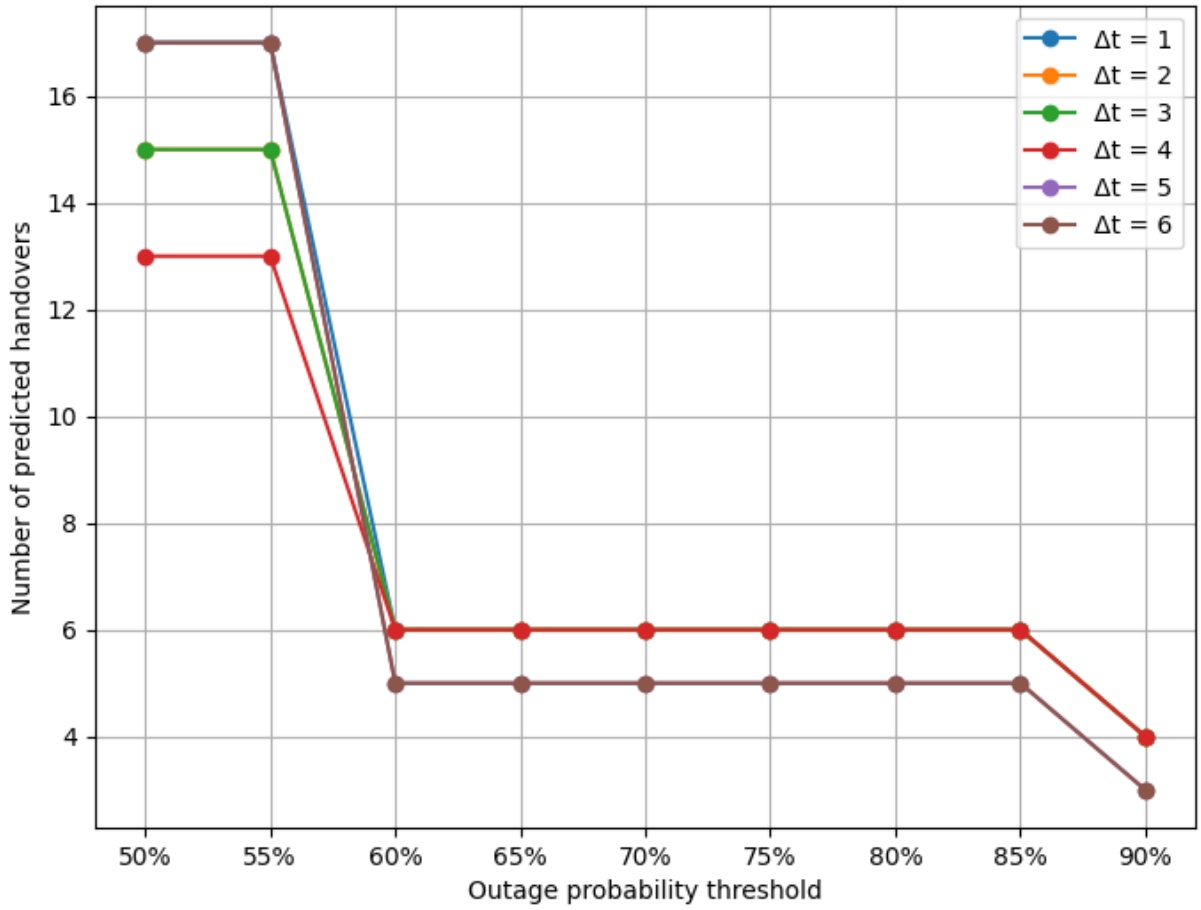

Figure 5. Handover prediction sensitivity as a function of time step size (Δt) and outage probability threshold

3.2. Coverage Probability Prediction

The RSRQ prediction model, detailed in D2.3 "NANCY Network Information Framework", based on the UMU dataset [9], serves as the machine learning portion of this framework for predicting the RSRQ metric in dynamically changing wireless communication systems. The model receives the features, previously extracted by processing active UE-BS connections, and generates probability-based forecasts that describe the anticipated signal quality performance. RSRQ represents an essential performance metric for measuring cellular network performance, since it combines signal strength with the ratio of the desired signal to the interference at the receiving device.

Therefore, the RSRQ prediction model receives the two primary inputs, distance and SINR. Distance describes the geometric distance between the UE and BS, measured in units based on spatial

coordinates that were acquired by GPS. Since distance determines how much electromagnetic signals deteriorate based on the increase of distance as per the path loss models, it can be described as one of the key indicators of signal attenuation due to the connection. SINR describes the ratio of the desired signal power to the total power of interfering signals plus noise present in the received signal. Therefore, SINR captures the effects of interference from neighboring cells and ambient noise on the signal quality, providing important information on the signal quality beyond just measuring signal strength.

These two input features were chosen since they represent the primary physical mechanisms that determine the RSRQ; namely, the interference environment experienced by the receiver and the distance-dependent path loss. The model uses these pre-processed and normalized input values to derive meaningful RSRQ forecasts. The distance and SINR inputs pass through the learned internal representations of the trained machine learning model to produce forecasts about the RSRQ for each UE-BS connection instance. In turn, the model produces a single numerical output value that represents the forecasted RSRQ value for each UE-BS connection. The model derives the numerical output value from the learned patterns and encodes the expected signal quality as a function of the specific distance and SINR of the connection.

The single numerical prediction value generated by the RSRQ prediction model, which represents the forecasted RSRQ for the specific distance and SINR input values, is then mapped to discrete quality categories. These quality categories provide a direct, actionable interpretation of the expected signal quality and are categorized as follows:

- Weak: When the prediction value is less than a predetermined lower threshold, this indicates that the signal quality is severely degraded, resulting in a connection that is generally unreliable and susceptible to disconnects and transmission failures.
- Medium: The prediction value lies in the lower-to-mid range and represents an acceptable but sub-optimal signal quality that allows for basic connectivity, while limiting the performance of the connection.
- Good: The prediction value lies in the mid-to-upper range and represents a good signal quality, such that most users and their respective activities can use the connection reliably and achieve the necessary data rate.
- Very Good: The prediction value lies in the upper range and represents an exceptional signal quality with a strong signal and little interference to allow high-speed data transfer.

Within the broader system architecture, the RSRQ prediction model operates as an intermediate processing stage. It receives pre-processed feature vectors from the Feature Extraction and Preprocessing component (Section 2.2) that have been normalized and cleaned from raw UE-BS connection data. The model generates predictions for each connection instance, producing probability distributions and associated quality categories. These outputs are then passed to the blockchain recording component for immutable archival, enabling transparent validation of prediction accuracy against actual observed signal quality as network conditions evolve.

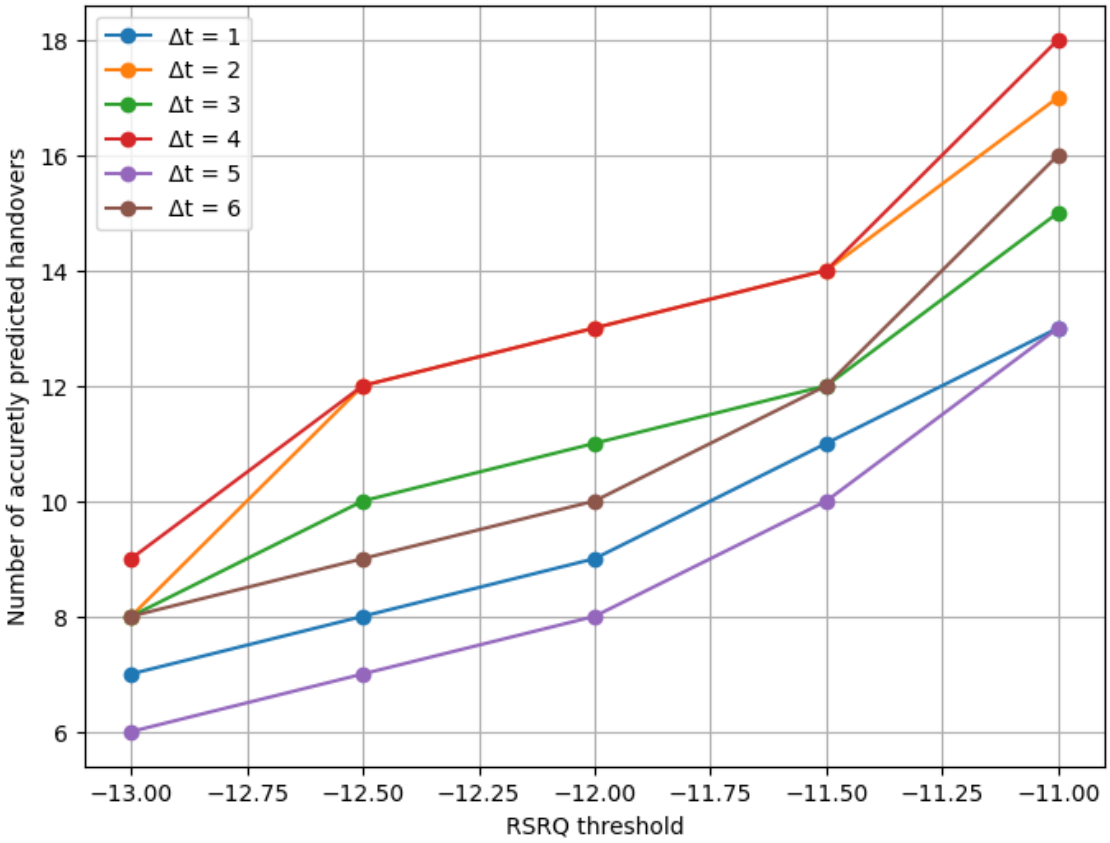

Figure 6. Handover prediction sensitivity as a function of time step size (Δt) and RSRQ threshold

Figure 6 illustrates the relation between the number of accurately predicted handovers and RSRQ thresholds as a function of six different time-based prediction windows within the Sionna simulator framework, for an artificial intelligence model that predicts how well channels will perform between users and base stations. For example, if Δt = 2, this would refer to 2 time-steps after the base station's coverage (RSRQ) prediction. The RSRQ threshold is based upon signal quality characteristics; a higher RSRQ value (above -11) means better channel quality, whereas lower RSRQ values (around -13), indicate weaker channel conditions. Each scenario demonstrates an upward trend, illustrating that as the RSRQ threshold goes from -13 to -11, the number of correctly predicted handovers also increases across each and every time-prediction window. Therefore, the model has established a direct relationship between the increasing RSRQ thresholds and the strictness of the model's criteria in order to classify a call as one that needs a handover; as the RSRQ threshold gets larger, only the "best" possible channel quality is considered acceptable; therefore, more calls will be classified as needing a handover. When the RSRQ threshold was at its lowest value of -13, all of the scenarios demonstrated a very low number of predicted handovers, ranging from about six to nine handovers. Conversely, when the RSRQ threshold was at its largest value of -11, the number of handovers increased significantly to a range of about thirteen to eighteen handovers depending on the Δt value. It is especially noteworthy that all six of the curves have a relatively small amount of variability to one another across the entire range of RSRQ thresholds, demonstrating that the AI model has been able to consistently predict the performance of the network regardless of what temporal window is being forecasted. The small difference between the curves indicates a high degree of temporal stability in the ability of the model to make accurate predictions. The similarity in predictions made for Δt of 1 and Δt of 6 over the entire range of the threshold values, demonstrates that the model is capturing inherent channel quality features that can be used as reliable predictors of the channel quality several time steps into the future, thereby enabling proactive decision-making regarding network operations

without compromising the accuracy of the predictions; additionally, it provides system designers with flexibility to design their systems to operate based upon quality-based handover strategies, by selecting RSRQ thresholds that meet specific service requirements.

3.3. SNN-based Optimal BS-UE Association

This work presents a new SNN-based approach [10] for user equipment-base station (UE-BS) association in non-terrestrial networks (NTNs). With the introduction of UAV's in wireless networks, the system architecture becomes heterogeneous, resulting in the need for dynamic and efficient management to avoid congestion and sustain overall performance. The presented framework compares two SNN-based optimization strategies. Specifically, a top-down centralized approach with complete network visibility and a bottom-up distributed approach for individual network nodes. The SNN is based on leak integrate-and-fire neurons with temporal components, which can perform fast and efficient event-driven inference. Realistic ray-tracing simulations are conducted, which showcase that the bottom-up model attains over 90% accuracy, while the top-down model maintains 80-100\% accuracy. Both approaches reveal a trade-off between individually optimal solutions and UE-BS association feasibility, thus revealing the effectiveness of both approaches depending on deployment scenarios.

3.3.1. Background

For the advancement of state-of-the-art networks, innovative strategies in mobility management are required to cope with the data surge as well as the variety of available services [11]. Since next-generation networks aim at improving connectivity through NTNs [12] and heterogeneous networks (HetNets) [13], proper association between user UEs and base stations (BSs) becomes critical for ensuring high system performance, ensuring quality of service guarantees, and avoiding congestion and interference. Utilizing AI for solving the UE-BS association problem improves the achievable throughput, increases accuracy, and increases the network adaptability, all of which are important in ultra-dense networks [14].

Adaptive user association is important in the field of wireless communications because it can help to maximize the throughput of a given network while minimizing the total energy used by all BSs, as well as decreasing a network's carbon footprint, as stated in reference [15]. Real-time modifications made by AI algorithms to user/BS associations can result in better utilization of each BS (load balancing) and therefore increased efficiency for the overall network [16, 17]. By utilizing renewable energy sources to power BSs, adaptive user association will allow these BSs to serve the greatest number of users possible. These actions will significantly decrease the amount of carbon emissions produced by the network [18]. Additionally, when AI algorithms are integrated into the process of both user association and resource allocation, the resulting increase in energy efficiency will be due to the fact that each BS is operating at its most efficient power level. It also may enable multiple BSs to work together to support a single user, reducing interference and increasing the network performance and ultimately showing improvements in user experience [19, 20].

Both SNNs and Reconfigurable Intelligent Surfaces (RIS) are new technologies that are advancing the fields of AI and wireless communication respectively. SNNs utilize event-driven processing similar to biological neurons, to perform complex computations through rank ordering of parameters reconfigured during the computation process [21]. RIS technology, on the other hand utilizes reconfigurable elements and signal-adaptive techniques to alter the path of wireless signals thereby creating enhanced coverage for wireless networks and/or providing improved wireless connectivity to remote areas having poor signal quality [22]. When combined, these two technologies have great

potential to advance future communication systems. The implementation of SNNs could enhance the dynamic capabilities of RISs, potentially creating an intelligent system capable of adjusting in real time to changing conditions of the user.

3.3.2. Methodology

This section presents the SNN-based intelligent UE-BS association approach for dynamically evolving UAV-enabled NTN systems. Specifically, it covers problem formulation, dataset creation, data preprocessing, SNN design, and training methodology.

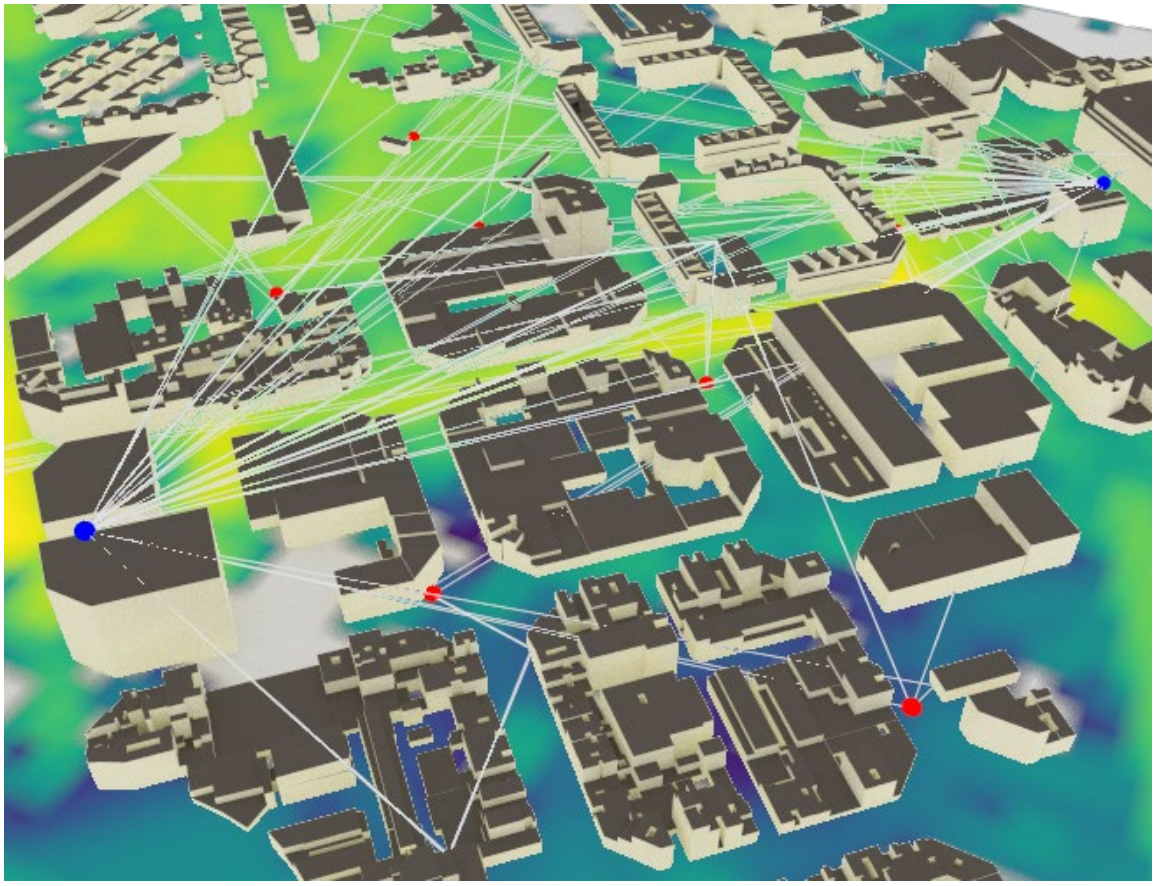

Figure 7. 3D representation of the system model

We assume that an urban geographic area is covered by two BSs and one UAV-mounted RIS that has been deployed ad-hoc for improving the coverage of the network. Both BSs are equipped with directional antennas based on the 3GPP TR38901 technical specification [23], while the UAV- mounted RIS is considered as an ideal phase gradient reflector. Moreover, the area under investigation is populated with multiple users that are assumed to be moving randomly in the environment, with each user equipped with a UE that houses a unidirectional receiving antenna.

The training dataset was generated through realistic ray- tracing wireless communication scenario simulations to ensure that the resulting data accurately reflects real-world channel conditions and network dynamics. The Sionna library was utilized for performing the ray-tracing simulations [24]. The simulation framework models various aspects of wireless propagation, including path loss, reflections, diffraction, shadowing effects, and multipath fading [25]. A 3D representation of the simulated setup is provided in Figure 7. The UEs move with a random walk algorithm. In this scenario, a real-world map of Barcelona was used for the simulation instead of the previous London map. This selection was made to highlight the applicability of the developed simulation in different locations. It was imported into

Blender [26] through the OpenStreatMap API [27]. In addition, real-world positions for the BSs were selected from the CellMapper [28].

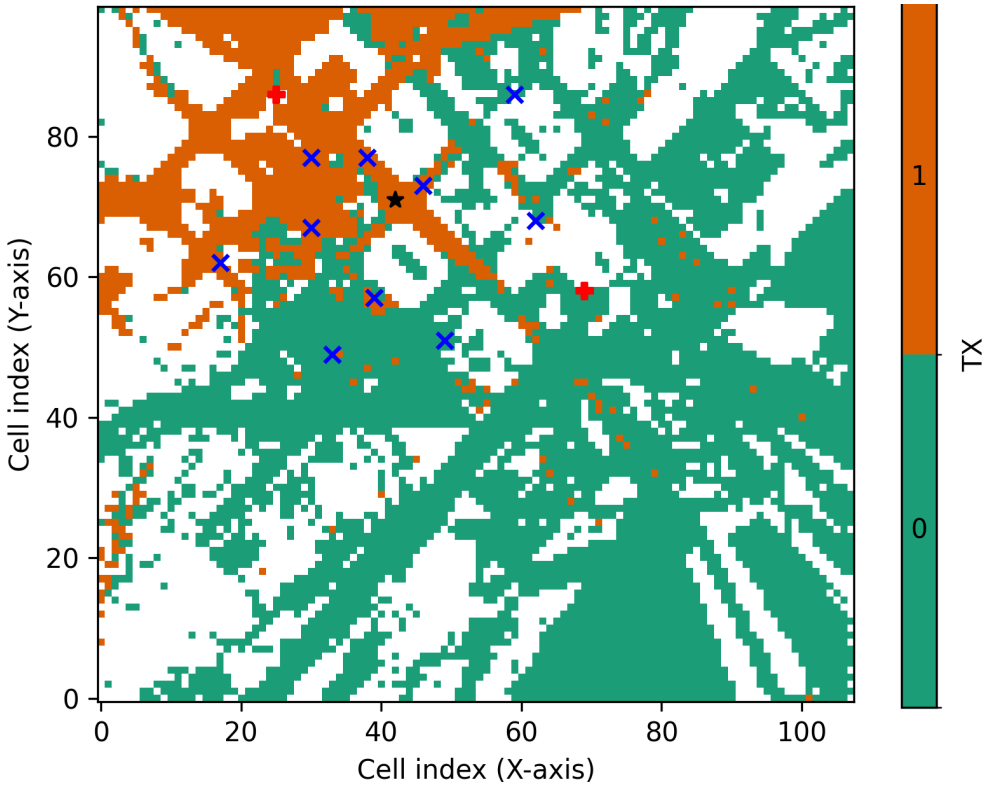

Figure 8. UE-BS association

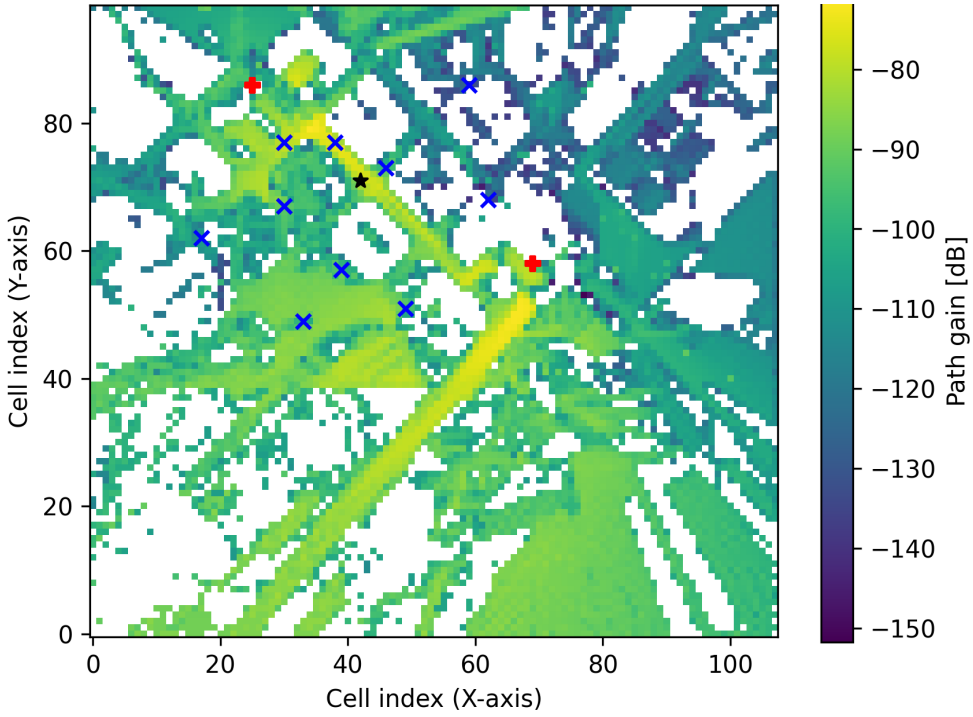

Figure 9. Coverage map.

The simulations were carried out in distinct time steps. Each dataset instance contains the complete data rate matrix representing all possible UE-BS associations within the network scenario. The

coverage map and the UE-BS associations of a single time step are visualized in Figure 8 and Figure 9, respectively. This complete description allows the learning mechanism to include the entire complexity of the association problem. In addition, it captures both the individual dependencies of each Rx association, as well as how they interact in terms of overall effect on the performance of the network. Optimal ground truth associations were determined by brute force optimization, which evaluated all possible combinations of Rx associations to determine which configuration maximized the total network data rate while meeting the operational requirements. This process ensured that the training targets are valid optimal solutions; thus, the trained models may be reliably compared to these targets as an assessment of model performance. All input features undergo a pre-processing stage to normalize the data rates for each scenario of the network. Standardizing the input features ensures that no single UE-BS pair has influence over the learning of the neural network due to scale difference and supports numerical stability in the learning process. Standardizing also preserves the relative relationship of the data rates to each other.

Problem Formulation

One of the most common problems in wireless communications networks is the UE-BS association problem, a combinatorial optimization challenge that aims to maximize the total data rate network throughput while satisfying realistic operational constraints. Given a set of receivers (RXs) and transmitters (TXs), each UE-BS pair exhibits a specific data rate based on channel conditions, interference patterns, and propagation characteristics. The objective is to determine the optimal association of each RX to a TX or RIS such that the aggregate network data rate is maximized. Let $R = r_1, r_2, \ldots, r_N$, represent the set of N RXs and $T = t_1, t_2, \ldots, t_M$ represents the set of M TXs in the network. For each UE-BS pair (r_i, t_j), we define d_{ij} as the achievable data rate between RX i and TX j. The association problem seeks to find a mapping function $f : R \rightarrow T$ that maximizes the total network throughput denoted by

$$max \sum_{I=1}^{N} d_{i,f}(r_i), \qquad (1)$$

$$\text{s.t. } |R \rightarrow t_i| \leq L, \qquad (2)$$

where $|R \rightarrow t_i|$, which denotes the numbed of RXs as- signed to a specific TX, t_i, cannot exceed the threshold L. This constraint reflects practical limitations in wireless systems, including hardware capabilities, power limitations, and interference management requirements. The constraint ensures that UE-BS associations remain feasible under real- world deployment scenarios.

The combinatorial nature of this optimization problem leads to exponential complexity as the number of RXs and TXs increases. Traditional exhaustive search methods become computationally prohibitive for large-scale networks, motivating the development of efficient AI-based approaches that can provide near-optimal solutions with significantly reduced computational overhead.

Spiking Neural Network

SNNs have many attributes that are desirable for use in the UE-BS association problem. They process data in a time-based manner and perform event-driven processing; therefore they incorporate time as a fundamental computation element [29]. Time-based processing is very relevant to wireless communications because timing issues are inherent in all wireless systems and channel changes occur dynamically. The event-driven attribute of SNNs causes them to only activate their neurons when the input signal is large enough to cause a spike. As a result, SNNs consume less power and require fewer computational resources than other types of neural networks; both are characteristics that make them highly attractive for use in applications where energy efficiency is critical; e.g., battery powered BSs or Edge Computing environments. SNNs also inherently represent sparsity, which is analogous to the

sporadic nature of UE-BS association updates; i.e., UE-BS association updates only occur when the UE's channel condition changes significantly. Finally, SNNs are well-suited to capturing complex temporal relationships in the data that feed-forward neural networks cannot; and, in the context of UE-BS association, capturing temporal relationships will enable modeling of the changing nature of channel conditions and the interdependencies among UE-BS association decisions made over time.

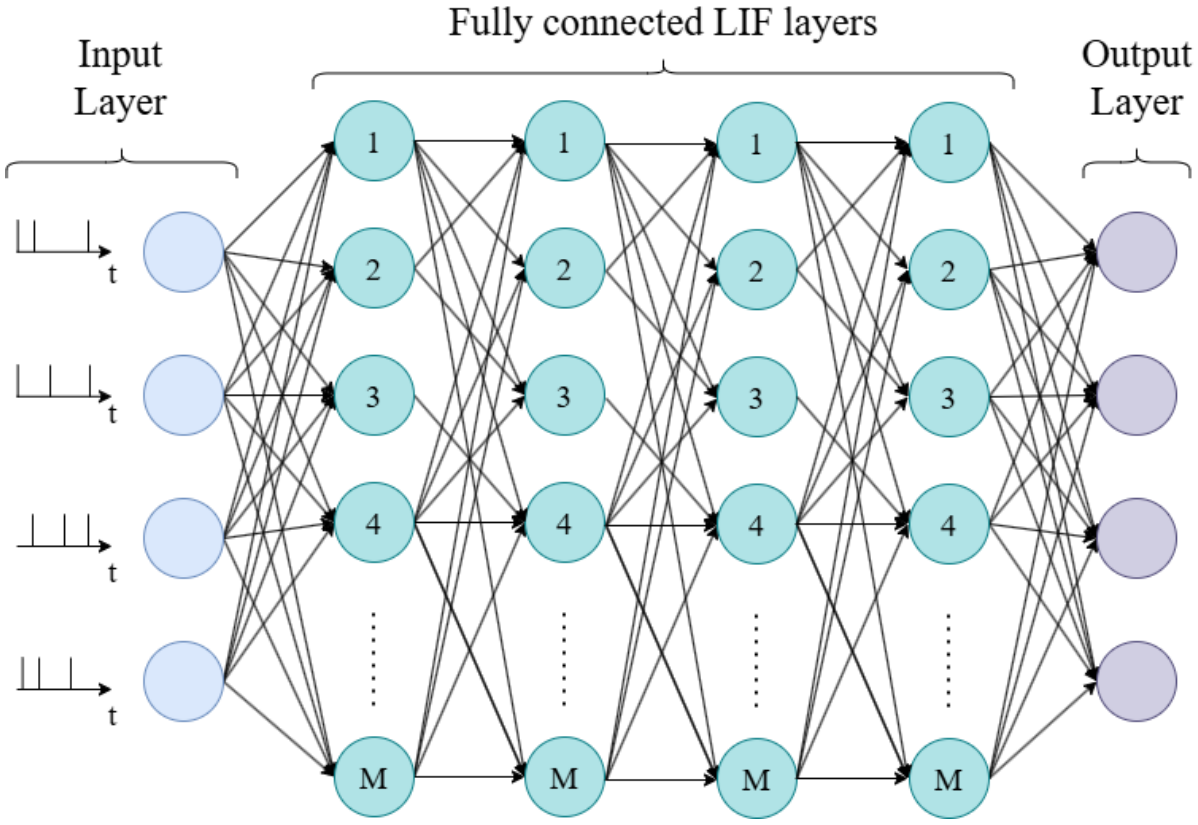

Figure 10. SNN architecture

Architecture Design: A single SNN architecture has been adopted for this research effort as shown in Figure 10. It uses Leaky Integrate-and-Fire (LIF) neurons [30] as its basic computation unit. LIF neurons have membrane potentials that accumulate current over time and fire a spike when the membrane potential exceeds a threshold. Because of the leaky property of the membrane potential, it decays back to zero after firing a spike. Therefore, there is no need to store previous inputs indefinitely. A deep network consisting of multiple fully connected layers with LIF neurons provides a rich non-linear mapping of the input data rates to the optimal UE-BS associations. Dropout regularization is applied to each layer to avoid overfitting and to enhance the generalization capabilities of the SNN. Each layer randomly disables a portion of the neurons during training. By doing so, the network develops robust internal representations that do not depend on specific neurons being enabled. The input layer accepts the flattened data rate matrix of all possible UE-BS associations. The hidden layers abstract the features received in the input layer, increasing the level of abstraction at each subsequent layer. These layers enable the network to learn both local patterns within individual UE-BS pairs and global dependencies across the entire network. The output layer produces the association probabilities for each RX. The dimensionality of the output layer matches the number of possible UE-BS associations. Temporal processing is achieved by iteratively processing the same input over multiple time steps. During each time step, the same input is processed through the SNN. The LIF neurons accumulate and process the input information temporally. The membrane potentials of the output neurons are monitored over all time steps. The final membrane potentials are used to predict UE-BS associations.

Loss Function Design: The training objective combines multiple loss functions to optimize the UE-BS association problem and enforce realistic operational constraints. Cross-entropy loss is used to encourage the SNN to predict the correct association based on the ground truth optimal associations, which were derived through brute force. The second loss function includes a constraint penalty term to discourage solutions that violate the realistic constraint that no more than x RXs can be associated with one TX. The penalty weight is adjusted to sufficiently penalize violations of the realistic constraint without overwhelming the primary optimization objective. The combined loss function balances the competing objectives through careful weighting and encourages the SNN to predict high-quality associations that are both near-optimal in terms of data rate and feasible under realistic constraints.

Integration Approaches

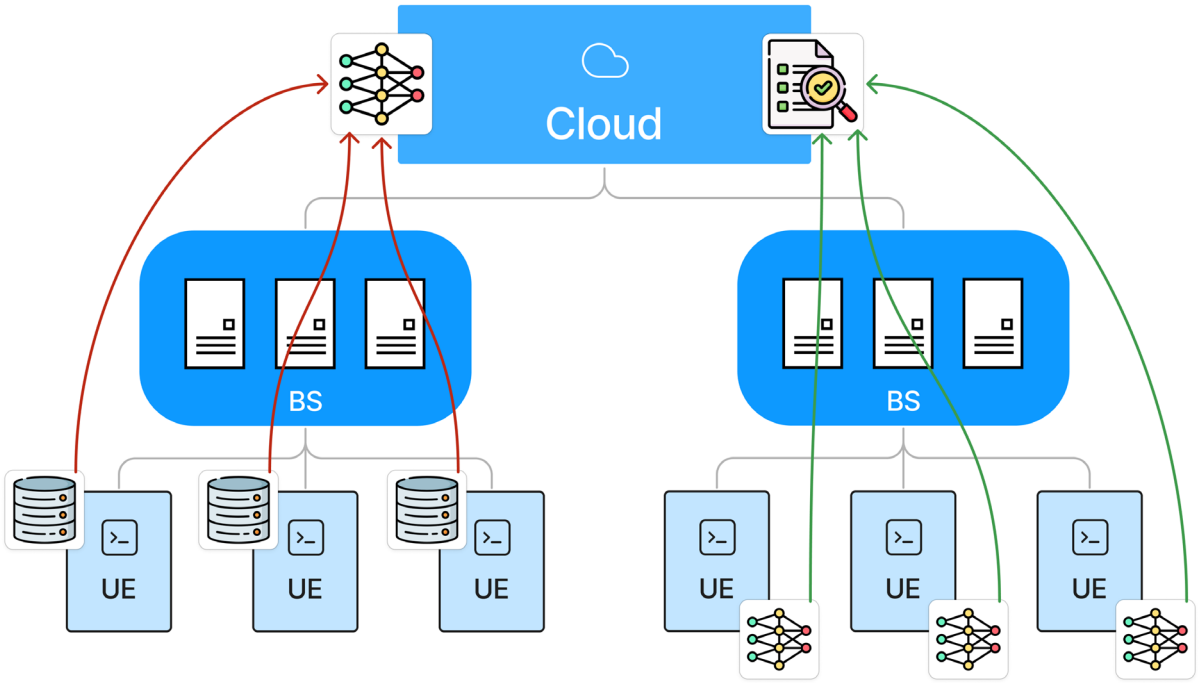

Figure 11. Top-down and bottom-up approaches

Two different integration approaches were identified as having great promise for solving the association problem; namely Top-Down and Bottom-Up approaches (Figure 11).

Top-Down Approach: The Top-Down approach represents a centralized approach to the UE-BS association problem. The Top-Down training approach considers the entire network as input for each training iteration. With the full network considered, the model understands how individual association decisions affect the other RXs and TXs in the entire network. The Top-Down model identifies those scenarios where certain association choices lead to cascading effects on the overall network performance. Due to its access to the full network, the model is able to effectively enforce realistic constraints and learn to effectively satisfy them. The training process for the Top-Down model gathers information from all network nodes at each simulated time step and transmits this information to the central location where the SNN model resides.

Bottom-Up Approach: The Bottom-Up approach represents a distributed approach that is applicable to the individual UEs, where the UEs have limited knowledge about the network. Unlike the Top-Down model, the Bottom-Up model does not take into account the connections to the other UEs. The Bottom-Up method has removed the ability to consider cross-RX dependency and to enforce global constraint satisfaction from the model. The Bottom-Up model utilizes the same dataset as the Top-Down model; however, it uses the same dataset to train the model in a way that focuses on individual

RX decision-making rather than overall system decision-making. The bottom-up model utilizes the same SNN architecture and neuron types as the top-down model; however, the input dimension differs between the two models. The training process utilizes adaptive learning rate optimization, to allow for optimization of the model. The learning rate schedule includes plateau-based learning rate reduction, where once the learning rate reaches a plateau in the validation performance, the learning rate is reduced. An adaptive learning rate allows for more stable convergence in the learning process, while also ensuring that the model does not terminate too early in the training process. The dataset is split into a training set and a validation set at an 80 − 20 ratio to provide sufficient data for both model training and unbiased evaluation of the model performance. The training process monitors both training and validation metrics to determine how prone the model is to overfitting, and to help adjust hyperparameters when necessary. To improve the speed of the training process and to help stabilize gradient estimates, batch processing is used. The batch size is chosen such that a balance is achieved between memory utilization and training stability; therefore, available computing resources can be utilized effectively and consistently to estimate gradients during the training iterations. The training process continues for a specified number of epochs. The use of early stopping, based on the validation performance, prevents the model from becoming overfit. During the training process, model checkpointing saves the best-performing model configurations. Therefore, the final model configuration is representative of the optimal trade-off between the training performance of the model and the generalization capabilities of the model.

3.3.3. Results

The cloud-based, high-performance computing capabilities of Amazon Web Services (AWS), including a GPU-enabled EC2 g6.4xlarge virtual machine, were utilized for model design and training. A key feature of the g6.4xlarge virtual machine is the inclusion of a 22.35 GB, NVIDIA L4 GPU that has 16 vCPU's and 64 GB of memory. This provided ample computing power for the large-scale training of the neural networks required for this research. Furthermore, the use of the GPU enabled significant reductions in training time relative to using only CPUs for each iteration of the temporal loop found in spiking neural networks.

In addition to utilizing computational resources for training the models, an evaluation framework exists to compare predicted values from the SNNs to actual ground-truth optimal values obtained by brute-force optimization. By evaluating the predicted values of the SNNs against the ground-truth optimal values, we have a means to evaluate the quality of the predictions and to perform a quantitative analysis of the trade-offs made between prediction accuracy and prediction efficiency, as well as other metrics.

Inference performance

Figure 12 represents the comparative performance of the top-down and bottom-up SNNs over 100 timesteps. As can be seen in the figure, the two SNNs exhibit different behaviours based on their respective optimization complexities. The bottom-up model (green line) achieves consistently high levels of accuracy (>90%) over the course of the evaluation period. This indicates that the bottom-up model is very stable and it is primarily because of the simplification of the optimization task at the level of the individual RX, which does not require consideration of network wide constraints. On the other hand, the top-down model (blue line) displays greater variability in terms of accuracy than the bottom-up model; specifically, the accuracy of the top-down model varies over an approximately 20% range (i.e., 80−100%). The greater variability of the top-down model in terms of accuracy reflects the inherent difficulty of simultaneously optimizing the individual RX and satisfying the network-wide constraint requirement. The fluctuations in accuracy of the top-down model also reflect the ongoing nature of the constraint-optimization trade-off that is required for the top-down model to generate

accurate predictions. Even though there are differences in terms of the degree of accuracy achieved by the top-down and bottom-up models, both models are able to produce sufficiently accurate results to demonstrate the basic utility of the SNN architecture for solving the UE-BS association problem, regardless of whether the model is simple or complex.

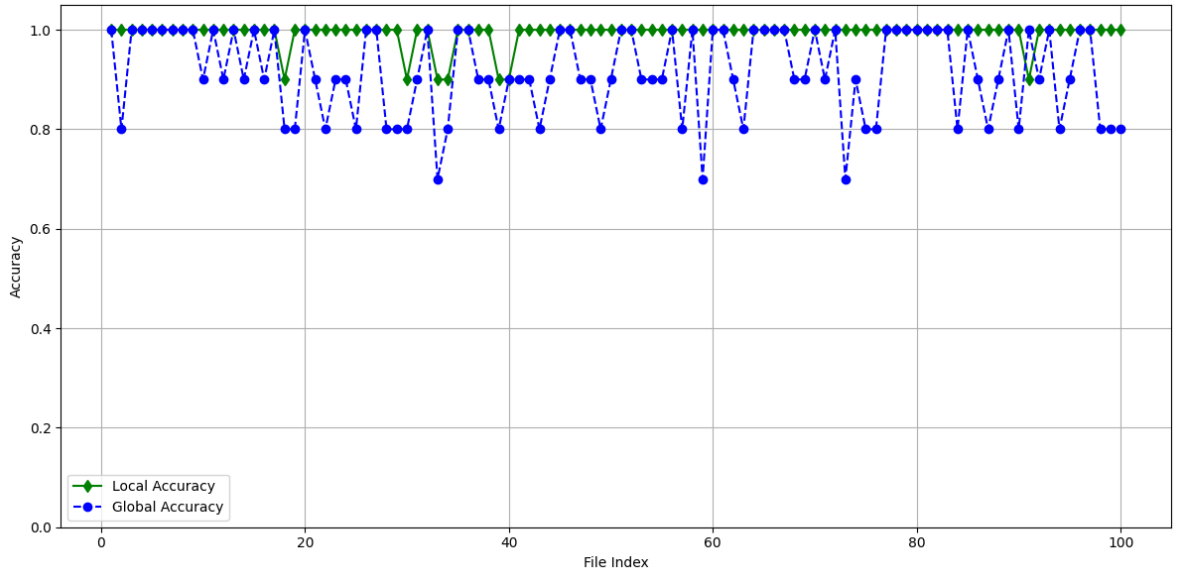

Figure 12. Comparison of the achievable accuracy of the two approaches

Although the models produce very similar results for the UE-BS association problem, their confusion matrices show that there is a trade-off between how well each UE or BS is optimized and how well the network satisfies its constraints.

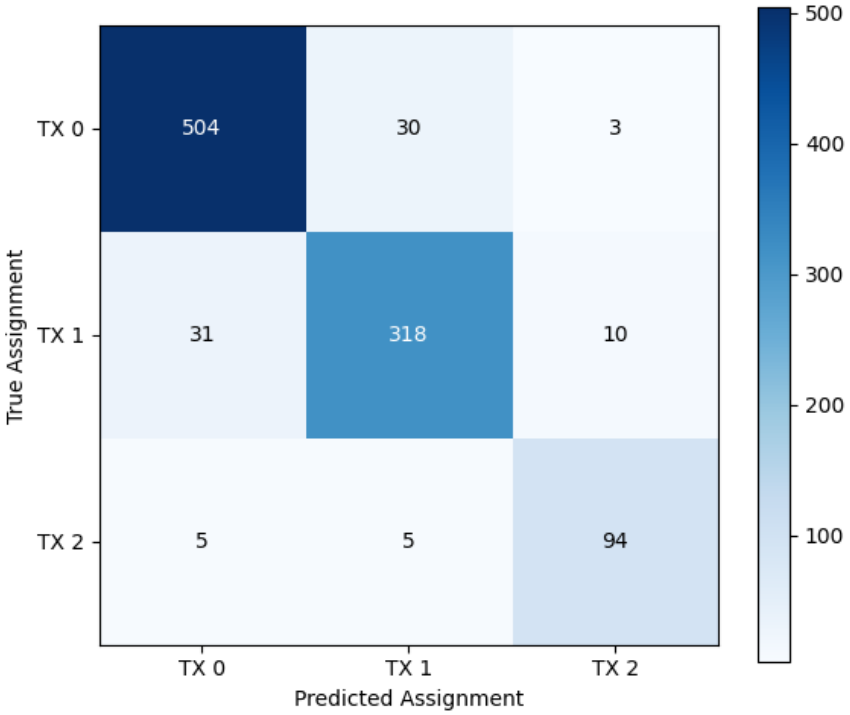

Figure 13. Confusion matrix of the top-down approach

The confusion matrix of the Bottom-Up model in Figure 13 shows that it optimizes each UE/BS almost perfectly, i.e., the diagonal is nearly perfectly filled and the off-diagonal entries are nearly zero; e.g., it

correctly predicts 574, 326, and 93 out of 575 UE/BS pairs' associations. The confusion matrix of the Top-Down model in Figure 14 reveals significant confusion between UEs/BSs' associations and a much larger number of misclassifications, e.g., around 30 misclassifications between UEs/BSs. This reflects the complexity of constraint-aware optimizations, where individually optimal solutions can be at odds with network-wide constraints. The Top-Down model's higher confusion among associations indicates that this represents realistic trade-offs, where misclassifications are likely representations of cases in which individual choices were less than optimal to achieve better overall network performance and constraint satisfaction.

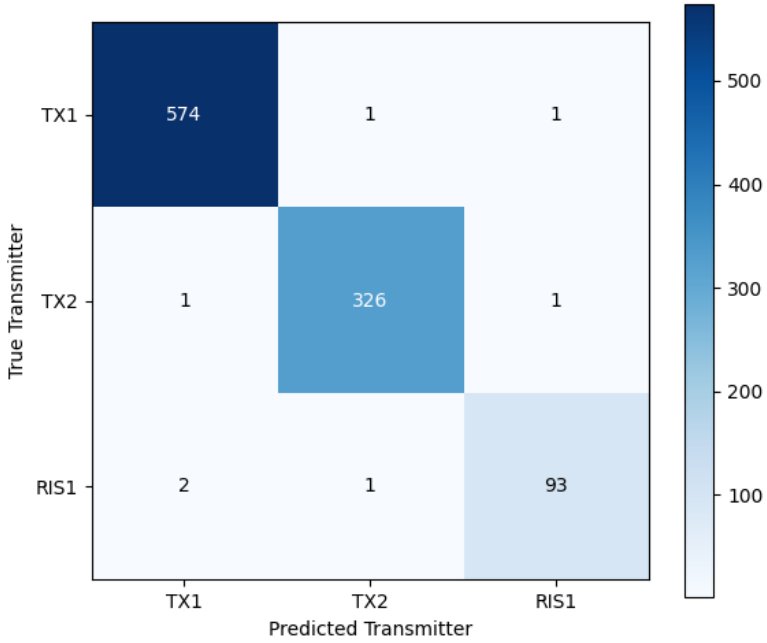

Figure 14. Confusion matrix of the bottom-up approach

Communication performance

Figure 15 shows that SNNs work effectively for the UE-BS association problem. Both the bottom-up and top-down models perform well in terms of predicting data rates associated with each UE/BS pair. However, they have differing objectives and will serve different operational purposes. The bottom-up model (green) predicts data rates that are nearly the same as those achieved by an ideal network that maximizes each UE/BS data rate. Therefore, the predicted data rates closely match theoretical optimal rates across nearly all time steps and only deviate slightly for a few time steps. In contrast, the top-down model (blue) provides a reasonable approximation to the theoretically optimal data rates. It operates with network-wide constraints and therefore does not achieve the same level of optimization as the bottom-up model. However, as seen in the other figures, the bottom-up model solely seeks to maximize individual UE and BS data rates and produces optimal individual decisions. Conversely, the top-down model makes relatively competitive individual decisions while constraining itself to provide a practical network solution. This explains why optimal reference values vary depending on whether the bottom-up or top-down model is used.

Figure 16 and Figure 17 present the top-down and bottom-up models' error distribution from optimal performance, respectively. The former showcases several prominent spikes reaching 300–500 spikes and consistent moderate errors throughout the time steps. In contrast, the latter exhibits most values at or near zero except for a few notable peaks with a deviation of 10 – 60. The error distribution plots for both scenarios reveal different optimization strategies where the top-down approach prioritizes system-wide constraints over UE-specific performance maximization.

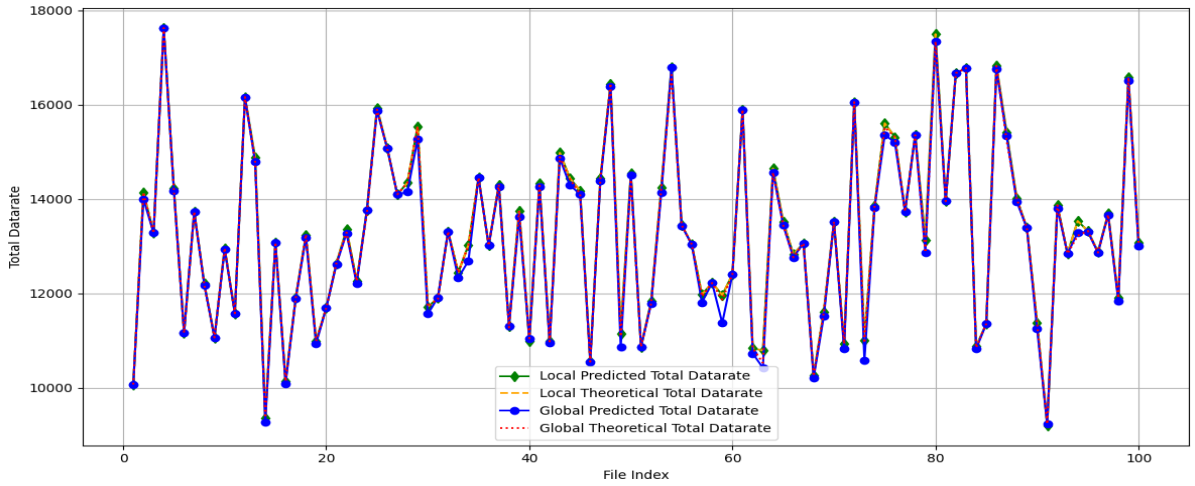

Figure 15. Comparison of the achievable data rate of the two approaches

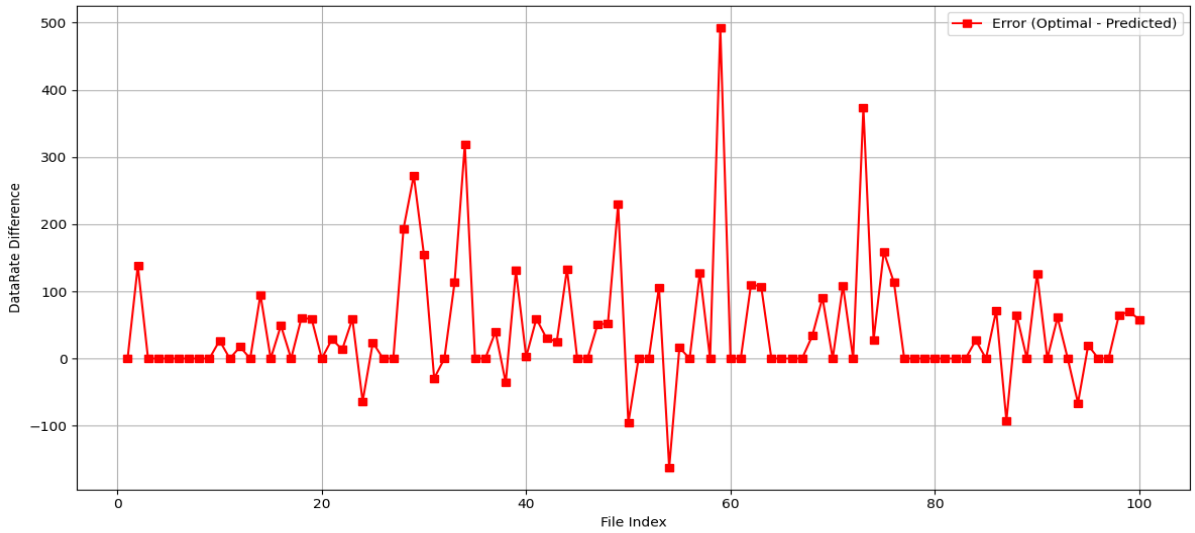

Figure 16. Data rate error of the "top-down" approach

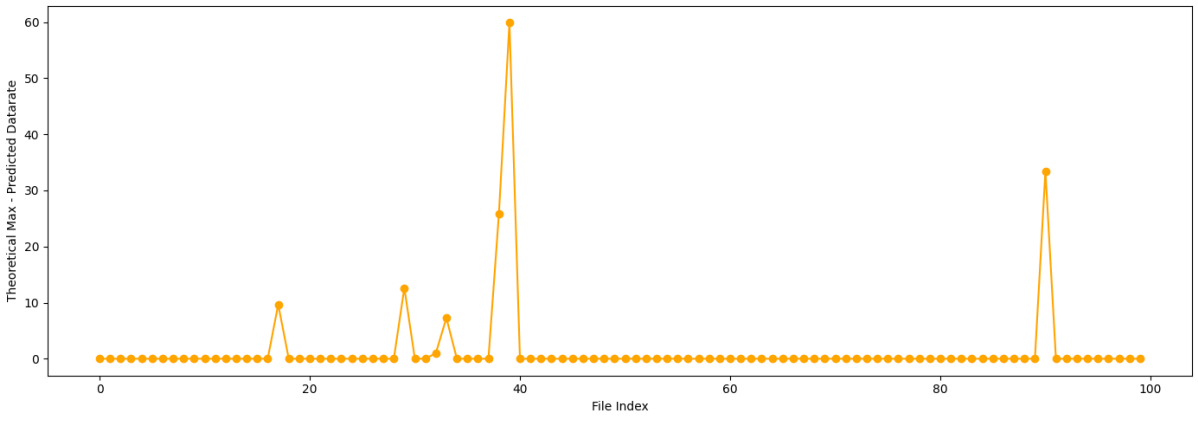

Figure 17. Data rate error of the ”bottom-up” approach

4. Blockchain

This section delves deeper into the integration of the theoretical framework of Blockchain that was presented in D2.2 “NANCY Experimental-Driven Modelling” as well as its practical implementations into the simulation framework.

4.1. B-RAN simulation framework

The NANCY B-RAN architecture is based on the utilization of distributed ledger technology (DLT) that is customized to provide a solution for managing wireless network access [31]. The customized nature of the NANCY blockchain provides complete control over the variables that will directly affect the performance metrics that are relevant to network management applications, and therefore the ability to tailor the DLT solution to meet the needs of a variety of different network environments.

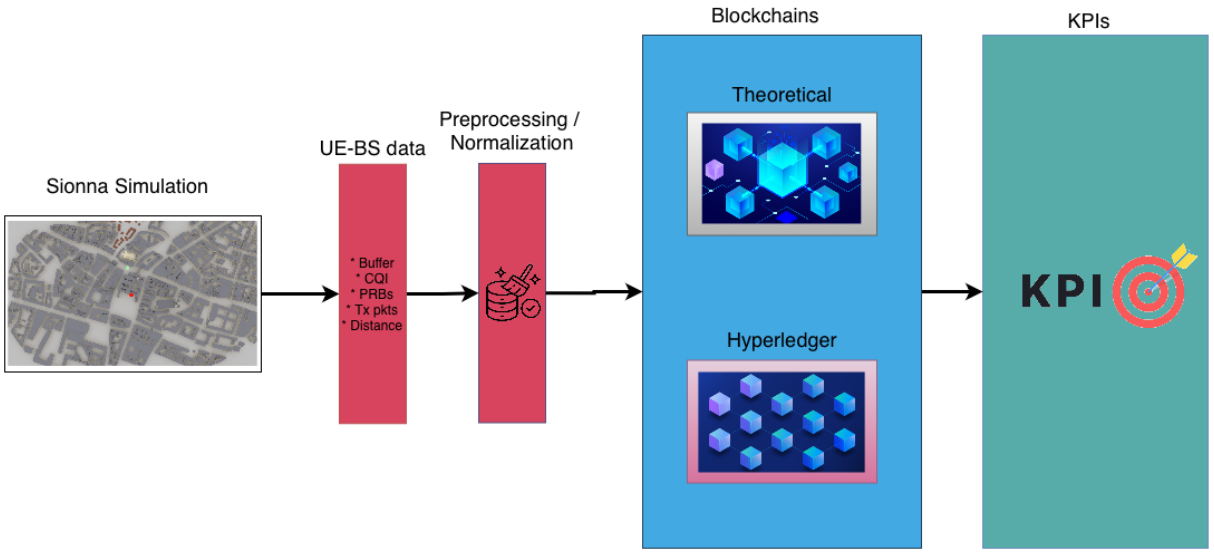

Figure 18. Blockchain-specific system model

Figure 18 presents the blockchain system model that was designed to support configurability and adaptability so that the unique conditions presented by each type of environment can be met. There are numerous types of wireless networks (urban, suburban, rural), all of which present their own set of challenges (network density, traffic flow, latency, etc.). The NANCY blockchain allows the modification of several parameters (degrees of freedom) that are necessary for such a task.

4.1.1. Configurable system parameters

There are many parameters in the blockchain system that are adjustable, and they define how the entire system operates as a whole. The adjustable parameters are an indicator of the flexibility required to make adjustments for optimal operation of most blockchain systems. Each of the parameters is linked to a specific area of performance, and when combined, the interactions between them create an optimal trade-off space for competing factors such as latency, throughput, security, and resources.

One of the most critical parameters that define the temporal behaviour of a blockchain network is the rate at which new blocks are generated, i.e., the block generation rate or the mining rate [32]. The block generation rate directly determines how often new blocks are produced and added to the distributed ledger. Since every transaction must be included in a block prior to being confirmed through propagation throughout the network, the block generation rate is the primary factor that

establishes the minimum delay associated with transaction confirmation. Higher block generation rates will allow for faster confirmation of transactions by increasing the number of opportunities for transaction inclusion, which allows for more rapid responses to network management events that require coordinated actions to occur in a timely manner. Conversely, however, increased block generation rates also increase the computational and communication overheads associated with propagating blocks, validating blocks, and achieving consensus among nodes. Furthermore, fast block generation can also result in increased fork rates in which a large number of nodes simultaneously generate alternative versions of the blockchain. As a consequence, the network requires more sophisticated methods for resolving forks and may be less efficient because it will utilize additional resources to resolve conflicting chain branches.

Additionally, the service rate parameter carries the meaning of the processing power of the blockchain nodes; therefore, the nodes should be able to validate and execute the transactions. This metric illustrates how much computation each node has to carry out the cryptographic operations, state updates, and verification steps, thus processing the arriving transactions and validating proposed blocks. The service rate is the one that directly caps the highest transaction throughput that the blockchain can keep, as the transactions arriving too fast will queue the node buffers, causing the rise of the latency and the possibility of transaction drops if there is no more space for the queue. In wireless network management contexts, the service rate must be adjusted to the expected rate of events in the network that need to be recorded on the blockchain, such as the decision of handover, updates of resource allocation, or agreements on spectrum sharing. Events with a high rate require a service rate as high as they are to prevent the blockchain from slowing down the network management workflow.

The block capacity component is the one specifying the utmost count of the requests or transactions that could be merged into one block. This parameter imposes a fundamental limitation on the momentary throughput of the blockchain system, as it only allows a certain number of transactions to be carried out in each block generation time. The block capacity affects the system throughput, which is a product of block capacity and block generation rate, alongside the block generation rate. That is to say that the block capacity times the block generation rate defines the overall system throughput in the ideal case of continued full blocks. The higher the block capacities, the more the potential throughput could be, but at the same time, the size of the blocks that will have to be transmitted across the network, the computational effort needed to validate all transactions in the block, and the storage requirements for keeping the complete blockchain history will also increase. On the other hand, the smaller block capacities relieve the overhead costs, but they may put a ceiling on the throughput when transaction arrival rates are high. The best block capacity is related to the nature of the network management transactions being recorded, such as their size, complexity, and the validation requirements.

4.1.2. Integration with Sionna simulation environment

The integration of the Sionna simulation environment with the custom blockchain framework results in a single evaluation platform that not only reflects the physical layer changes of the wireless propagation but also the distributed coordination mechanisms accomplished by the blockchain. This approach reveals how well systems based on the blockchain perform in a network subjected to conditions from the real world, considering the complicated interaction of the radio channel properties, the user mobility patterns, and the limitations of the distributed ledger processing that we have in real-life conditions.

The interaction between the two simulation modules is carried out by a request generation mechanism that converts network events in the Sionna environment to blockchain transactions. When the Sionna simulation is going through its discrete time steps and is calculating the channel characteristics for each active UE-BS connection pair, the system is also checking whether these connections are the ones that lead to events for which a record in the blockchain is needed. Each established or updated UE-BS connection leads to the creation of a request that includes the relevant information about the connection state, such as resource allocation parameters, channel quality metrics, and connection identifiers. These requests have the format of blockchain transactions and are sent to the theoretical blockchain system for processing. The pace of the generation of blockchain requests from connection events is the outcome of the dynamics of the simulated network, user mobility, handover frequency, and resource reallocation policies, being some of the factors that determine the transaction arrival pattern experienced by the blockchain.

The simulated system environment provided by the integrated framework enables latency measurement, which takes into account wireless transmission delays, commensurate with the delays incurred by the wireless modes adopted by the Sionna system, and blockchain layer processing delays. The latency calculation starts with the occurrence of a network event simulated in the Sionna system and the creation of a corresponding request to enter the blockchain system, proceeding to the submission of the request into the blockchain system and queueing at nodes on the blockchain system, then to the inclusion of a transaction into a block and broadcast of such a block on the blockchain system, and finally ending when the confirmation of the transaction reaches a requested confirmation level, which qualifies such a transaction as finalized. By tracking timestamps at each stage of this process, the simulation quantifies the total latency from network event occurrence to blockchain confirmation, providing insights into whether the blockchain system can meet the timing requirements of real-time network management applications. The latency measurements capture the impact of blockchain configuration parameters on system responsiveness, revealing how choices regarding block generation rate, block capacity, and service rate affect the time required to coordinate network management decisions across distributed entities.

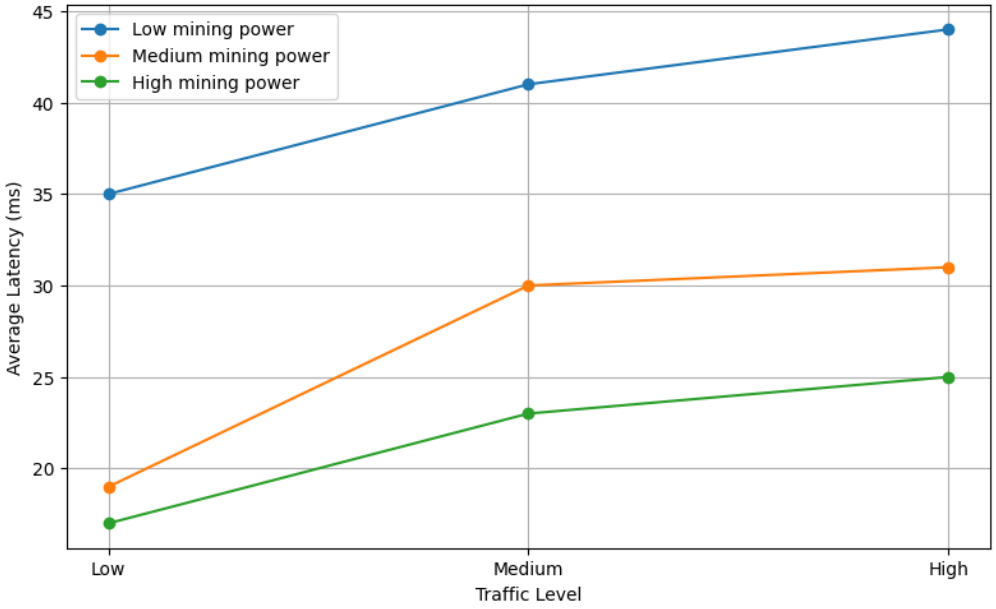

Figure 19. Latency performance under varying traffic and mining power conditions

In Figure 19, the relationship between average latency and traffic level is investigated for three different mining power scenarios within the Sionna simulation scenario, incorporating the theoretical

integration of blockchain. From the results, it can be seen that the three different scenarios of low, medium, and high mining power follow similar patterns for the traffic levels, reflecting the similar structural properties between traffic level and latency response regardless of mining power. It is worth noting that the medium and high mining power scenarios are remarkably similar in their patterns compared to the low mining power scenario in each level of traffic, except that the latter scenario indicates moderate increases in the level of latency despite the lower level of traffic, meaning that beyond this point, mining power may be of little effect in further improving the level of latency.

Firstly, the convergence of the medium and high mining power scenarios indicates that mining power has indeed saturated at some point and that additional computational power leads to diminishing returns in terms of performance. In regard to traffic progression, it can be seen that in all scenarios, the point at which low and medium traffic converge in terms of latency marks the beginning of increased contention and delays due to traffic control. Conversely, when transitioning from medium to heavy traffic, it can be seen that the slope at which increased traffic causes higher latencies becomes much less pronounced for each mining power scenario, leading towards an aspirational boundary at which point the system can be said to level off in regard to performance. In light of this observation that increased traffic beyond some point leads to ever-decreasing incremental increases in system latencies, it is possible to determine that beyond some point of traffic, increases to traffic level lead to less and less system performance disparity, in turn suggesting that traffic level itself no longer remains the dominant factor in performance at higher levels of operation.

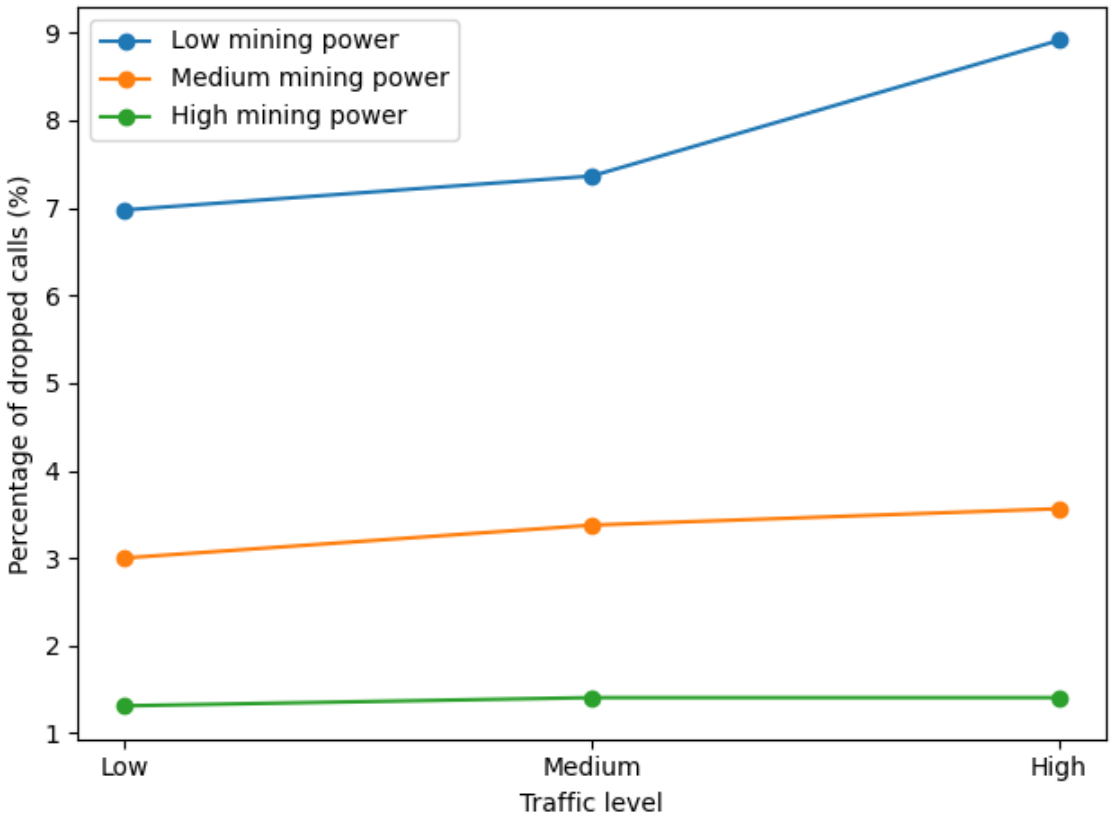

Figure 20. Impact of traffic load on call drop rate across different mining power configurations

Figure 20 compares the relationship between the percentage of dropped calls and the level of traffic across three different mining power configurations in the Sionna simulation environment, with the custom blockchain integration. All three lines in the graph have an upward trend, indicating that as the level of traffic increases from Low to Medium to High, the percentage of dropped calls also increases.

There is no variation in directionality between the three lines; this implies that traffic has a dominant influence over call drop out rates, irrespective of the amount of computational resource allocated to mining. Additionally, there is a clear indication that the two high and medium mining power configuration lines are very closely positioned across the full spectrum of traffic, and that they maintain a very low rate of dropped calls relative to the low mining power line; these lines show less than a 4% rate of dropped calls when the traffic level is high, whereas the low mining power line shows a 9%, 7.5%, and 7% rate of dropped calls respectively, at high, medium and low traffic levels. The large difference between the high/medium and low mining power lines, provide strong evidence for the existence of a "threshold" effect in mining capacity, such that if sufficient mining capacity exists, then adding more computational resources will have little or no effect on call completion success rates. Furthermore, the fact that the high and medium mining power configuration lines converge at certain points indicate that if a minimum level of mining capacity is provided, then there is limited incremental improvement in call completion success rates, with increasing levels of computational resource. The significantly higher rate of dropped calls seen in the low mining power line is likely caused by the longer transaction verification times that exist in computationally resource-constrained blockchain systems, where inadequate mining power results in long validation periods that may lead to call timeouts or abandonment prior to achieving consensus. This temporal bottleneck in transaction processing creates a "domino effect," where legitimate calls are dropped not solely because of network congestion, but rather because the blockchain cannot process the verification procedures within a reasonable timeframe. Additionally, the data illustrates that although all three configurations display an increase in dropped calls as the level of traffic increases, the rate of increase varies significantly among configurations; the low mining power configuration shows a more rapid degradation from medium to high traffic, while the high and medium mining power configurations show a more gradual increase in dropped calls. This variation in the sensitivity of each configuration to changes in traffic level highlights the need for maintaining an adequate level of mining capacity as a fundamental prerequisite for reliable operation, especially in applications where call completion rates directly affect end-user experience and service quality.

4.2. Hyperledger fabric

The B-RAN Model was implemented using the Hyperledger Fabric as a Blockchain in a Comparative Study. This allowed for the deployment of an actual Fabric Ledger (production grade) to test how a commercially available blockchain system would compare with the theoretical custom blockchain implementation that was developed based on the B-RAN model; with respect to performance during the same simulation experiments and AI predictive workloads. To test the performance of Fabric under various configurations and workloads, Hyperledger Caliper was utilized. Caliper is a tool designed to systematically evaluate the performance of a blockchain system by measuring its performance (including transaction throughput, latency, resource usage, and success rate), utilizing different configurations. The complete end-to-end simulation pipeline was then tested across numerous test configurations, including multiple transaction rates, various block sizes, variable number of endorsing peers, and various consensus configurations. The data collected indicated the impact of network traffic patterns, concurrent user load, and limited resources on the blockchain recording performance. Bottlenecks in performance were identified and used to determine the optimum configuration settings for the blockchain to monitor and adjust for changing network traffic patterns. Additionally, the established benchmarks provided a means to directly compare the performance of Fabric versus the theoretical custom blockchain implementation when both were tested under the exact same workload conditions.

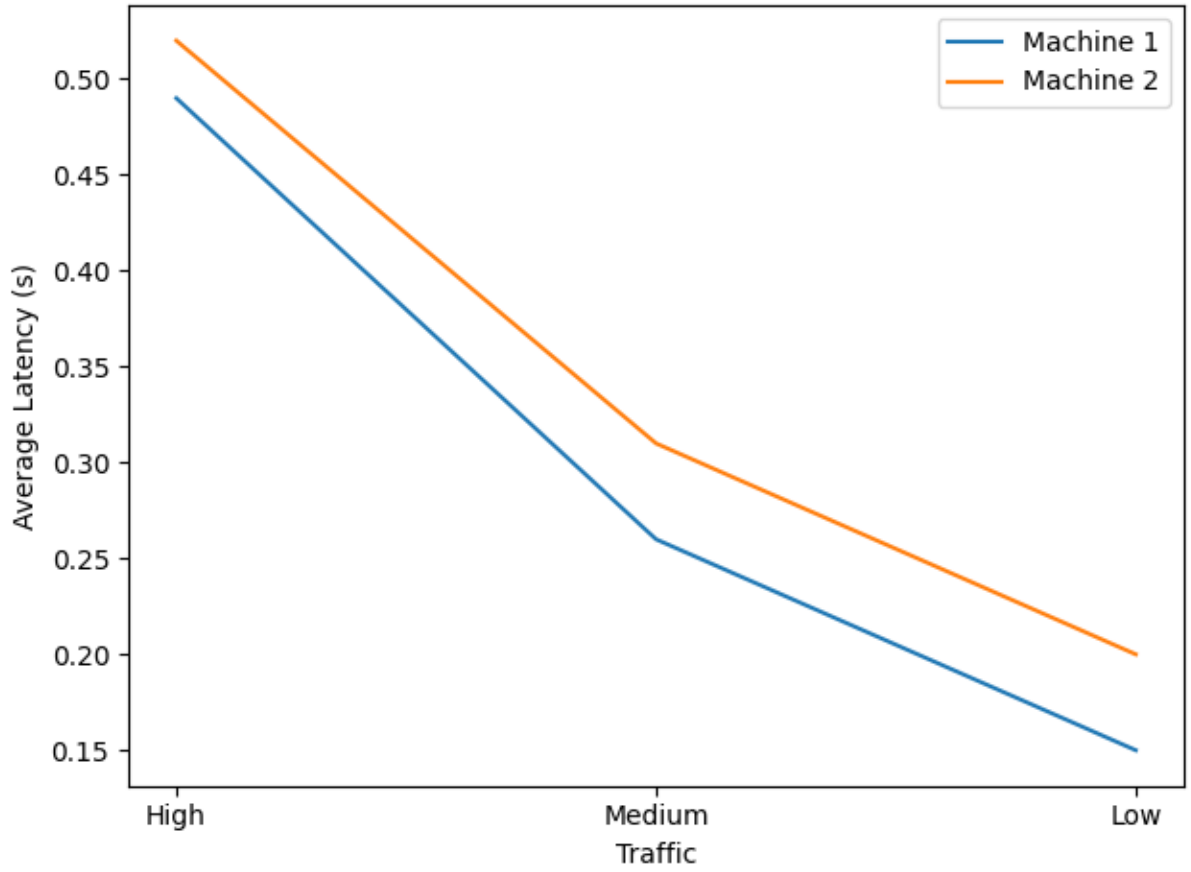

Figure 21. Latency performance comparison under varying traffic conditions

Figure 21 shows the relationship between average latency and traffic level for the Sionna simulation integrated with Fabric, executed on two distinct machines with different computational capabilities. Machine 1 was an AWS GPU-enabled EC2 g6.4xlarge virtual machine with 22.35 GB, an NVIDIA L4 GPU with 16 vCPU's and 64 GB of memory. Machine 2 featured an RTX 3080 GPU with 16GB of memory and 32 GB of RAM. Although the plot does display an inverse counter-intuitive trend, both machines move down in trajectory as the traffic moves from high to low, thus both machines show the same pattern throughout the traffic ranges. For example, the latency values for Machine 2 are reduced from about .52 sec at high traffic to about .20 sec at low traffic, and the latency values for Machine 1 are also reduced from approximately .48 sec to approximately .15 sec over the same traffic range. In addition, the two machines continue on parallel paths for the entire traffic spectrum; Machine 2 has slightly higher latency values across all conditions because it is less capable computationally than Machine 1. The vertical space between the two lines remains relatively steady (.04- .05 seconds) throughout the traffic ranges, which suggests that the differential performance of the systems is proportional to their ability to compute regardless of the traffic conditions. As a result of this, Machine 1, which has better computational capabilities, will process transactions faster, and therefore, the computational capability directly influences Hyperledger Fabric consensus operations, transaction validation rates, and cryptography computation rate. Additionally, there is a greater decline in the steepness of the lines from high to medium traffic as opposed to the slight decline in the steepness of the lines from medium to low traffic, suggesting that the system is more efficient when moving away from resource-constrained situations and the reduction in traffic will yield smaller and smaller reductions in latency. Finally, the fact that both lines have the same shape indicates that the overall relationship between traffic and latency in Hyperledger Fabric is the same regardless of the specifications of the hardware

and that the computational capability only effects the absolute performance levels and not the qualitative behavior of the system. Overall, these findings indicate that Hyperledger Fabric implementations can benefit from having enough computational capability to minimize baseline latency, however the inherent architectural characteristics of the blockchain framework ultimately dictate the overall performance trends exhibited across various traffic conditions.

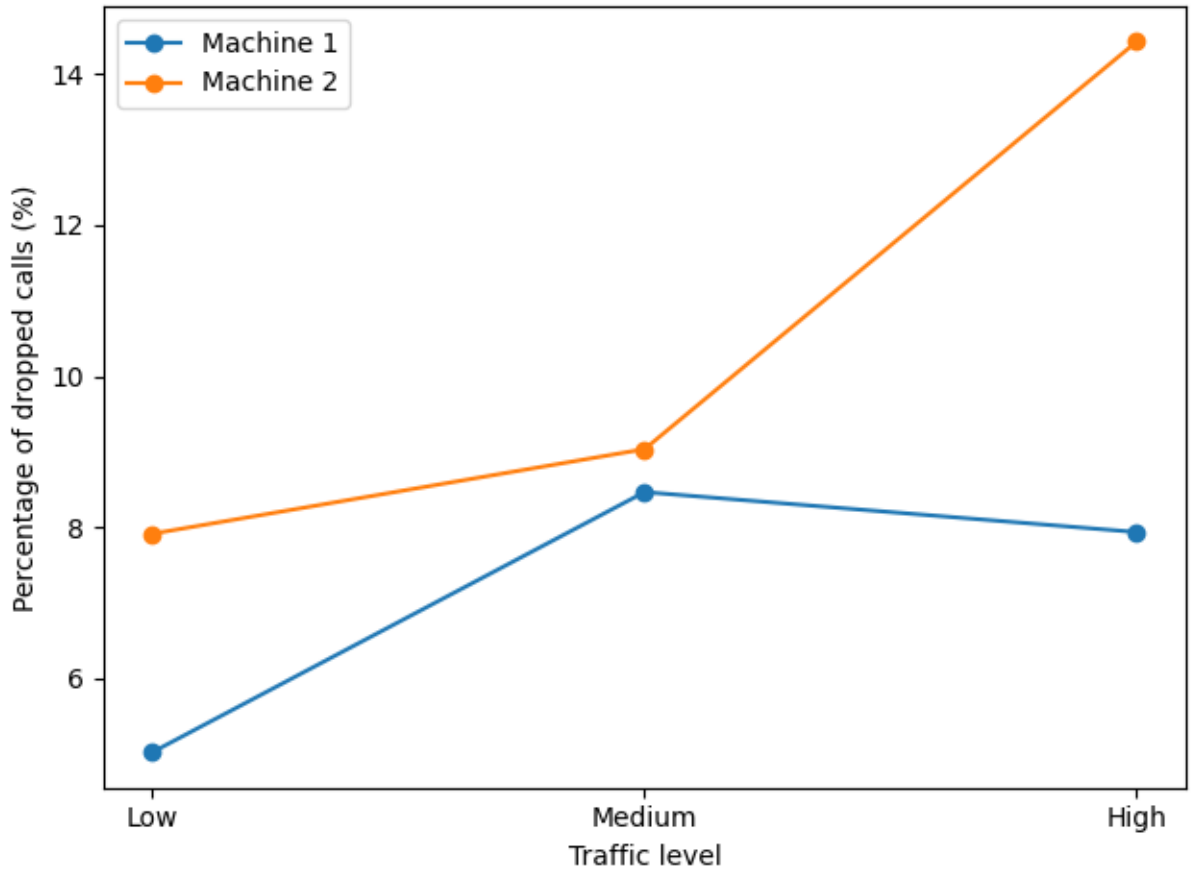

Figure 22. Call drop performance comparison under varying traffic conditions

Figure 22 represents the percentage of call drops that occur across various traffic levels for the Sionna simulation that was created using the Hyperledger Fabric blockchain running on two computers with differing computational capabilities. The graph clearly shows that there are some very distinct behaviors in both of the computers illustrated in the graph, and these behaviors illustrate how the differences in the computing power of the two computers affect their reliability in completing the call in this simulation. As shown in the graph, machine one had a somewhat consistent trend line in that its rate of dropped calls was constant through the various traffic levels, starting at around 5% at the low-traffic level, increasing to about 8% in mid-traffic level, and then again decreasing to the 5% level in the high-traffic levels, finally leveling off at 8% in the high-traffic levels. This trend line of machine one clearly shows that because of the better computational capabilities of machine one, the machine remained a consistent performer throughout the entire test period, with only a small decrease in performance as the traffic levels increased. On the other hand, machine two had a trend line that showed much higher rates of dropped calls than machine one and a much greater rate of increase in the rates of dropped calls as the traffic levels increased, showing a trend line that started at approximately 8% at the low-traffic levels, and rapidly escalated to approximately 14.5% in the high-traffic levels. This steep rise in machine two's trend line clearly indicates that the less powerful

computational capabilities of machine two caused a severe bottleneck in the transaction processing of the Hyperledger Fabric blockchain, and that the severity of this bottleneck became increasingly more severe as the traffic levels increased. The gap between the two machines also increases as the traffic levels increase, with the gap growing from approximately 3% at the low-traffic levels to over 6% at the high-traffic levels, indicating the critical role that adequate computing resources play in the operation of a blockchain system. Also, the dramatic increase in dropped calls at the high-traffic levels in machine two is due to the extreme resource intensity of the operational procedures of the Hyperledger Fabric blockchain and the distributed consensus protocols used to process the transactions in the blockchain, requiring that the transactions be processed within time limits so that transaction timeouts do not occur. Thus, when the computers' resources are constrained, the amount of time needed to process the transactions exceeds the acceptable latency threshold, resulting in calls being dropped before the successful verification of the blockchain. On the other hand, machine one demonstrated greater resilience to increased traffic, with the drop-out percentages remaining relatively contained under high loads, illustrating that sufficient computational capability may provide a buffer against the declines in performance caused by increased traffic. Therefore, these results demonstrate that the computational capability of a Hyperledger Fabric deployment is a major factor in determining the reliability of the deployment, and that less capable systems will experience disproportionately higher drop-out percentages that will also degrade more rapidly as traffic increases, and therefore, organizations desiring to implement this blockchain framework for real-time services should ensure that they have a substantial hardware infrastructure in place to support acceptable service quality.

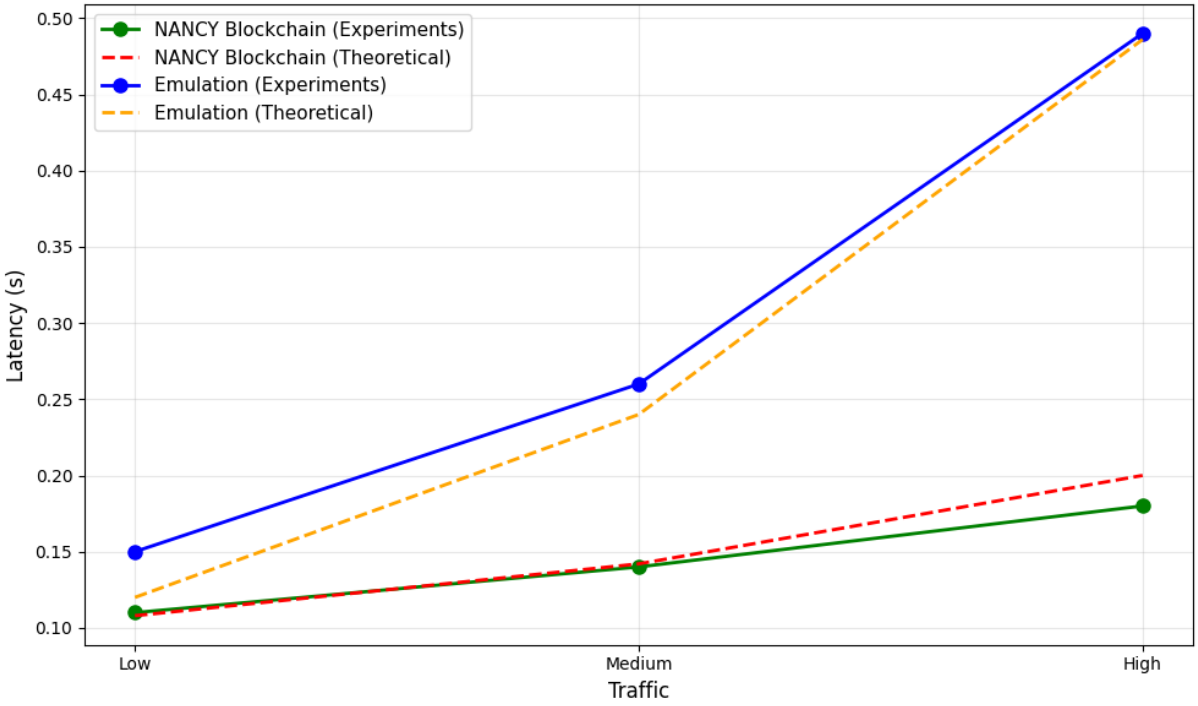

Figure 23. Latency Performance Across Blockchain Implementations Under Varying Traffic Intensity

Figure 23 illustrates the relationship between network latency and traffic intensity across four scenarios comparing experimental measurements with theoretical predictions for two blockchain implementations: the NANCY blockchain, made by NEC, and an emulated system (Fabric). Data from the experimental NANCY blockchain (in green), along with the experimental emulation (in blue), are shown side by side with the theoretical models of the NANCY Blockchain (the red dashed line) and Emulation (the orange dashed line). As shown, both experimental and theoretical models show an

increase in latency as traffic increases in level from Low to Medium to High. This is consistent with how networks behave when they are subjected to increasingly higher loads. It can be seen that both experimental and theoretical data sets track one another very closely. Specifically, it can be observed that the theoretical model of the NANCY Blockchain tracks the experimental NANCY Blockchain data very closely across all traffic levels (beginning at 0.11 seconds under low traffic and ending at 0.18-0.20 seconds under heavy traffic conditions). In addition, the theoretical model of the emulation also tracks the experimental emulation data set very closely (beginning at about 0.15 seconds and ending at approximately 0.45-0.49 seconds under heavy traffic conditions). The emulated system exhibited a much greater increase in latency over the NANCY Blockchain system and the difference increased dramatically as traffic intensified, indicating a decrease in scalability of the emulated system. Most importantly, this graph shows that the theoretical blockchain model developed for predicting performance characteristics of blockchains is capable of modeling both of these blockchain systems and therefore, will allow for the accurate forecasting of the performance of blockchains in various environments.

5. End-to-End performance assessment

This section provides the definition of the end-to-end system model, accompanied by detailed performance assessment results that can serve as design guidelines for future B-RAN systems.

5.1. End-to-end System Model

In this section, we present the end-to-end evaluation of the complete system, assessing how the integrated Sionna simulation, AI module, and blockchain module shape overall performance. We examine the resulting latency, delays, and connection effectiveness between the UE and BS, highlighting the contribution of each module to the system's final behaviour.

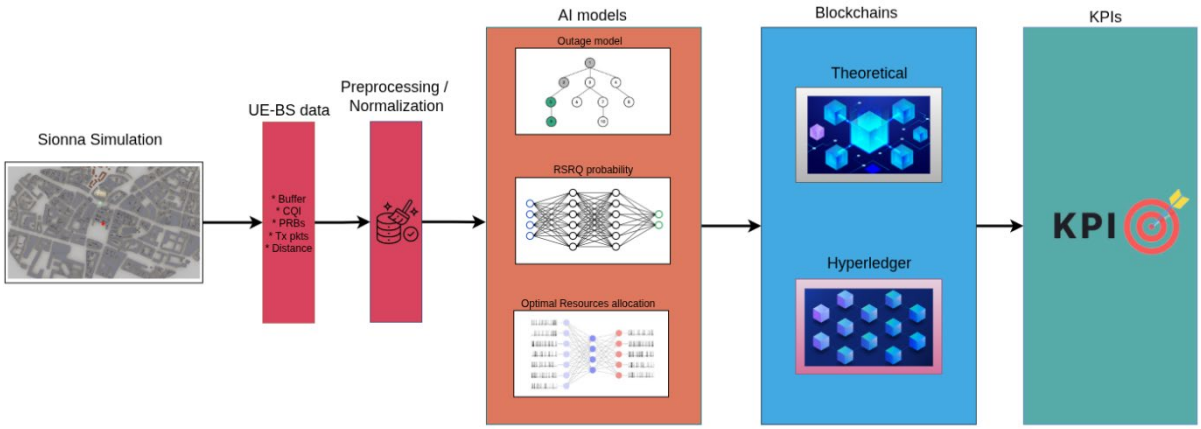

Figure 24. End-to-end system model

The End-To-End Model shown in Figure 24 begins with the first Sionna simulation, which produces the metrics that are needed to measure the connection between UE and BS, like CQI, RSS, and many other indicators used to evaluate the state of the link between UE and BS. The output of the previous step is processed in order to be normalized to obtain the inputs for the AI modules, whose task is to perform the right forecast about the reallocation of the call that can occur in order to maintain the continuity of the connection. Next, we have the blockchain that validates and confirms the transition from one BS to another. Finally, we have the KPIs generated by this simulation.

5.2. Performance Assessment

This section presents a comprehensive evaluation of the integrated system through multiple scenarios that combine Sionna-based UE-BS simulation, AI prediction models, and blockchain recording mechanisms. Each scenario represents a complete pipeline from network simulation through feature extraction, AI-driven prediction (where applicable), blockchain recording, and performance metric derivation. The scenarios are designed to systematically explore different combinations of AI models and blockchain implementations, enabling comparative analysis of prediction accuracy, blockchain performance, and overall system effectiveness.

5.2.1. Temporal dynamics

This section focuses on the temporal dynamics that are in the evaluation of the end-to-end system that strive to avoid dropped calls by executing timely handovers using AI-predictive assistance. In the following analysis, we define two critical operational thresholds. The orange line threshold serves as the trigger for handover procedures in the absence of intelligent network management, while the red call drop threshold represents the absolute lower bound for service continuity. When AI models are

integrated, the star marker indicates the precise timestep where the model is called. Its purpose is to forecast the future RSS trajectory, represented by the green prediction line. This illustrates a shift from reactive to proactive network control. Finally, the black arrow defines the finite time window between the “un”-intelligent handover start threshold and the drop call threshold, during which a successful handover must be completed.

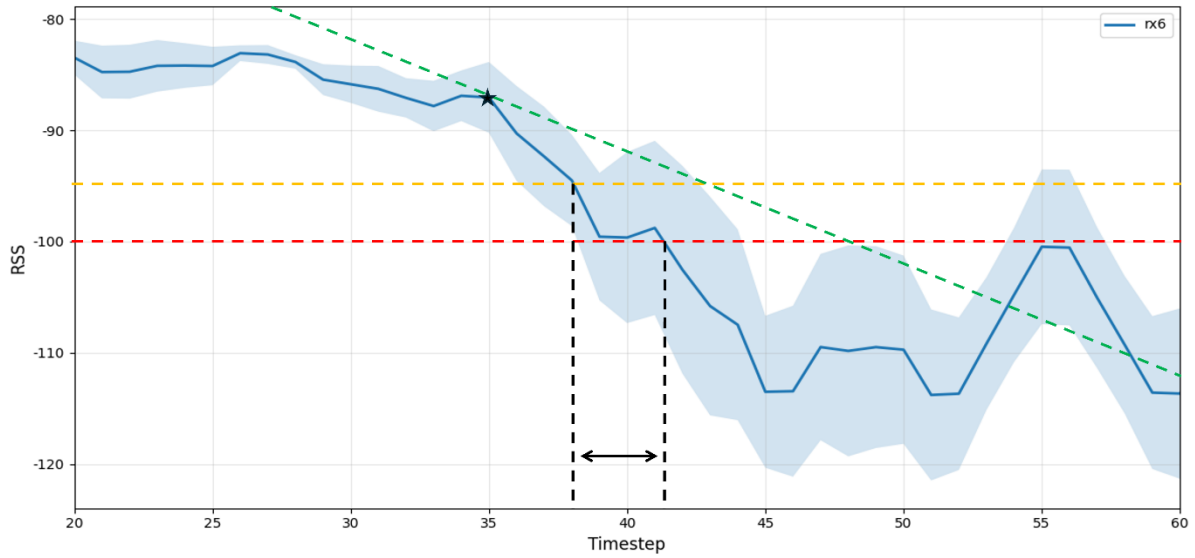

Figure 25. Timeseries of RSS for an individual user

Figure 25 presents a clean and proactive handover case, which shows a classic, monotonic decline in signal strength. The RSS profile is a typical predictable model (e.g., increasing distance or shadowing). The black arrow shows a quite big handover window. It is observed that the AI is called early, while the RSS is still above the handover threshold. The green line visualizes the prediction of the AI model. This indicates the model is operating under near-ideal conditions where the underlying channel dynamics are well-learned. The accurate prediction provides the network with enough time to orchestrate a seamless handover before the RSS even approaches the drop threshold.

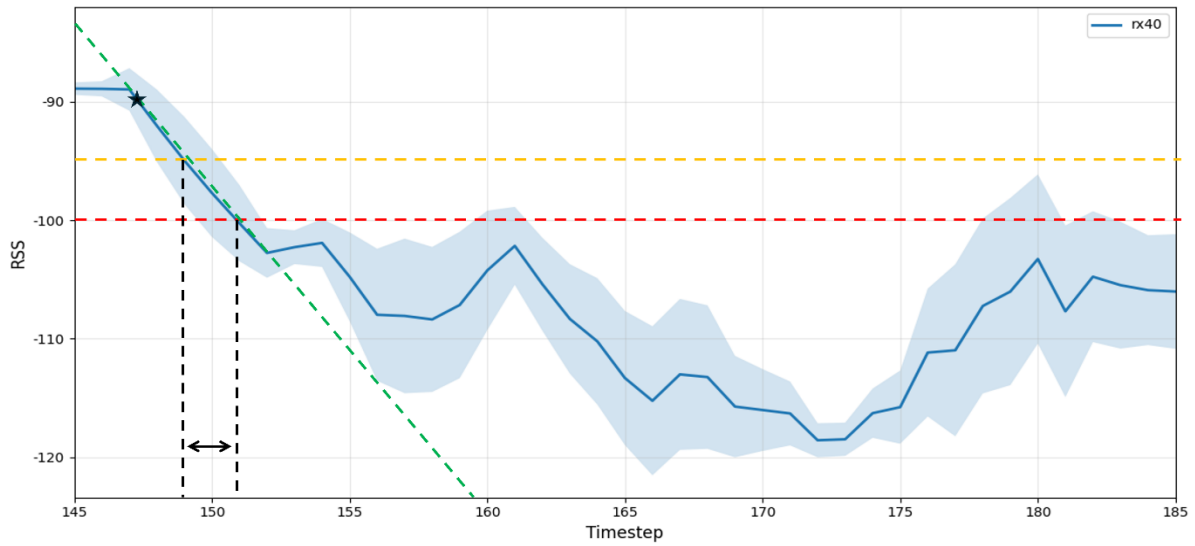

Figure 26. Timeseries of RSS for an individual user

Figure 26 shows a scenario where there is a delayed detection and a critical recovery situation. The RSS profiles show a larger degree of variability and a greater rate of decrease. The RSS has already exceeded the handover threshold a few time steps after the AI model is invoked, and it is possible that the fading or interference is suddenly occurring. The AI Model is invoked when the UE is in the deep fade, near the minimum value of the RSS. The green prediction curve predicts the rapid increase of the RSS, and the positive divergence from the current negative trend indicates an optimistic prediction. The black arrow indicates a very short, urgent time period. The combination of late detection and an optimistic prediction creates a high-risk situation for prediction during the highly non-stationary channel conditions.

5.2.1. AI impact

In this subsection, we investigate the effects of the AI Module on the End-To-End model. We study how the simulation parameters affect the performance of the AI model and the effect of the decisions made by the AI module on the overall behavior of the system. By studying these aspects, we show the contributions of the AI module to improve the stability, the resource management, and the efficiency of the simulation.

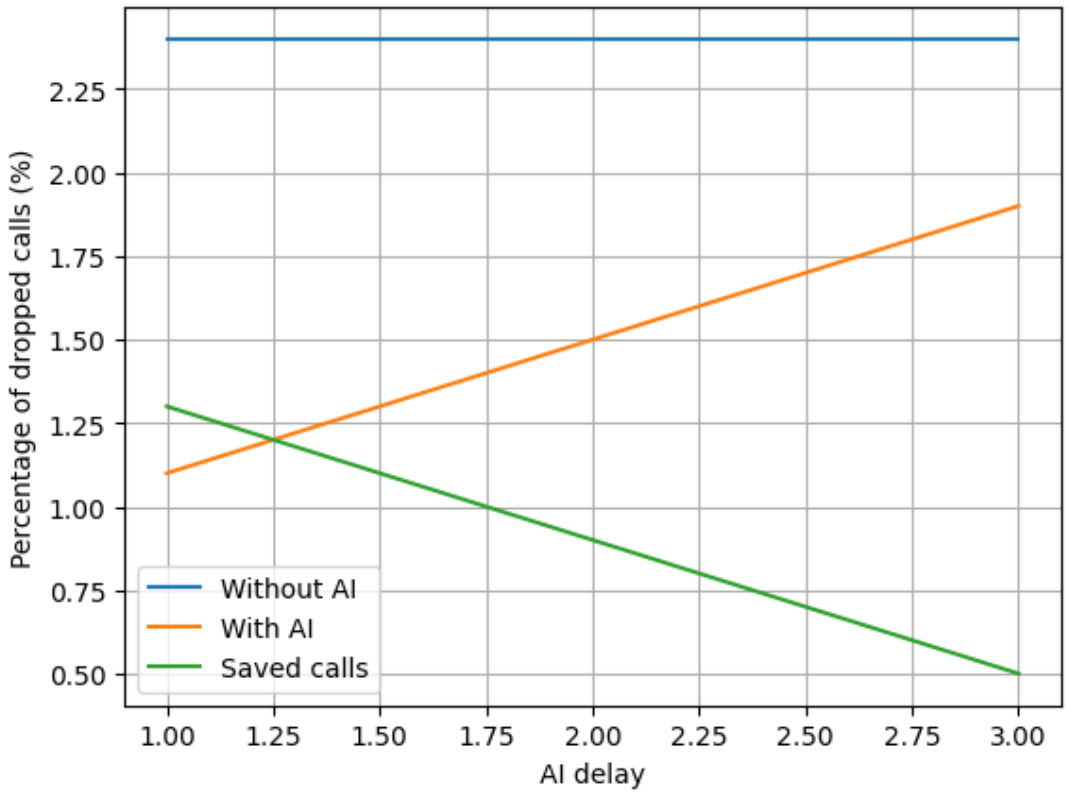

Figure 27. Call drop performance and saved calls as a function of AI processing delay

Figure 27 depicts the relationship between the percentage of calls dropped and the AI delay, where the AI delay is the number of time steps in the Sionna Simulation needed for the AI Model to complete the calculation. For example, an AI delay of 2 indicates that the AI model needs 2 time steps in Sionna simulation to finish processing. The plot displays 3 cases: Baseline Case without AI, Performance case with AI and Number of calls recovered using AI. The results show an inverse relationship between AI delay and the number of recovered calls, i.e., when the AI delay is increased, the number of calls recovered decreases and the percentage of calls recovered by the AI is increased. This happens

because when the AI delay is longer, it means that the model takes more time to calculate and therefore calls that could have been recovered lose the connection during the additional time that the model has to process. The reason for this is that the AI model provides forecasting functions that predict whether a UE-BS connection is likely to fail in the immediate future, thus allowing proactive relocation of the UE to another BS before the connection is lost. Therefore, the faster the AI model calculates, the more calls are recovered because the predictions are provided quickly enough for a successful relocation to take place before the call is actually dropped, illustrating a critical trade-off between the time spent calculating the AI model and the effectiveness of the operation in timely telecommunications contexts.

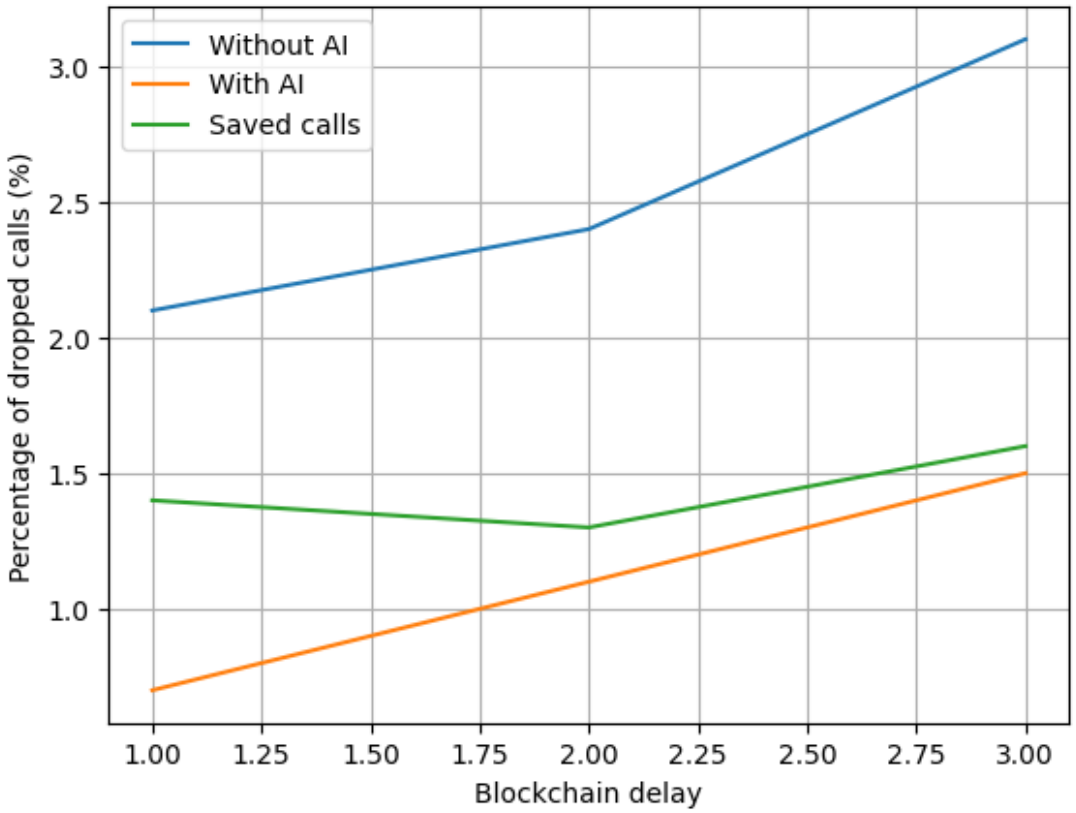

Figure 28. Effect of Blockchain delay on dropped call rates with and without AI integration

Figure 28 shows the relationship between the percentage of dropped calls and the blockchain delay, where the blockchain delay represents the number of Sionna simulation time steps required for the blockchain to validate incoming reallocation requests from a UE to a BS. The plot includes three scenarios: the baseline with blockchain but without AI, the integrated blockchain–AI scenario, and the number of calls saved through the combined blockchain validation and AI prediction. As expected, the scenario with the blockchain and no AI module, shows the highest percentage of dropped calls across all delay values, and in both blockchain scenarios the dropped call rate increases as validation latency grows, confirming the significant impact of blockchain delay. Nevertheless, the inclusion of the AI model consistently reduces dropped calls compared to the blockchain-only case. Furthermore, when considered alongside the results from Figure 28, we can conclude that the AI module plays the most influential role in improving system reliability, followed by the blockchain, and that even under substantial blockchain delay, the presence of AI can still prevent a meaningful portion of otherwise dropped calls.

5.2.2. Blockchain impact

In this subsection, we analyse and present the role and impact of the blockchain module in the end-to-end simulation. We examine how this module validates and secures reallocations between UEs and BSs, as well as how different simulation variables influence the system's performance. Through this analysis, we show how the blockchain component contributes to system reliability and the overall behaviour of the end-to-end system model.

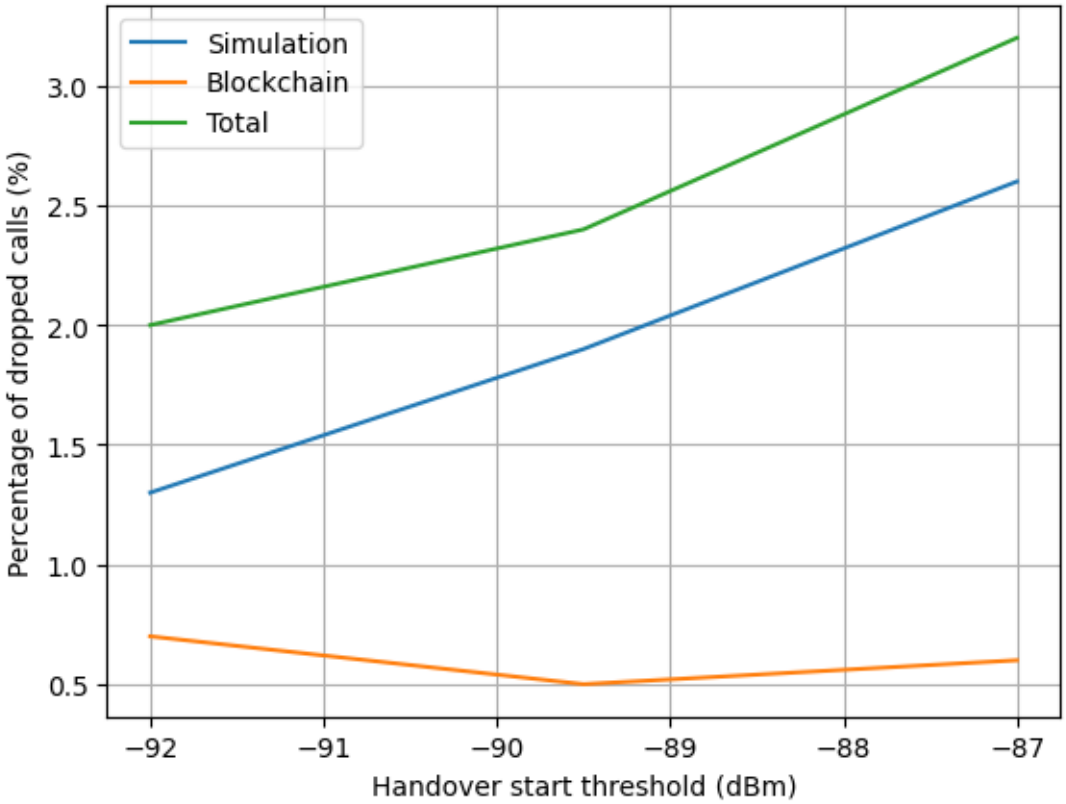

Figure 29. Impact of handover start threshold on system performance with and without Blockchain

The relationship between the starting handover threshold (in dBm) and the percentage of dropped calls is illustrated in Figure 29. The starting handover threshold is defined as the quality of the signal at which a call is declared to be lost. The blue line refers to the Sionna simulation alone, without any additional modules (blockchain and/or AI). The blue line serves as a reference to evaluate the contribution of other elements to the basic simulation. Moreover, the orange line represents the dropped calls due to the Blockchain overhead. The results clearly show that as the starting handover threshold increases (i.e., the lower the signal quality tolerated), the percentage of dropped calls increases in the same way for each of the 3 scenarios. It is interesting to note that the dropped calls due to the blockchain are very stable across all values of the handover starting threshold (between 0.5% and 0.7%), and therefore indicate that the handover starting threshold does not significantly affect the dropped calls due to the blockchain. On the contrary, the baseline shows a significant sensitivity to the variations of the handover starting threshold (from approximately 1.3% at -92 dBm to approximately 2.6% at -87 dBm). As a consequence, the scenario including both the blockchain and the AI (green line) follows the baseline simulation (blue line) instead of the blockchain (orange line) and indicates that the overall behavior of the dropped calls is mainly determined by the sensitivity of the simulations to the threshold, while the blockchain contributes a small and relatively constant overhead that is independent of the timing of the handover start.

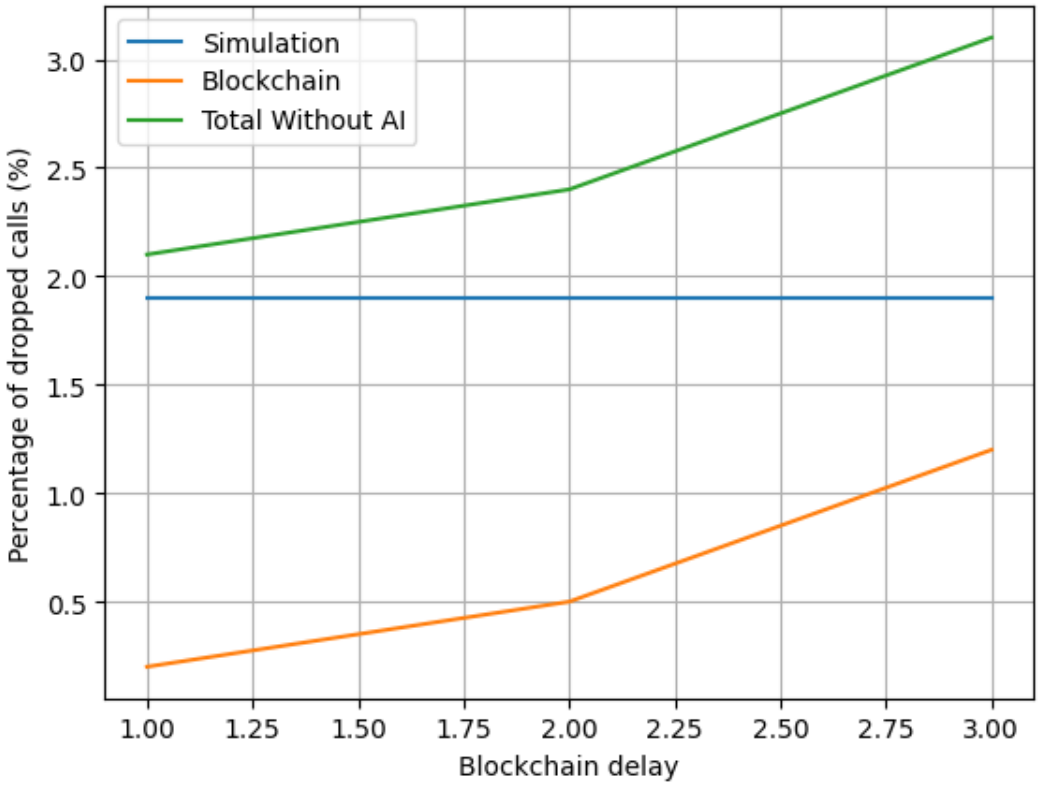

Figure 30. Impact of Blockchain delay on call drops

Figure 30 presents the percentage of dropped calls as a function of the blockchain delay for three different scenarios. In the first case, no blockchain or AI were used; the Sionna simulation was run (the blue line), and was utilized to determine how much better or worse the two other cases performed. In the second case, we included the percent of dropped calls caused solely by the use of the blockchain (orange line); in the third case, we utilized the same percent of dropped calls produced by the Sionna simulation, plus the additional percent of dropped calls generated by the blockchain validation process (green line). Consistent with our expectations, we observed that the scenario that did not utilize a blockchain, showed no effect due to the blockchain delay. We found that utilizing a blockchain produced a large increase in the percentage of dropped calls. The combined case (green line) exhibited the largest percentage of dropped calls, as it consisted of the percentage of dropped calls produced by the Sionna simulation, plus the additional percentage of dropped calls generated by the blockchain validation process. Of interest is that, while the percent of dropped calls increased at a relatively low rate until reaching a threshold of about 2 delay steps (from the Sionna simulation time frame), once past this threshold, the rate of increase became extremely large. Thus, the delay associated with the blockchain impacts the network significantly at low delay times, but significantly more so as the delay grows.

5.2.3. Handover thresholds impact

In this subsection, we analyze how the chosen thresholds for dropped calls influence the overall simulation behavior. We examine both the minimum and maximum values used to determine when a call is considered lost, highlighting how these limits affect system performance, reliability, and the resulting end-to-end outcomes.

Figure 31 demonstrates the relationship between the percent of dropped calls and the call drop threshold, which is the minimum acceptable signal strength, below which a call will be terminated. In

the graph, we demonstrate three scenarios: the baseline case without AI, the performance with AI, and the number of calls that were saved using AI. The results demonstrate that as the call drop threshold increases, the percentage of dropped calls rises across all scenarios, indicating that higher thresholds (corresponding to stricter signal quality requirements) lead to more call failures. Also, the difference between the baseline case without AI and the case with AI continues to grow as the threshold increases, indicating that the AI's ability to make predictions regarding call termination and to take action to avoid termination becomes more important at higher thresholds. Additionally, the number of calls saved by the AI continues to grow as the threshold increases, indicating that the AI is saving more calls that would have been lost at higher thresholds, and that the AI is performing its intended functions, i.e., making predictions regarding calls that are likely to fail, and taking action to relocate calls before they are terminated.

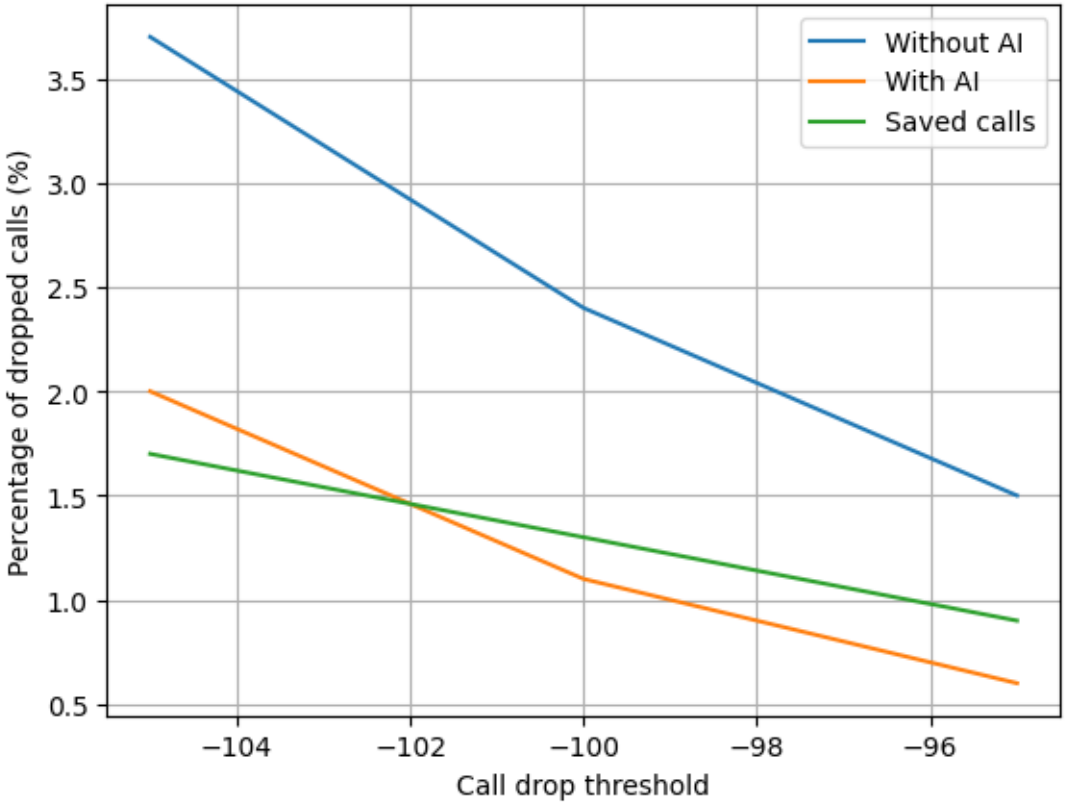

Figure 31. Impact of call drop threshold on system performance with and without AI integration

Figure 32 illustrates the percent of dropped calls as a function of the call drop threshold, measured in dBm. Three cases are illustrated: the baseline simulation (blue line), the dropped calls attributed to the processing delays of the blockchain (orange line), and the total percent of dropped calls, including both the baseline simulation and the blockchain processing delays (green line). As the threshold relaxes (i.e., moving from -105 dBm toward -95 dBm), the percentage of dropped calls decreases for all three cases, consistent with the expected behavior, as relaxing the signal strength requirement allows more calls to remain connected as the signal degrades. It should be noted that the percentage of dropped calls attributable to the total case (green line) tracks much closer to the baseline simulation (blue line) than to the blockchain component, illustrating that the processing overhead attributable to the blockchain represents a relatively minor contributor to overall call failures. Additionally, the distance between the total and baseline cases decreases as the threshold increases, such that the percent of dropped calls due to blockchain decreases from approximately 1% at the most restrictive threshold to approximately 0.1% at the least restrictive threshold. This pattern illustrates that the processing delays

attributable to the blockchain represent a small but measurable contributor to network performance degradation, and that this degradation diminishes as the signal strength requirements become less stringent, and thus the timing margins become wider.

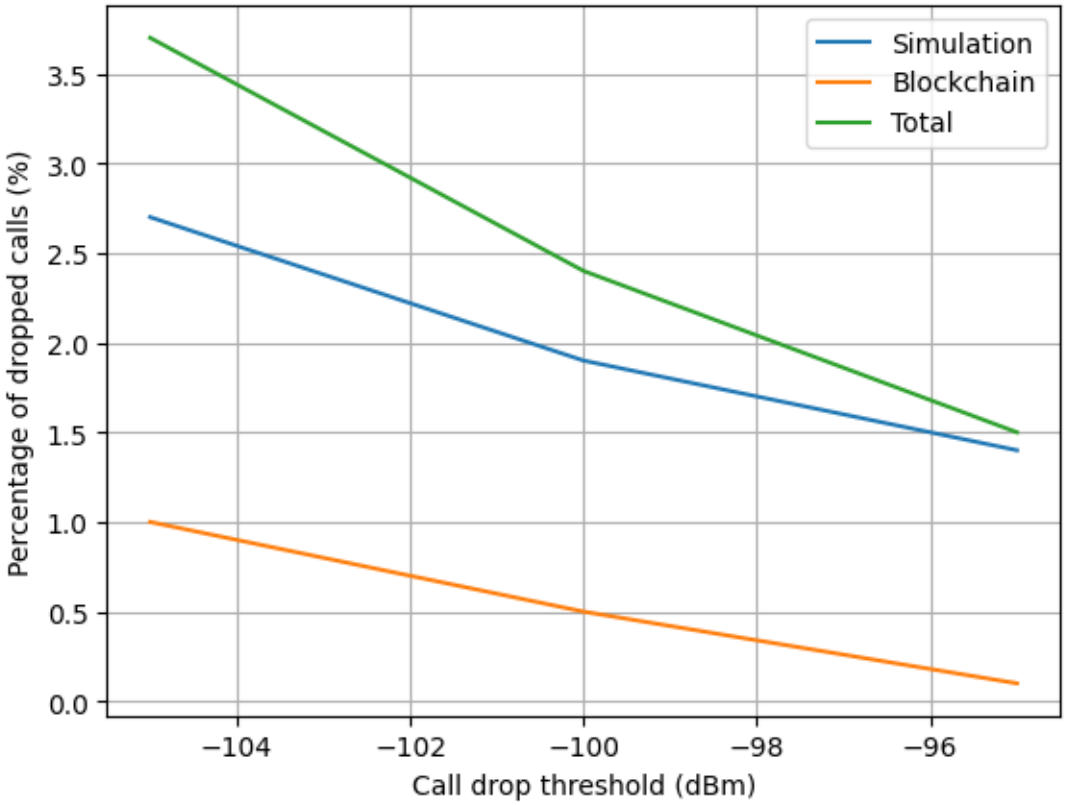

Figure 32. Impact of call drop threshold on system performance with and without Blockchain authorization

5.2.4. Experimental results

Figure 33 demonstrates how the average latency behaves over time (timestamps). The average latency in this figure is the time from the user clicking play until the video starts. There are 3 distinct scenarios that are presented for comparison; the blockchain-integrated simulation (blue line) depicting latency when blockchain-based validation mechanisms are actively employed, the baseline simulation without blockchain (orange line) representing system performance under traditional validation approaches, and the latency difference (red line) calculated as the delta between the blockchain-integrated and baseline scenarios to isolate the performance impact attributable to blockchain integration. As can be seen, the results reveal significant temporal fluctuations in latency across all scenarios, with the blockchain-integrated system consistently exhibiting elevated latency compared to the baseline, particularly pronounced during periods of intensified transactional activity occurring roughly between 14:32 and 17:40. The difference metric (red line) quantifies this overhead, reaching a maximum of approximately 77.12 ms near 15:00:10, which represents the peak latency penalty introduced by blockchain operations under the tested conditions and network load. It is worth noting that, the baseline simulation maintains relatively stable latency throughout the observation period, suggesting that the observed variations in the blockchain scenario stem primarily from the computational and consensus overhead inherent to distributed ledger operations rather than underlying network or simulation dynamics. All experimental results were generated by OTE, which provided a controlled environment for evaluating blockchain integration impacts on telecommunications network performance.

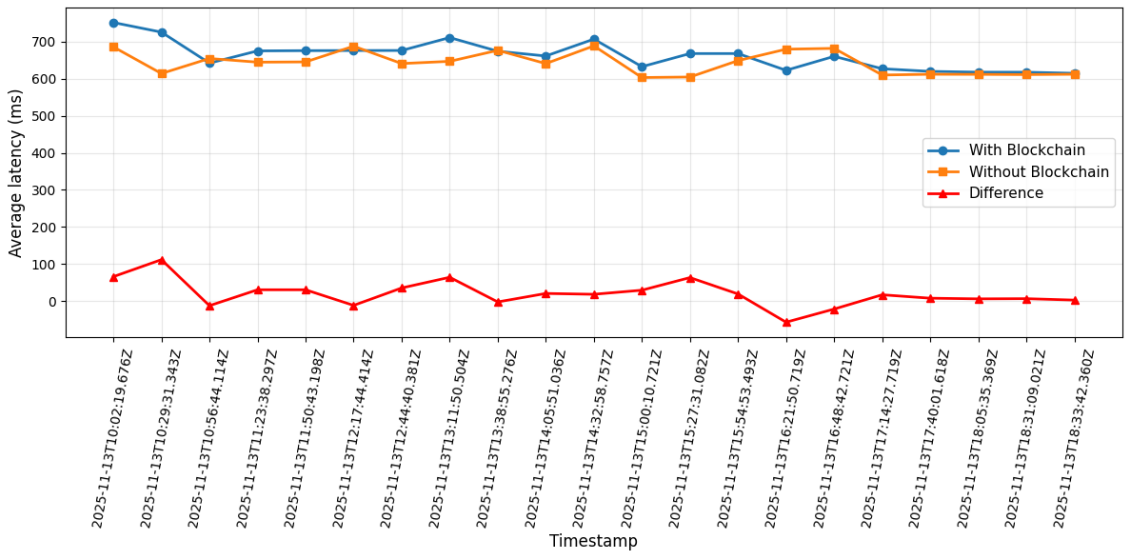

Figure 33. Experimental results by OTE

6. Conclusions

This deliverable assessed the performance achieved by the NANCY AI and blockchain models through simulations and compared the results with experimental data. A high-fidelity validation platform for next-generation wireless network management systems was developed using ray-tracing simulations to test future wireless network management systems. Using NVIDIA Sionna to simulate electromagnetic waves propagating in a real-world environment (with multi-path effects, reflections, diffraction, and RIS), close-to-real-world channel variability and interference dynamics were modeled.

The evaluated intelligent models demonstrated significant predictive capabilities. The outage probability prediction model consistently identified connections at risk of being disrupted and enabled resource allocation in advance of the possibility of call drops. The coverage probability prediction model offered spatial awareness of the current coverage conditions of the network. In terms of base-station (BS) to user-equipment (UE) association, the SNN-based models demonstrated high accuracy in solving the complex network-wide optimization problem of matching UE to BS. A critical distinction was found between the use of bottom-up and top-down SNN methods; these two methods illustrated a fundamental trade-off between the optimization of individual vs. network-wide performance. While the bottom-up SNN method demonstrated near-perfect accuracy, it lacked awareness of the global constraints of the network. The top-down SNN method traded off on the accuracy of individual predictions to ensure that network-wide performance met constraints.

In terms of integrating AI models into the end-to-end workflow of the network, the analysis demonstrated that the timely application of AI predictions could prevent a significant number of possible call drops by allowing the proactive relocation of users to a BS that would provide better service as channel conditions deteriorate. Nevertheless, this benefit is directly related to the delay introduced by the process of AI inference; thus, each additional timestep used for inference limits the number of possible call drops prevented. This indicates that AI-based optimization of wireless networks is time-sensitive. Therefore, computational resources allocated to AI inference must be allocated so that model complexity and prediction accuracy can meet the temporal constraints introduced by the changing behavior of wireless channels due to variations in channel conditions and user mobility.

The evaluation of the blockchain addressed the critical issue of determining whether it is feasible to achieve high performance when integrated into time-constrained processes of wireless network management. Contrary to the perception that blockchain is inherently incompatible with demanding real-time systems, the results of this deliverable illustrate that properly designed and configured blockchain systems can operate within acceptable latency limits of network access. Evaluation results demonstrated a threshold in mining power's influence on dropped call rates. Low mining power exhibits significantly increased call drop rates (~9%) under high-traffic, while medium and high mining power configurations converge at much lower call drop rates (<4%). This demonstrates that, with sufficient computational capacity already allocated to blockchain, additional mining power results in diminishing performance improvements. This suggests that B-RAN requires sufficient computational capacity to match expected network management event rates. Moreover, the saturation effect establishes a window of efficient operation.

The analysis presented in this deliverable highlighted specific system tradeoffs that need to be carefully taken into consideration in order to design B-RAN systems that can meet the necessary requirements.

- Inference delay vs. Prediction accuracy: The trade-off between the processing time of AI algorithms versus prediction accuracy is a primary constraint to the degree of complexity that

may be achieved with AI models. While more complex AI models provide better predictions and are thus desirable, they also require more time to execute compared to less complex models. One of the most critical examples of this is the use of AI to determine the likelihood of a call drop occurring based on the location of a UE and the available BSs. If the AI model requires excessive time to complete its estimation, it may not be able to intervene in a timely manner to successfully complete the handover. Therefore, the B-RAN system designer will have to strike a balance between the need for an increasingly accurate model and the time required to execute it.

- Computational resources vs. Performance: The performance of both the AI and blockchain components is bound to the capabilities of the available hardware. The number of computational resources required for each component to operate at an acceptable level of performance introduces specific high thresholds, beyond which additional amounts of computational resources will have little to no effect on the performance of the module. The implication of this is that the B-RAN system designer must accurately select the number of computational resources assigned to AI and blockchain components in order to match the anticipated workload of the system.
- Coverage expansion vs. Complexity: The simulations demonstrated that advanced physical-layer technologies such as RIS may be successfully integrated into a B-RAN system. RIS may enable extension of coverage to areas of the network where a UE is physically located behind a barrier preventing them from receiving a direct signal from the closest BS. However, for a RIS installation to operate correctly, it must be included as part of the set of variables being optimized for in the algorithm managing the network. The trade-off between extending coverage and increasing the complexity of the system must be evaluated thoroughly for each potential deployment of a B-RAN system.

Bibliography

[1] J. Hoydis, S. Cammerer, F. A. Aoudia, A. Vem, N. Binder, G. Marcus, and A. Keller, "Sionna: An open-source library for next-generation physical layer research." arXiv preprint arXiv:2203.11854 ,2022.

[2] F. A. Aoudia, J. Hoydis, M. Nimier-David, B. Nicolet, S. Cammerer, and A. Keller, "Sionna RT: Technical Report." arXiv preprint arXiv:2504.21719, 2025.

[3] N. Vaara, P. Sangi, M. B. López, and J. Heikkilä, "Differentiable High-Performance Ray Tracing-Based Simulation of Radio Propagation with Point Clouds." arXiv preprint arXiv:2507.04021, 2025.

[4] C. Pan, G. Zhou, K. Zhi, S. Hong, T. Wu, Y. Pan, H. Ren, M. D. Renzo, A. Lee Swindlehurst, R. Zhang, and A. Y. Zhang, "An Overview of Signal Processing Techniques for RIS/IRS-Aided Wireless Systems," IEEE Journal of Selected Topics in Signal Processing, vol. 16, no. 5, pp. 883-917, Aug. 2022.

[5] Z. Zhang, L. Dai, X. Chen, C. Liu, F. Yang, R. Schober, and H. V. Poor, "Active RIS vs. Passive RIS: Which Will Prevail in 6G?," IEEE Transactions on Communications, vol. 71, no. 3, pp. 1707-1725, March 2023

[6] S. Zeng, H. Zhang, B. Di, Z. Han, and L. Song, "Reconfigurable Intelligent Surface (RIS) Assisted Wireless Coverage Extension: RIS Orientation and Location Optimization," IEEE Communications Letters, vol. 25, no. 1, pp. 269-273, Jan. 2021.

[7] J. Hoydis, F. A. Aoudia, S. Cammerer, M. Nimier-David, N. Binder, G. Marcus, and A. Keller, "Sionna RT: Differentiable Ray Tracing for Radio Propagation Modeling," 2023 IEEE Globecom Workshops (GC Wkshps), 2023, pp. 317-321.

[8] M. Polese, L. Bonati, S. D'Oro, S. Basagni, and T. Melodia, "ColO-RAN: Developing Machine Learning-based xApps for Open RAN Closed-loop Control on Programmable Experimental Platforms," IEEE Transactions on Mobile Computing, pp. 1-14, July 2022.

[9] R. Asensio-Garriga, A. J. Pogo Medina, G. Alarcon-Hellin, L. Bernal Escobedo, R. Sanchez-Iborraand A. Skarmeta Gómez, "University of Murcia 5G Dataset 1". Zenodo, May 26, 2025. doi: 10.5281/zenodo.15516876.

[10] Kouvakis, V., Trevlakis, S.E., Arapakis, I. and Boulogeorgos, A.A.A., 2025. Spiking Neural Networks for Resource Allocation in UAV-Enabled Wireless Networks. arXiv preprint arXiv:2508.03279.

[11] E. Koutsonas, S. E. Trevlakis, and A.-A. A. Boulogeorgos, "A deep learning mobility management approach for RIS-empowered millimeter wave wireless systems," 2023 IEEE Virtual Conference on Communications (VCC), 2023, pp. 177–182.

[12] G. Araniti, A. Iera, S. Pizzi, and F. Rinaldi, "Toward 6G Non-Terrestrial Networks," in IEEE Network, vol. 36, no. 1, pp. 113-120, January/February 2022.

[13] B. Agarwal, M. A. Togou, M. Marco and G. -M. Muntean, "A Comprehensive Survey on Radio Resource Management in 5G HetNets: Current Solutions, Future Trends and Open Issues," in IEEE Communications Surveys & Tutorials, vol. 24, no. 4, pp. 2495-2534, 2022.

[14] V. Ponnusamy and A. Vasuki, "AI-Enabled Intelligent Resource Management in 6G," in AI and Blockchain Technology in 6G Wireless Network, M. D. Borah, P. Singh, G. C. Deka, Eds. Springer, Singapore, 2022, pp. 71–92.

[15] S. E. Trevlakis, M. Belesioti, H. Koumaras, A. A. Boulogeorgos, I. Chochliouros, and T. A. Tsiftsis, "An innovative architectural blueprint towards sustainable 6G systems", IEEE 29th International Workshop on Computer Aided Modeling and Design of Communication Links and Networks (CAMAD), 2024, pp. 1–8.

[16] X. Lu, J. Yuan, C. Chen, X. Chen, C. Wu, and R. Yin, “Energy-efficient user association and resource allocation for decentralized mutual learning,” IEEE Global Communications Conference (GLOBECOM), 2022, pp. 867–872.

[17] B. Agarwal, M. A. Togou, M. Ruffini, and G.-M. Muntean, “A low com- plexity ml-assisted multi-knapsack-based approach for user association and resource allocation in 5G HetNets,” IEEE International Symposium on BMSB, 2023, pp. 1-6.

[18] D. Liu, Y. Chen, K. K. Chai, T. Zhang, and C. Pan, “Adaptive user association in hetnets with renewable energy powered base stations,” in 2014 21st International Conference on Telecommunications (ICT), 2014, pp. 93–97.

[19] H. Lin, C. Zhang, Y. Huang, R. Zhao, and L. Yang, “User association for cache enabled millimeter wave ultra dense downlink networks,” 2019 IEEE/CIC International Conference on Communications in China (ICCC), 2019, pp. 432–437.

[20] Ebrahim and E. Alsusa, “Adaptive decoupling and multi-BS association in heterogeneous networks,” IEEE Access, vol. 5, pp. 18 121–18 131, 2017.

[21] V. Kouvakis, S. E. Trevlakis, A.-A. A. Boulogeorgos, H. Liu, W. Khalid, T. A. Tsiftsis, and O. A. Dobre, “Hierarchical blockchain radio access networks: Architecture, modelling, and performance assessment,” IEEE Open Journal of the Communications Society, vol. 6, pp. 576-592, 2025.

[22] A.-A. A. Boulogeorgos, S. E. Trevlakis, and T. A. Tsiftsis, “Holographic ris empowered thz communications with hardware imperfections under adverse weather conditions,” IEEE Transactions on Communications, vol. 73, no. 1, pp. 662–676, 2025.

[23] 3GPP, “Study on channel model for frequencies from 0.5 to 100 ghz,” 3rd Generation Partnership Project (3GPP), Technical Report TR 38.901, March 2017.

[24] J. Hoydis, S. Cammerer, F. Ait Aoudia, M. Nimier-David, L. Maggi, G. Marcus, A. Vem, and A. Keller, “Sionna,” 2022. [Online]. Available: https://nvlabs.github.io/sionna/

[25] I. Crysovergis, S. E. Trevlakis, D. Kleitsas, A. -A. A. Boulogeorgos, T. A. Tsiftsis, and D. Niyato, "A Digital Twin Based Reconfigurable Intelligent Surface Phase Adaptation Using Spiking Reinforcement Learning Policy Optimization," IEEE International Conference on Machine Learning for Communication and Networking (ICMLCN), 2025, pp. 1-7.

[26] Blender Foundation, “Blender,” 2025. [Online]. Available: https://www.blender.org/

[27] OpenStreetMap contributors, “Openstreetmap,” 2025. [Online]. Available: https://www.openstreetmap.org/

[28] CellMapper, “Cellmapper,” 2025. [Online]. Available: https://www.cellmapper.net/

[29] S. Li, Z. Zhang, R. Mao, J. Xiao, L. Chang and J. Zhou, "A Fast and Energy-Efficient SNN Processor With Adaptive Clock/Event-Driven Computation Scheme and Online Learning," IEEE Transactions on Circuits and Systems I: Regular Papers, vol. 68, no. 4, pp. 1543-1552, Apr. 2021.

[30] Y. Wei, F. Yao and Y. Kang, "Design and Application of Spiking Neural Networks Based on LIF Neurons," 3rd International Conference on Data Analytics, Computing and Artificial Intelligence (ICDACAI), 2024, pp. 10-15.

[31] V. Kouvakis, S. E. Trevlakis, A. -A. A. Boulogeorgos, H. Liu, T. Tsiftsis, and O. A. Dobre, “Markov Chain-based Model of Blockchain Radio Access Networks,” IEEE 29th International CAMAD , Oct 2024, pp. 1-6..

[32] M. N. Halgamuge, G. K. Munasinghe and M. Zukerman, "Time Estimation for a New Block Generation in Blockchain-Enabled Internet of Things," IEEE Transactions on Network and Service Management, vol. 21, no. 1, pp. 535-557, Feb. 2024.