ITUJournal
Future and evolving technologies

ITU Journal on Future and Evolving Technologies, Volume 7, Issue 2, June 2026

ARTICLE

ML-driven network orchestrator for 6G O-RAN: Resolving multi-xApp conflicts in near-RT RIC

Can Kurtulan[1,2], Onur Kalinagac[1,2], Erdinc Ozdemir[1,2], Burak Gorkemli[1,2], Gokhan Karakus[1,2], Semih Aktas[3], Mehmet Basaran[3]

[1] Odine, Istanbul, Turkey, [2] OdineLabs, Izmir, Turkey, [3] Turkcell, Istanbul, Turkey

Corresponding author: Can Kurtulan, can.kurtulan@odine.com

The evolution toward 6G networks necessitates a shift toward fully-automated, AI-native, and zero-touch operational solutions to handle unprecedented network complexity. Within the Open RAN (O-RAN) architecture, the near-Real-Time RAN Intelligent Controller (near-RT RIC) plays a pivotal role in hosting third-party xApps for radio resource management. However, the deployment of multiple xApps from different vendors often leads to operational conflicts, such as overlapping control commands and resource competition, which can degrade network performance. This paper addresses this challenge by proposing an ML-driven network orchestrator specifically designed for conflict detection and resolution in a multi-xApp 6G O-RAN environment. Our framework utilizes advanced machine learning techniques to autonomously identify potential steering conflicts and execute resolution strategies in real time. Experimental results demonstrate that the ML-driven orchestrator effectively mitigates performance degradation caused by xApp collisions, maintaining high-fidelity service delivery even under dense traffic conditions. This study contributes to the realization of self-optimizing 6G networks by providing a scalable and resilient orchestration logic for future autonomous RAN operations.

Keywords: 6G O-RAN, ML orchestration, multi-objective utility optimization, near-RT RIC, xApp conflict resolution, zero-touch network automation

1. INTRODUCTION

The transition from fifth-generation (5G) to sixth-generation (6G) networks marks a fundamental paradigm shift from AI-assisted functionalities to AI-native orchestration, where intelligence is an intrinsic component of the Radio Access Network (RAN) [1, 2, 3]. Unlike previous generations that relied on predefined procedures and static resource allocation, 6G is envisioned as an advanced, adaptive, and fully cognitive platform requiring sophisticated orchestration across communication, sensing, and computation layers. Central to this vision is the "AI orchestrator," an entity composed of connected, task-specific algorithms designed to manage the network end to end by dynamically allocating radio resources and adapting configurations in real time [4]. This evolution is supported by an industry-wide shift toward virtualization and cloud-native network functions, establishing the flexibility needed for integrating emerging technologies. Specifically, the Open RAN (O-RAN) perspective emphasizes disaggregating network components and implementing open interfaces to foster a dynamic, programmable RAN environment essential for meeting 6G's stringent requirements [5]. At the heart of the O-RAN architecture lies the near-Real-Time RAN Intelligent Controller (near-RT RIC), a pivotal logical function that hosts third-party xApps to execute Radio Resource Management (RRM) decisions within control loops of 10 ms to 1 s [6, 7]. In this study, we propose an ML Orchestrator (MLO) strategically positioned as an intelligent interceptor layer between xApps and the near-RT RIC core.

The system's monitoring and control capabilities are standardized through O-RAN service models. The E2SM Key Performance Measurement (KPM) service model enables reporting critical metrics, such as throughput and latency, from the gNodeB (gNB) to the RIC [8]. Our MLO leverages these metrics for real time situational awareness. For control execution, the E2SM RAN Control (RC) service model facilitates procedures like handover control and resource allocation [9]. By intercepting control messages, the proposed framework identifies and resolves operational conflicts in multi-vendor xApp deployments, ensuring stable performance.

The disaggregated nature of O-RAN, while promoting vendor neutrality, introduces significant challenges regarding xApp-level conflicts. Theoretical frameworks categorize these interactions into direct conflicts involving parameter collisions, indirect conflicts characterized by Key Performance Indicator (KPI) interference, and implicit conflicts arising from shared resource competition [10, 11]. To date, the literature has predominantly proposed low-latency, rule-based mitigation strategies leveraging Service Level Agreements (SLAs) and Quality of Service (QoS) thresholds [10]. However, the complexity of 6G necessitates advanced, explainable machine learning frameworks capable of evaluating causal relationships between RAN Control Parameters (RCPs) and KPIs. Static solutions that lack fine-grained analysis of RCP impacts remain insufficient for autonomous 6G orchestration [12, 13, 14].

Recent research has explored advanced machine learning architectures to overcome these limitations. For instance, knowledge distillation can synthesize the intelligence of multiple pretrained xApps into a single "student" model, minimizing operational outages during concurrent deployment [13]. Additionally, data-driven methods utilizing Graph Convolutional Networks (GCNs) show promise in predicting conflict types by capturing complex topological interdependencies between network entities [15]. Furthermore, context-aware mitigation frameworks employing Reinforcement Learning (RL), such as the Advantage Actor-Critic (A2C) algorithm, have been proposed to reconcile competing xApp actions based on real time network variables without requiring retraining [16]. Despite these advancements, achieving seamless integration for real time multi-vendor coordination within a unified 6G orchestration layer remains a critical challenge.

To address these hurdles, this paper presents contributions developed within the framework of the 6G-SMART project, which aims to revolutionize mobile network management through large-scale automation. The core of the solution is a unified orchestration platform designed to coordinate multiple ML algorithms for efficient automation. A central contribution is the development of an open, pluggable ML Convergence (MLC) platform that allows third-party developers to deploy AI-based xApps in a harmonized and simplified manner. This architecture specifically targets conflicting decisions that arise from simultaneous model operations. By integrating MLC procedures into existing O-RAN near-RT RIC specifications, the 6G-SMART framework provides a resilient foundation for autonomous, self-organizing 6G networks [17].

A preliminary version of this work has been accepted for presentation at ICT 2026 [18]. The present journal article extends that conference version along several axes. Architecturally, we formalize the orchestrator as an open, pluggable MLC platform that decouples third-party xApps from the near-RT RIC through a northbound RESTful API and a southbound E2 protocol-translation layer, presenting itself as a "Virtual RIC" to applications. The data acquisition campaign is enlarged to 300 independent simulation seeds, totaling 15 000 s of high-fidelity network telemetry. Analytically, we move from a single-event illustration of conflict resolution to an aggregate statistical analysis over 521 detected conflicts, reporting the inter-xApp interaction distribution, the resolution-outcome split between modification and complementary application, the per-KPI improvement profile of the MLO-enabled architecture, and a predictive-fidelity benchmark of the orchestrator's KPI forecasts against ns-3 ground truth. The journal version further contributes a standardized mapping of RAN control parameters to E2SM-RC service styles and action identifiers, an expanded background and related-work treatment of the xApp conflict taxonomy, and a more detailed physical-layer modeling description that includes a three-layer hybrid propagation model, Uniform Planar Array (UPA) antenna configurations, and dynamic beamforming-gain computation. Material that is reused from the conference version, namely the proactive decision algorithm, the multi-objective utility formulation, the XGBoost training pipeline, and the Node-002 micro-event case study, is retained in expanded form and explicitly identified as such throughout the manuscript.

The experimental validation in this paper is conducted at 3.5 GHz (FR1) with a 100 MHz channel, the mid-band configuration that currently anchors the majority of commercial multi-vendor O-RAN deployments and constitutes the spectrum baseline from which early-6G systems are expected to evolve. FR1 is therefore the most operationally representative setting in which to validate an AI-native, multi-vendor orchestration architecture today. The MLO itself operates at the E2SM-RC control-message layer and is decoupled from carrier-frequency-specific assumptions, so the architectural contribution applies equally to upper-mid-band (FR3) and mmWave (FR2) deployments that 6G is expected to incorporate; quantitative validation of the MLO under those bands is identified as a planned extension of this work.

2. BACKGROUND AND RELATED WORK

2.1 O-RAN near-RT RIC and service models

The O-RAN architecture promotes network programmability and openness by centralizing radio resource management through the near-RT RIC [6]. Within this ecosystem, the interaction between third-party applications (xApps) and the underlying RAN is standardized via E2 Service Models (E2SM) over the E2 interface. Two critical service models form the technical foundation of this study:

E2SM Key Performance Measurement (KPM): This model is essential for monitoring the network state by enabling the collection of performance metrics from E2 nodes, such as the gNodeB (gNB). KPM utilizes the RIC INDICATION message structure to provide a continuous stream of telemetry [8]. Specifically, the RIC INDICATION HEADER identifies the reporting source, while the RIC INDICATION MESSAGE carries high-fidelity measurements including throughput, packet delay, and cell load. These data points serve as the primary features for the MLO's proactive decision-making process.

E2SM RAN Control (RC): This model provides the mechanism for the near-RT RIC to execute functional changes on the network [9]. Control commands generated by xApps are transmitted to the E2 node via the RIC CONTROL REQUEST message. This message structure consists of a CONTROL HEADER, which specifies the target parameters, such as transmission power, antenna tilt, or handover thresholds, and a CONTROL MESSAGE that contains the specific action instructions.

The proposed MLO strategically intercepts these standardized RIC INDICATION and RIC CONTROL REQUEST messages. By processing these messages within the MLC platform, the orchestrator evaluates the impact of control commands in conjunction with real time KPM data, identifying and resolving operational conflicts before they are committed to the RIC core.

2.2 Taxonomy of xApp conflicts

The disaggregated architecture of Open RAN allows multiple xApps to operate simultaneously; however, this architectural flexibility introduces significant challenges regarding operational conflicts [10]. According to the O-RAN technical specifications and current literature, these conflicts are classified into three primary categories: direct, indirect, and implicit conflicts.

Direct conflicts: These occur when two or more xApps attempt to modify the same RCP of a specific E2 node at the same time. For instance, if an energy-saving xApp and a Mobility Robustness Optimization (MRO) xApp simultaneously attempt to adjust the transmission power of the same cell, the conflicting commands can lead to network instability or failure of the intended optimization [11].

Indirect conflicts: An indirect conflict arises when multiple xApps modify different RCPs that eventually influence the same KPI. A typical example involves one xApp adjusting antenna tilt while another modifies handover thresholds; although the parameters are distinct, both actions impact the handover success rate and cell coverage, potentially leading to suboptimal performance [10].

Implicit conflicts: These conflicts occur when xApps compete for limited shared resources, such as hardware processing power or Physical Resource Blocks (PRBs), without necessarily targeting the same control parameters. Such resource competition can degrade the performance of all active applications if not properly orchestrated.

Defining this taxonomy is essential for developing robust mitigation strategies, as rule-based solutions often struggle to account for the complex interdependencies and hidden correlations inherent in 6G multi-vendor environments.

2.3 Literature review and research gap

In recent years, research focus has shifted toward advanced Machine Learning (ML) architectures to overcome the inherent limitations of conventional rule-based conflict mitigation frameworks. One prominent approach in the literature is "knowledge distillation," which aims to synthesize the intelligence of multiple pretrained xApps into a single "student" model [13]. While this method reduces the risk of operational outages, it significantly restricts the "plug-and-play" flexibility and modularity promised by O-RAN, as the integrated model requires extensive retraining whenever a new xApp is introduced to the network.

Alternatively, data-driven methods utilizing Graph Convolutional Networks (GCNs), such as the GRAPHICA framework, have demonstrated significant potential in conflict prediction by capturing complex topological relationships and interdependencies between various network entities. However, the high computational complexity associated with GCN models poses a challenge for implementation under the microsecond-level near-Real-Time (near-RT) constraints required by 6G ecosystems [15]. Reinforcement Learning (RL)-based strategies, particularly context-aware schedulers employing the Advantage Actor-Critic (A2C) algorithm, have also been proposed to coordinate xApp actions according to dynamic network states. Nevertheless, RL-based solutions often face convergence issues and prolonged training periods, which may lead to instability in highly dynamic network environments [16].

Furthermore, explainable ML (XAI) and causal inference techniques have been leveraged to analyze the underlying relationships between RCPs and KPIs. Despite their analytical depth, most of these studies remain evaluation-centric and lack a proactive, real time intervention mechanism. The fundamental research gap in existing literature is the absence of a low-latency orchestration layer that maintains modularity through an "interceptor" architecture while proactively maximizing a multi-objective utility function within a narrow 200 ms window. In alignment with the 6G-SMART vision, this study addresses this gap by integrating high-speed XGBoost predictors within a centralized orchestrator to provide real time, scalable, and conflict-free network management.

3. PROPOSED SYSTEM ARCHITECTURE

3.1 MLC platform and interceptor layer

The core of our proposed architecture is the MLC platform, a high-performance middleware designed to decouple third-party applications (xApps) from the underlying Near-Real-Time RAN Intelligent Controller (near-RT RIC), as shown in Fig. 1. Unlike traditional O-RAN deployments where xApps establish direct E2 sessions with the RIC core, our framework introduces a control plane abstraction layer. Within this layer, the MLO serves as an intelligent proxy, presenting itself as a "Virtual RIC" to the xApps.

Architectural decoupling via northbound RESTful API: To ensure vendor neutrality and reduce the computational overhead on the xApp side, the MLC platform exposes a high-throughput northbound RESTful API. xApps are configured to target the MLC's IP address instead of the physical near-RT RIC. This "RIC-agnostic" approach allows xApps to subscribe to network metrics and submit RAN Control (RC) requests using standardized JSON-based REST calls. This architectural shift significantly simplifies the xApp development lifecycle, as developers are no longer required to implement complex ASN.1 encoding or manage raw E2SM (KPM/RC) message stacks.

Proactive interception and protocol translation: The MLO module functions as a sophisticated interceptor that manages the end-to-end signaling flow. When an xApp initiates an E2 setup or a subscription request, the MLO captures the intent via the REST interface. For the southbound communication, the MLO acts as a protocol translator; it encapsulates the validated optimization intents into standardized RIC CONTROL REQUEST messages and transmits them to the southbound near-RT RIC (e.g., FlexRIC) via the E2 interface. This centralized interception ensures that no control command is executed on the RAN without first passing through the MLO's proactive conflict resolution logic.

High-performance infrastructure and feature caching: To meet the stringent 6G latency requirements, the entire MLC platform is deployed on a 128-core high-performance server. To achieve near-zero latency in data preparation, the MLO implements an in-memory feature store. This cache continuously aggregates E2SM KPM telemetry, received from the RIC at 100 ms intervals, and maintains a history of lag values for each network metric. Consequently, the features required for the XGBoost-based prediction models are always "Inference-Ready" in the cache. This eliminates the preprocessing bottleneck, allowing the orchestrator to immediately trigger parallel utility evaluations as soon as the decision window closes.

3.2 Proactive decision windowing and conflict labeling

The MLO implements a time-slotted aggregation mechanism to handle the asynchronous and overlapping nature of xApp control requests. This mechanism serves as the temporal boundary for transforming independent requests into a collective optimization problem.

Temporal aggregation and windowing logic: While network telemetry is retrieved via E2SM KPM every 100 ms, the MLO utilizes a 200 ms fixed-windowing strategy to balance near-RT responsiveness and concurrent xApp conflict observability. Shorter aggregation windows reduce the probability of capturing overlapping control requests, whereas significantly longer windows may introduce stale network-state evaluations and delayed orchestration responses. During this interval, the MLO buffers all incoming REST-based requests instead of forwarding them immediately. This transition from a reactive "First-Come-First-Served" model to a proactive holistic model allows the orchestrator to label the entire set of received actions as a "Potential Conflict Group."

Conflict labeling: Once the window expires, the MLO identifies the number of distinct requests (N) targeting different or overlapping RCPs. By treating every concurrent request set as a potential collision, the system remains robust against both direct conflicts (parameter collisions) and indirect conflicts (KPI interference).

Stability-aware cooling mechanism: To prevent control-plane oscillations, any xApp request rejected during the resolution phase is placed into a 5 s cooling window. During this lockout period, the MLO blocks identical

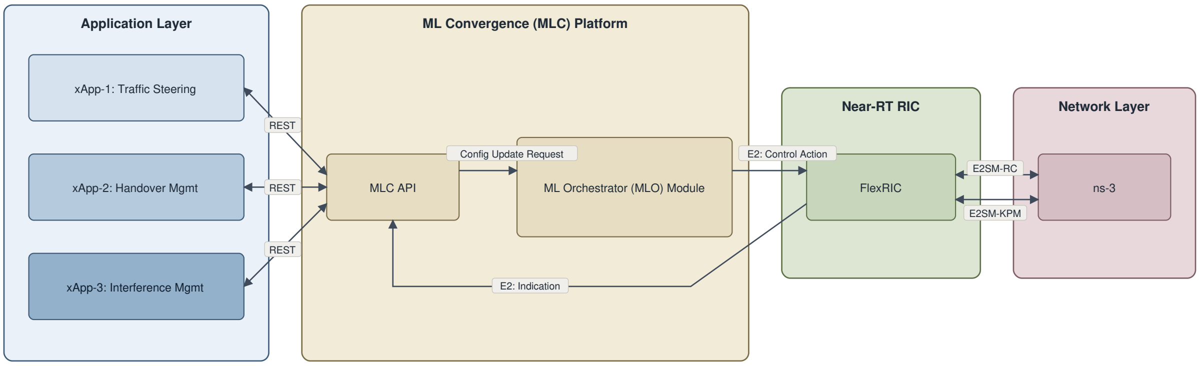

Figure 1 – ML-driven 6G O-RAN system architecture

submissions from the rejected xApp, allowing the network sufficient time to reach a steady state and mitigating high-frequency "Ping-Pong" effects in parameter tuning. The selected cooling duration is aligned with the proactive 5 s KPI prediction horizon of the MLO framework, ensuring that the impact of previously applied control actions can be sufficiently observed before new conflicting requests are reconsidered.

3.3 MLO internal workflow and parallel inference

The internal workflow of the MLO is engineered to transform "Potential Conflict Groups" into optimized, conflict-free network configurations through a high-speed, parallelized decision-making phase.

Combinatorial action space evaluation: Upon the closure of the decision window, the MLO evaluates the entire combinatorial space of 2^N possible scenarios. To meet the strict latency requirements of near-RT control, the architecture leverages a 128-core server infrastructure, enabling fully parallel inference. The computational complexity of this evaluation phase is $O(2^N \cdot M)$, where N is the number of concurrent xApps and M is the number of KPI prediction models ($M = 5$ in this study). For the evaluated $N = 3$ configuration, this yields $2^3 \times 5 = 40$ parallel XGBoost inference calls; the measured end-to-end decision latency for this configuration is 76 ms, well within the near-RT RIC control loop budget of 1 s. Empirical scaling measurements for larger N values (e.g., $N \in \{6, 9\}$) constitute a planned extension of this work.

Predictive utility selection: The orchestrator utilizes a multi-objective predictive engine to estimate the impact of each action combination on the network's KPIs. The MLO selects the specific set of actions that yields the highest predicted global utility. The detailed mathematical formulation of the utility function and the underlying XGBoost prediction models are presented in Section 5.

Standardized execution: Once the optimal combination is identified, the winning actions are mapped to the standardized E2SM-RC format. These optimized commands are then transmitted to the FlexRIC southbound interface for immediate execution in the RAN. If the "All-Apply" scenario produces the maximum score, it indicates a complementary relationship between requests; otherwise, conflicting requests are suppressed to protect the overall system utility.

4. DATA ACQUISITION AND ML MODELING

4.1 Large-scale scenario generation

To provide a robust and high-fidelity dataset for the MLO, a large-scale data acquisition campaign was conducted using the ns-3.42 discrete-event network simulator. The simulation environment was specifically designed to capture the causal impact of RAN parameter modifications on KPIs within a dynamic multi-cell environment.

Network topology and simulation setup: The experimental setup consists of a two-cell (2-gNB) topology serving a dense cluster of 60 User Equipment (UE) devices. The network operates on a millimeter-wave (mmWave) spectrum, utilizing the `ns3::mmWaveHelper` to simulate realistic channel conditions. Each UE is configured with a Constant Bit Rate (CBR) traffic model via `UdpClientHelper`, ensuring a continuous and stable data load that allows the ML models to isolate performance variations caused specifically by radio parameter tuning rather than traffic volatility.

Heterogeneous mobility profiles: To reflect a realistic and changing network state, the 60 UEs are partitioned into three heterogeneous mobility profiles, ensuring the dataset covers diverse operational conditions.

Temporal dynamics and dataset scale: Each simulation run spans a duration of 60 s, and the data acquisition process is repeated across 300 unique RNG seeds dedicated exclusively to offline ML model training. To eliminate potential data leakage, all telemetry generated from these 300 seeds was reserved solely for XGBoost training, validation, and hyperparameter tuning. The resulting dataset provided approximately 18 000 s of operational network telemetry with 100 ms KPM reporting granularity. An additional set of 200 previously unseen simulation seeds is separately generated and reserved exclusively for testing purposes.

To handle the computational load of large-scale scenario generation, a Python-based parallel execution framework was deployed on a 128-core server infrastructure. This setup enabled the concurrent execution of isolated ns-3 simulation instances, significantly accelerating dataset generation while preserving deterministic seed isolation and reproducibility across all experiments.

Exploratory parameter perturbation (random walk): To train the XGBoost models on the sensitivity of KPIs to control actions, a stochastic parameter perturbation strategy was implemented. Every 5 s (matching the orchestrator's decision horizon), the system applies a single-step change to one of the following RCPs:

- Transmission power (TxPower): Perturbed in steps of ±4 dBm within the $[30.0, 46.0]$ dBm range.
- Antenna tilt (e-Tilt): Adjusted in steps of ±2.5° within a $[5.0, 15.0]°$ range.
- A3 offset: Selected from a discrete set of values $\{-3, 0, 3\}$ dB, with constraints to prevent simultaneous −3 dB configurations across both cells to maintain handover stability.

4.2 Feature engineering and metric prediction with XGBoost

To capture the complex, non-linear dependencies between RAN configurations and network performance, raw telemetry from ns-3 simulations is transformed into a high-dimensional feature space. The pipeline resamples raw logs into a uniform 0.1 s (10 Hz) temporal grid, ensuring precise synchronization across all simulation seeds.

Temporal feature architecture: The MLO utilizes a multi-layered temporal history to inform its predictions, rather than relying on isolated snapshots. The feature set, comprising exactly 211 engineered indicators, is structured as given below.

- **Lagged and rolling observations:** To capture immediate trends and long-term stability, the model incorporates lagged features at 0.1 s (Lag1), 0.5 s (Lag5), and 1.0 s (Lag10) intervals. Additionally, a 5.0 s rolling window is applied to compute moving averages and standard deviations, providing a broader context of network load patterns.
- **Momentum and derivative features:** First-order differences (Δ) and momentum indicators are calculated to detect rapid fluctuations in SINR and throughput, enabling the model to distinguish between transient noise and persistent network trends.

Prediction target definition: The prediction target is defined as the forward rolling mean over the next 50 steps (5 s). Rather than attempting to predict a noisy, instantaneous KPI value at a single future point, the model estimates the average performance expected over the upcoming 5 s interval. This approach ensures that control actions are optimized for sustained network utility, aligning with the proactive stability requirements of the O-RAN ecosystem.

4.3 Model training and offline validation

The ML pipeline utilizes extreme Gradient Boosting (XGBoost), selected for its superior performance on structured tabular data and its ability to maintain high prediction accuracy under strict inference latency constraints.

Seed-based validation and symmetrization: To ensure robust generalization, a seed-based (run-isolated) split strategy is employed. The 300 unique simulation seeds are partitioned into an 80/20 train-test ratio. Crucially, all timesteps belonging to a specific simulation seed are kept together in either the training or testing set, preventing "temporal leakage" and ensuring that the model is tested on entirely unseen network scenarios.

Training strategy: The models are trained using 5-fold GroupKFold cross-validation, with simulation Run-IDs serving as the grouping criteria to maintain strict isolation between folds. The XGBoost configuration utilizes a `max_depth` of 7 to capture high-order interactions and a conservative `learning_rate` of 0.05 to ensure stable convergence. The training process is highly optimized through multi-threaded parallelization, allowing for rapid model iteration without compromising prediction fidelity.

5. CONFLICT RESOLUTION LOGIC AND UTILITY FUNCTION

5.1 Multi-objective utility function design

The MLO utilizes a multi-objective utility function to aggregate various network performance indicators into a single scalar value for optimal decision-making. This function acts as the optimization engine, enabling the MLO to evaluate 2^N action combinations and select the

Table 1 – Performance metrics of XGBoost models for KPI prediction

Target metric	CV avg. RMSE	CV avg. R^2	Test RMSE
Active UEs in cell 1 (count)	1.51	0.94	1.41
Active UEs in cell 2 (count)	1.52	0.94	1.41
Global avg. throughput (kbps)	254.28	0.78	272.66
Global avg. latency (ms)	0.01	0.91	0.01
Global avg. SINR (dB)	0.48	0.92	0.78
Global avg. handover event (count)	0.12	0.76	0.15
Global avg. ping-pong event (count)	0.10	0.79	0.13

configuration that maximizes the overall network utility. The contribution of this study lies in integrating predictive utility evaluation into a proactive near-RT orchestration framework, where concurrent xApp actions are jointly evaluated using ML-predicted future KPI states prior to execution.

Metric selection and causal foundation: The selection of the utility components is grounded in the causal analysis of RCP dependencies. Research indicates that physical layer parameters, such as transmission power (TxPower) and antenna tilt, exhibit direct causal impacts on SINR and throughput. Specifically, explainable ML tools like SHAP have demonstrated that TxPower is a dominant driver for throughput and spectral efficiency. Simultaneously, mobility parameters like the A3 offset are identified as primary drivers for handover stability and the occurrence of ping-pong effects in dynamic multi-cell environments [12]. Based on these established relationships and grounded in collaboration with domain network experts, the framework defines five key metrics for the utility function: global throughput, average SINR, network latency, ping-pong rate, and load imbalance.

Mathematical formulation and normalization: The total utility (U_{total}) is calculated as a weighted sum of normalized KPI predictions generated for the next 5 s horizon. The multi-objective utility expression defined in *(1)* is formulated in the present work as a weighted aggregation over the predicted KPI components.

$$U_{\text{total}} = \sum_{i=1}^{5} w_i \cdot \hat{y}_i \quad (1)$$

To handle varying units and scales, each predicted KPI ($\hat{y}_i$) is normalized within a $[0, 1]$ range. For "higher-is-better" metrics such as throughput (Th) and SINR, the normalized value is used directly. Conversely, for "lower-is-better" metrics, specifically latency (Lat), ping-pong count (PP), and load imbalance (LB), an inversion logic is applied to ensure that a reduction in these negative indicators contributes positively to the total utility, as given in *(2)*.

$$U = w_1\,\text{Th} + w_2\,\text{SINR} + w_3\,\text{Lat}_{\text{inv}} + w_4\,\text{PP}_{\text{inv}} + w_5\,\text{LB}_{\text{inv}} \quad (2)$$

Weighting policy and intent-driven flexibility: The MLO framework is intrinsically intent-driven, allowing network operators to dynamically adjust weightings based on real time priorities. This modularity enables prioritizing stability by increasing w_4 and w_5 during high-mobility scenarios to mitigate xApp conflicts, or adjusting weights to meet specific Service Level Agreement (SLA) requirements for different network slices. In the experiments reported in this paper, all five weights are set to equal values ($w_1 = w_2 = w_3 = w_4 = w_5 = 0.2$), reflecting a balanced, operator-neutral baseline.

In contrast to reactive utility evaluation approaches based solely on instantaneous KPI observations, the proposed framework performs utility estimation over ML-predicted future network states within a proactive 5 s decision horizon. This enables the MLO to evaluate the potential impact of concurrent xApp actions before execution and proactively suppress combinations that are predicted to reduce the overall network utility.

5.2 Decision-making algorithm

The decision-making process of the MLO is designed to transition from reactive parameter tuning to proactive, utility-based optimization. The algorithm effectively distinguishes between complementary and conflicting xApp requests by evaluating their collective impact on the network's future state, as shown in Algorithm 1.

Conflict identification and resolution policy: A conflict is formally defined as a scenario where the simultaneous execution of multiple xApp actions leads to a lower predicted global utility compared to the execution of a subset of those actions or the maintenance of the current state. Conversely, a non-conflict state occurs when the "All-Apply" scenario, where every received request is executed, yields the highest U_{total} among all 2^N combinations. By prioritizing the combination that maximizes the objective function, the MLO ensures that individual xApp goals do not compromise the overall stability and performance of the 6G cell.

Stability management and cooling window: To prevent oscillations from rapid resubmission by rejected xApps, the MLO enforces a 5 s cooling window: Once an action is rejected, the originating xApp cannot resubmit the same request for 5 s, allowing the network to settle on the winning configuration. This interval is deliberately matched to the XGBoost prediction horizon — the models forecast the forward rolling mean over the next 5 s (50 steps at 100 ms granularity). Issuing a new intervention before this window elapses would override the previous decision before its effects can be observed, violating the model's temporal assumptions and risking instability in the control loop. A shorter window (e.g., 1 – 2 s) would therefore be inconsistent with the model design, while a longer one (e.g., 10 s) would only delay conflict resolution without further benefit. Five seconds is thus the smallest cooling interval consistent with the prediction horizon, and prevents ping-pong effects in control signaling.

6. EXPERIMENTAL SETUP AND IMPLEMENTATION

6.1 ns-3 and FlexRIC integration

The experimental evaluation of the proposed framework is conducted using an integrated simulation environment that combines the ns-3.42 network simulator with the FlexRIC software-defined controller. This integration is realized through the ns-O-RAN-flexric open-source framework developed by Orange, which enables a high-fidelity E2 interface simulation between the RAN nodes and the near-RT RIC [19].

E2 interface and service model extensions: In this setup, each base station (gNB) in the ns-3 environment operates an E2 agent that establishes a connection with the FlexRIC southbound interface. To facilitate the testing of conflict detection and resolution logic, we utilized the existing E2SM framework within the ns-O-RAN project and specifically enhanced the functional mapping of E2SM-RC (RAN Control) messages.

Algorithm 1 Proactive conflict resolution and utility maximization

Input: Set of incoming xApp requests R, pretrained models M, current state S_{curr}
Output: Optimal action set A^*

1: $A^*, L_{eval} \leftarrow \emptyset$
2: $R_{valid} \leftarrow \{r \in R \mid r \notin C_{list}\}$
3: $Combinations \leftarrow \mathcal{P}(R_{valid})$ {Power set of requests}
4: **for all** $c_i \in Combinations$ **do in parallel do**
5: $\quad \hat{Y}_i \leftarrow \{M_1, \ldots, M_5\}(S_{curr}, c_i)$ {Predict 5 s KPIs}
6: $\quad U_i \leftarrow \sum_{j=1}^{5} w_j \cdot \text{Normalize}(\hat{y}_{i,j})$
7: $\quad L_{eval}.insert(U_i, c_i)$
8: **end for**
9: $A^* \leftarrow \text{argmax}(L_{eval})$ {Select max utility}
10: **if** $A^* \subset R_{valid}$ **then**
11: $\quad$ {Indirect/Implicit Conflict detected}
12: $\quad$ **for all** $r \in (R_{valid} \setminus A^*)$ **do**
13: $\quad\quad C_{list}.add(r, duration = 5\,\text{s})$ {Apply Lockout}
14: $\quad$ **end for**
15: **end if**
16: **return** A^*

While the standard framework provides the interface for telemetry, we implemented specific control functions within the ns-3 simulation core to handle and apply parameter modifications received via FlexRIC. This includes the functional implementation of three core control actions, as shown in Table 2.

The synchronization between FlexRIC and ns-3 ensures that control messages issued through the E2SM-RC service model are accurately reflected in the simulated physical and protocol layers, providing a realistic feedback loop for evaluating xApp interactions in a virtualized O-RAN environment [20].

6.2 Network configuration and physical layer modeling

To provide a robust and high-fidelity evaluation environment for the MLO, we designed a dense urban 6G deployment using the ns-3 mmWave-LENA module. The simulation framework is meticulously configured to serve as a digital twin that accurately reflects the physical dependencies between RCPs and KPIs.

The simulation operates on a center frequency of 3.5 GHz (FR1) while leveraging the high-fidelity PHY/MAC stack of the mmWave-LENA module. This hybrid approach allows the testbed to utilize advanced phased-array modeling and beam-forming capabilities of the mmWave module within a sub-6 GHz spectrum. All channel parameters of the propagation model presented below — including the path-loss exponent, the shadowing standard deviation, and the beamforming-gain calibration — are configured for the 3.5 GHz Urban Micro-cell (UMi) scenario rather than for mmWave bands; the mmWave-LENA module is therefore reused as a high-fidelity PHY/MAC and phased-array simulation engine, not as a mmWave channel model. The system bandwidth is fixed at 100 MHz, providing approximately 273 Physical Resource Blocks (PRBs) under 15 kHz Subcarrier Spacing (SCS). The Medium Access Control (MAC) layer utilizes the `MmWaveFlexTtiMacScheduler` with Hybrid Automatic Repeat Request (HARQ) enabled, ensuring realistic frame-level latency and throughput dynamics.

A key differentiator of our simulation is the implementation of a three-layer hybrid propagation loss model, which is critical for evaluating the impact of antenna e-Tilt and transmission power adjustments. The total

Table 2 – Mapping of RAN control parameters to E2SM-RC standard

Parameter name	E2SM-RC service style	E2SM-RC action ID	Description
NR DL TX power	2	10	Adjusts the downlink transmission power of the gNB.
A3 offset	3	2	Modifies the offset value for event A3 to trigger handovers.
Antenna tilt	10	2	Updates the vertical downtilt angle of the phased-array antenna.

received power (P_{rx}) is calculated as a cumulative effect of large-scale fading, stochastic shadowing, and directional beamforming gains:

$$P_{\text{rx}}\,[\text{dBm}] = P_{\text{tx}} - L_{\text{LD}}(d) - X_{\text{shadow}} + G_{\text{BF}}(\theta, \phi) \quad (3)$$

- **Large-scale path loss (Layer 1):** We utilize a log-distance model representative of a dense urban Non-Line-of-Sight (NLoS) environment. The path loss $L_{\text{LD}}(d)$ at distance d is formulated as:

  $$L_{\text{LD}}(d) = 43.3 + 38 \cdot \log_{10}(d)\,[\text{dB}] \quad (4)$$

  where the path loss exponent ($n = 3.8$) and the reference loss ($L_0 = 43.3\,\text{dB}$ at $d_0 = 1\,\text{m}$) are derived from standardized UMi measurements [21].
- **Stochastic shadowing (Layer 2):** To capture environmental obstructions, a zero-mean log-normal shadowing component X_{shadow} is applied with a standard deviation of $\sigma = 7.0\,\text{dB}$.
- **Two-ray spectrum propagation (Layer 3):** For physical fidelity in beam-level coverage, we employ the two-ray spectrum propagation loss model. Unlike scalar models, this model computes the received power across the full spectrum signal, facilitating the simulation of ground reflection and phased-array gain matrices [22].

The antenna system is modeled using UPA for both the gNBs and UEs, utilizing the `ThreeGppAntennaModel` to implement a 65° Half-Power Beamwidth (HPBW) element pattern.

- **gNB configuration:** Each cell is equipped with a 16×4 array (64 elements). The down-tilt angle attribute shifts the bore-sight elevation of the array, allowing the MLO to physically alter the cell coverage area.
- **UE configuration:** Each UE is equipped with a 2×2 array (4 elements) to simulate mobile beam forming gain.
- **Beam-forming execution:** The `MmWaveDftBeamforming` helper computes optimal steering weights $\mathbf{w}$ based on periodic channel estimates [23]. The directional gain G_{BF} for a steering vector $\mathbf{a}$ at azimuth ϕ and elevation θ is expressed as:

  $$G_{\text{BF}}(\theta, \phi) = \left|\mathbf{w}^H\,\mathbf{a}(\theta, \phi)\right|^2 \quad (5)$$

  This gain is dynamically updated every 100 ms, directly responding to changes in TxPower and tilt commands issued by the MLO.

To test the resilience of the conflict resolution logic, we deployed 60 UEs across a 1000 m × 400 m corridor with an Inter-Site Distance (ISD) of 500 m. The mobility distribution is tiered to represent a diverse urban traffic environment [24].

The traffic model consists of a UDP/CBR downlink stream per UE, with an offered load of 5 Mbps (1024-byte packets at 1 639 μs intervals). This creates a high-interference environment in which the MLO must jointly handle load balancing and MRO.

6.3 Implementation of rule-based xApps and conflict scenarios

In this study, we implemented three independent data-driven xApps within the FlexRIC-enabled ns-3 environment. While the target 6G-SMART architecture and the MLC pipeline ultimately envision fully autonomous ML-driven xApps, the current evaluation intentionally employs deterministic rule-based implementations to provide controlled, reproducible, and interpretable conflict scenarios. The trigger thresholds and action logic of each xApp were derived empirically from an exploratory data analysis of 300 preliminary simulation runs (18 000 s of network telemetry), ensuring that the rules reflect statistically grounded, network-improving decisions rather than arbitrary heuristics. Since the proposed MLO framework operates on incoming control intents and their predicted KPI impacts rather than the internal optimization mechanisms of individual xApps, the orchestration workflow remains inherently decoupled from the underlying decision-generation methodology, isolating the effectiveness of orchestration and conflict-resolution logic from stochastic variations, convergence behavior, or instability originating from independently evolving optimization agents.

We acknowledge that rule-based xApps exhibit deterministic, predictable behavior that may not fully repli-

Table 3 – Heterogeneous mobility profile configuration

Mobility group	UE count	Velocity	ns-3 mobility model
Static UEs	20	0 km/h	`ConstantPosition`
Pedestrian UEs	20	3–5 km/h	`RandomWalk2d`
Vehicular UEs	20	36–72 km/h	`ConstantVelocity`

cate the non-stationary dynamics of ML-driven xApps; the observed conflict distribution (64.7% modification / 35.3% complementary) should therefore be interpreted as a lower-bound characterization of the MLO's conflict detection capabilities under controlled conditions. Because the utility-based interceptor operates on observed E2SM-RC control messages rather than on internal xApp logic, the architecture remains applicable to production deployments employing online learning or adversarial exploration strategies, although the absolute conflict rate and resolution distribution may differ; evaluating the MLO with ML-driven xApps is a planned extension of this work within the broader 6G-SMART roadmap.

Threshold determination via statistical analysis: The operational logic for all three xApps was derived from an extensive preliminary data generation phase consisting of 300 independent simulation runs. By analyzing the distribution of signal quality, traffic load, and mobility stability across 18 000 s of network activity, we identified the statistical "elbow points" used to define the xApp triggers.

6.3.1 Inter-site interference management xApp

- **Metrics subscription:** This xApp subscribes to real time SINR and throughput telemetry streams from the near-RT RIC.
- **Controlled parameter:** It modifies the NR DL TX power (parameter ID: 1) to balance coverage and interference.
- **Operational logic:** The xApp triggers an emergency power increase when signal degradation is detected or executes an energy-saving reduction when channel conditions are optimal, utilizing context-aware lookup tables based on current power levels.

6.3.2 Inter-site traffic steering management xApp

- **Metrics subscription:** This xApp monitors active UE counts (Load) per cell and UE distance history to evaluate spatial traffic distribution.
- **Controlled parameter:** It manipulates the antenna e-Tilt angle to physically redirect traffic and manage cell boundaries.
- **Operational logic:** The xApp focus is on load balancing; it increases tilt to narrow the beam on oversaturated cells (user shedding) and decreases tilt to expand coverage on under-utilized cells.

6.3.3 Ping-pong handover management xApp

- **Metrics subscription:** This xApp subscribes to handover statistics, specifically the ping-pong rate calculated over a sliding temporal window.
- **Controlled parameter:** It optimizes the A3 offset (parameter ID: 7), which serves as the hysteresis margin for mobility events.
- **Operational logic:** Upon detecting high instability, the xApp increases the hysteresis margin to suppress rapid, unnecessary handovers, or reduces it during stable conditions to facilitate seamless UE mobility.

7. RESULTS

To evaluate the effectiveness of the proposed MLO framework, an extensive experimental campaign was conducted within a high-fidelity simulation environment. The testing framework integrated the ns-3 network simulator with the FlexRIC near-RT RIC platform, enabling the execution of the three specialized xApps under the management of the MLO. The evaluation was performed across 200 independent simulation seeds, each spanning a duration of 50 s. These 200 seeds are entirely disjoint from the 300 seeds used for XGBoost training, yielding a combined experimental footprint of 500 unique RNG seeds with zero overlap, thereby precluding any possibility of data leakage between the training and evaluation phases. This resulted in a comprehensive dataset of 10 000 s of active network telemetry, providing a statistically significant basis for analyzing conflict resolution efficiency, predictive accuracy, and overall utility gains in a dynamic 6G environment.

7.1 Conflict detection and resolution statistics

Throughout the resulting 200 seeds observation period, the MLO identified a total of 521 potential conflicts, occurring within the defined decision windows. The resolution outcomes for these potential conflicts are categorized into two, as depicted in Fig. 2.

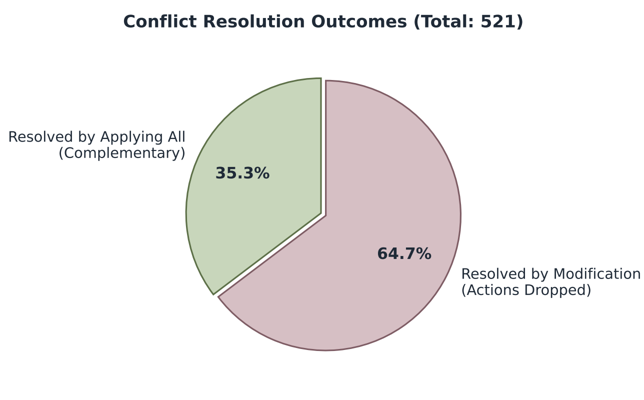

Figure 2 – Potential conflict distribution

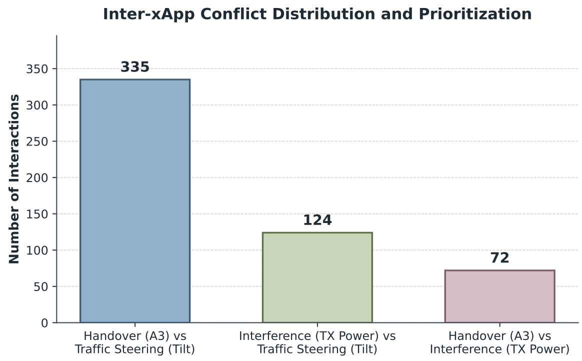

Figure 3 – Inter-xApp conflict distribution

- **Resolved by modification (actions dropped):** In 337 cases (64.7%), the orchestrator detected actual conflicts where simultaneous execution would have compromised system utility. In these instances, the MLO intervened by dropping or modifying specific xApp actions.
- **Resolved by applying all (complementary):** In 184 cases (35.3%), the requests were identified as complementary or safe for parallel execution, allowing the orchestrator to apply the full set of proposed actions.

The Inter-xApp conflict distribution provides further insight into the prioritization logic across the 521 detected events, as shown in Fig. 3 and detailed below.

- **Ping-pong handover management vs. inter-site traffic steering management (335 interactions):** This was the most frequent interaction. When a choice was required, the MLO prioritized the inter-site traffic steering management xApp (antenna tilt) over the ping-pong handover management xApp (A3 offset), with tilt adjustments being 2.2x more likely to be retained to preserve spatial load balancing.
- **Inter-site interference management vs. inter-site traffic steering management (124 interactions):** These interactions were found to be very balanced, with inter-site traffic steering maintaining a slight prioritization edge over the inter-site interference management xApp (TX power control).
- **Ping-pong handover management vs. inter-site interference management (72 interactions):** In mobility-signal quality conflicts, the inter-site interference management xApp (TX power) was prioritized more often than the ping-pong handover management xApp (A3 offset), addressing the root cause of signal degradation rather than merely delaying the handover event.

7.2 Predictive performance and fidelity analysis

The predictive fidelity of the MLO was evaluated by comparing the performance forecasts generated during the 521 conflict instances against the actual 5 s ground-truth averages recorded in the ns-3 environment. Across individual metrics, MLO demonstrated highly accurate forecasting, achieving an exceptional 90.71% accuracy for the total utility score (0.737 predicted vs. 0.812 actual), as shown in Fig. 4. At the component level, the predictions showcased remarkable precision for throughput (95.35% accuracy; 4.029 Mbps predicted vs. 3.850 Mbps actual), load balance (93.13% accuracy; 0.698 predicted vs. 0.750 actual), and network latency (91.92% accuracy; 1.261 ms predicted vs. 1.167 ms actual). Even for highly dynamic factors, stability and signal quality maintained strong accuracies of 86.19% (0.517 predicted vs. 0.600 actual) and 71.67% (13.55 dB predicted vs. 10.56 dB actual), respectively. This comparison validates that orchestrator decisions are grounded in high-fidelity predictions, ensuring temporal stability between the 200 ms decision window and the subsequent network state.

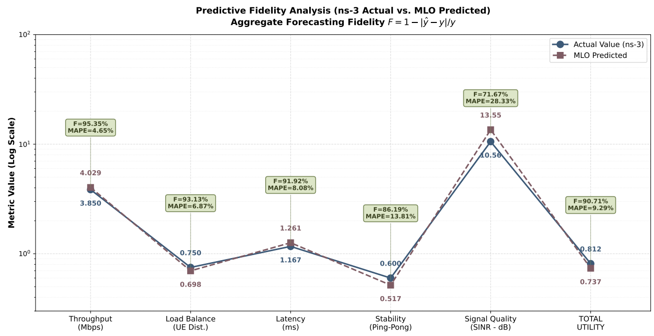

Figure 4 – Predictive fidelity analysis graph

7.3 Comparative performance analysis: MLO-enabled vs. traditional architecture

To quantify the operational benefits of the proposed framework, we conducted a controlled comparison between the MLO-enabled system and an uncoordinated baseline. In the baseline configuration, xApps operate as independent silos without a centralized coordination layer, resulting in unmanaged concurrent control actions; this corresponds to the traditional deployment approach in current multi-vendor O-RAN environments and captures the behavior the MLO is designed to improve upon. The evaluation focuses on the average network performance during the 5 s post-conflict window following each of the 521 detected conflict events. Benchmarking against additional conflict-arbitration schemes from the literature is planned as an extension of this work.

As demonstrated by the 200-seed simulation results, the MLO-enabled architecture consistently outperforms the traditional approach by preventing performance degradation typically caused by xApp oscillations, culminating in an overall +2.18% improvement in the total utility score (0.8310 vs. 0.8092 baseline), as shown in Fig. 5. To confirm that these gains are not attributable to stochastic simulation noise, a two-tailed paired t-test was conducted

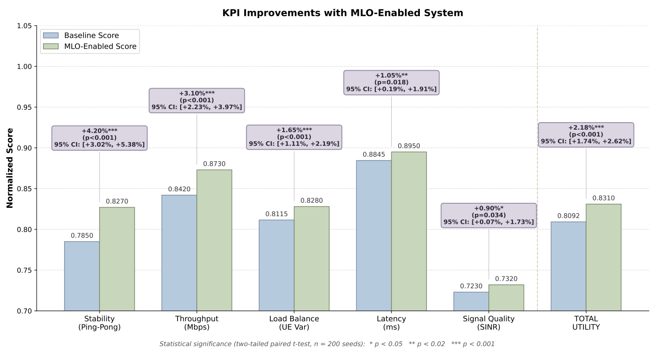

Figure 5 – KPI improvements with the MLO-enabled system across five components and the aggregate utility score

across the 200 independent seed pairs. The total utility improvement is statistically highly significant ($p < 0.001$, 95% CI: [+1.74%, +2.62%]). The most dramatic operational gains were observed in stability (ping-pong rate) at +4.20% (0.8270 vs. 0.7850 baseline, $p < 0.001$), and throughput at +3.10% (0.8730 vs. 0.8420, $p < 0.001$). Other critical network components also exhibited consistent positive gains over the baseline, including load balance (+1.65%, $p < 0.001$), latency (+1.05%, $p = 0.018$), and signal quality (+0.90%, $p = 0.034$, 95% CI: [+0.07%, +1.73%]), demonstrating the effectiveness of centralized conflict resolution.

7.4 Micro-level analysis of specific conflict scenario

This scenario illustrates a direct operational conflict at Node-002, where the inter-site interference management xApp (requested TargetPower: 42.0) and the ping-pong handover management xApp (requested A3_Offset: −3 dB) submitted simultaneous control requests. As shown in Table 4, the MLO performed a 5-aspect combinatorial evaluation to determine the optimal action set:

- **Scenario evaluation:** The joint execution of both actions (C2_POWER + C2_A3) yielded a utility of 0.5370, whereas applying only the A3_Offset adjustment (C2_A3) resulted in a superior utility score of 0.5910.
- **The decision:** To avoid a "Utility Gap," the MLO proactively rejected the TxPower request from xApp_1 and exclusively applied the handover optimization.

The impact of this orchestration is clearly visible in the real time telemetry graphs (Fig. 6). Following the intervention (marked by the red vertical line), the MLO-enabled (orchestrator ON) system significantly diverges from the traditional architecture. While the unmanaged baseline (orchestrator OFF) suffered from SINR degradation and increased handover oscillations, the MLO-enabled system achieved a +8.51% gain in total network utility. Specifically, this intervention resulted in a +3.02% increase in throughput, a +9.30% improvement in SINR, and a critical −16.45% reduction in ping-pong handovers, validating the orchestrator's ability to maintain high-fidelity performance during concurrent xApp operations.

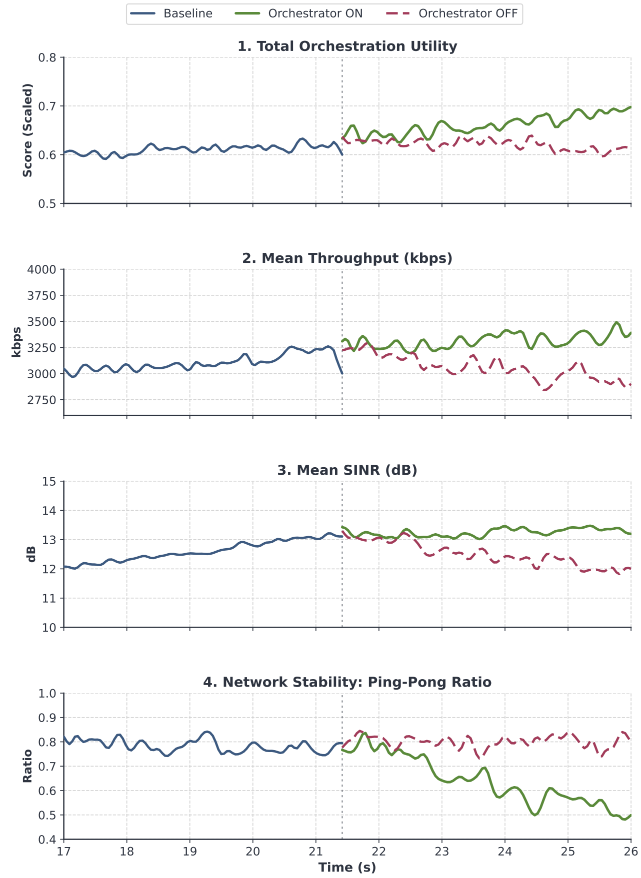

Figure 6 – Performance metrics comparison

8. CONCLUSION

This paper presented a proactive ML Orchestrator (MLO) designed as an intelligent interceptor layer within the O-RAN Near-Real-Time RIC architecture, where decoupling third-party xApps from the RIC core through the ML Convergence (MLC) platform allows for a successful transition from reactive parameter tuning to proactive, utility-driven orchestration. The proposed framework remains inherently xApp-agnostic, as orchestration decisions are performed over concurrent control intents and their predicted KPI impacts rather than the internal optimization methodology of individual xApps.

Extensive evaluation conducted within a high-fidelity ns-3 and FlexRIC environment validates that the integration of high-speed XGBoost ensembles within a narrow 200 ms decision window is sufficient to capture and forecast complex RAN dynamics over a multi-second

Table 4 – Multi-objective utility evaluation for conflict resolution at Node-002

Action set	Throughput	Latency	Stability	Load balance	SINR	Total util (U_{total})	Status
POWER (xApp-1)	0.211	0.863	0.188	0.863	0.521	0.5100	Rejected
POWER + A3 (joint)	0.233	0.792	0.213	0.913	0.565	0.5370	Rejected
A3 (xApp-2)	0.267	0.738	0.248	0.952	0.624	0.5910	**Applied**

horizon, enabling the system to mitigate performance degradation before it occurs. Furthermore, the MLO-enabled architecture effectively identifies and resolves operational conflicts that would otherwise lead to network instability by prioritizing physical layer reconfigurations over logical parameter adjustments to maintain superior spatial equilibrium in dense urban scenarios. Compared to the traditional architecture, the proposed framework demonstrates enhanced system-wide stability and a significant reduction in control plane oscillations, proving that proactive mitigation of mobility-related instabilities and centralized orchestration are essential for the reliability of multi-vendor 6G environments.

Empirical scaling measurements of the orchestrator's end-to-end decision latency for larger numbers of concurrent xApps (e.g., $N \in \{6, 9\}$) are planned as future work to validate the $O(2^N \cdot M)$ complexity behavior beyond the $N = 3$ configuration evaluated in this study. Likewise, because the orchestrator operates at the E2SM-RC control-message layer rather than on carrier-frequency-specific channel models, the MLO architecture is inherently carrier-frequency agnostic; quantitative evaluation under upper-mid-band (FR3) and mmWave (FR2) deployments, beyond the 3.5 GHz FR1 baseline studied here, is identified as a planned extension of this work.

In summary, the 6G-SMART MLO framework provides a resilient and scalable foundation for the AI-native orchestration required by autonomous 6G networks, with future work extending this platform to support online learning, adaptive optimization of orchestration parameters such as decision-window and cooling durations, and evaluating its performance in large-scale hexagonal deployments with heterogeneous network slices.

ACKNOWLEDGEMENT

This work was supported by UKRI (UK ID: 640240), CDTI (ES), ANI (PT ID: COMPETE2030-FEDER-01602800), and TUBITAK (TR ID: 9240013) within the EUREKA CELTIC-NEXT project 6G-SMART, C2023/2-20.

This work was also supported in part by The Scientific and Technological Research Council of Turkiye (TUBITAK) 1515 Frontier R&D Laboratories Support Program for Turkcell 6GEN. LAB under Project 5229902.

REFERENCES

[1] ITU-R. *Framework and Overall Objectives of the Future Development of IMT for 2030 and Beyond*. Tech. rep. Recommendation ITU-R M.2160-0. International Telecommunication Union, Nov. 2023.

[2] Ting Li, Qi Sun, Jinri Huang, Yang Wang, Xiang Li, Nan Li, and Chih-Lin I. "AI-Native Management and Orchestration in O-RAN: Enabling the Path to Intelligent 6G Automation". In: *IEEE Vehicular Technology Magazine* 20.4 (2025), pp. 36–45. DOI: 10.1109/MVT.2025.3623774.

[3] Mehmet Başaran, Semih Aktaş, and Buse Bilgin. "6G Network Slicing Vision for Post-Disaster: AI-Enabled User Prioritization and Energy Management". In: *2024 IEEE International Conference on Advanced Telecommunication and Networking Technologies (ATNT)*. Vol. 1. 2024, pp. 1–4. DOI: 10.1109/ATNT61688.2024.10719190.

[4] Partemie-Marian Mutescu, Adrian-Ioan Petrariu, Eugen Coca, Cristian Patachia-Sultanoiu, Razvan Marius Mihai, and Alexandru Lavric. "AI-Native PHY-Layer in 6G Orchestrated Spectrum-Aware Networks". In: *Sensors* 25.23 (2025). ISSN: 1424-8220. DOI: 10.3390/s25237206. URL: https://www.mdpi.com/1424-8220/25/23/7206.

[5] Bharat Agarwal, Ralf Irmer, David Lister, and Gabriel-Miro Muntean. "Open RAN for 6G Networks: Architecture, Use Cases and Open Issues". In: *IEEE Communications Surveys & Tutorials* (2025). DOI: 10.1109/COMST.2025.3562429.

[6] Zexu Li, Qingtian Wang, Yue Wang, and Tao Chen. "The Architecture of AI and Communication Integration towards 6G: An O-RAN Evolution". In: *Proceedings of the 30th Annual International Conference on Mobile Computing and Networking*. ACM MobiCom '24. Washington D.C., DC, USA: Association for Computing Machinery, 2024, pp. 2329–2334. ISBN: 9798400704895. DOI: 10.1145/3636534.3701550. URL: https://doi.org/10.1145/3636534.3701550.

[7] Osman Tugay Basaran, Mehmet Basaran, Derya Turan, Hamide Gul Bayrak, and Yagmur Sabucu Sandal. "Deep Autoencoder Design for RF Anomaly Detection in 5G O-RAN Near-RT RIC via xApps". In: *Proc. IEEE Int. Conf. Commun. Workshops (ICC Workshops)*. 2023, pp. 549–555. DOI: 10.1109/ICCWorkshops57953.2023.10283501.

[8] O-RAN Alliance. *O-RAN.WG3.TS.E2SM-KPM-R004-v07.00: E2 Service Model (E2SM), KPM*. Technical Specification. O-RAN Alliance, 2025.

[9] O-RAN Alliance. *O-RAN.WG3.TS.E2SM-RC-R004-v09.00: E2 Service Model (E2SM), RAN Control*. Technical Specification. O-RAN Alliance, 2025.

[10] Abdul Wadud, Fatemeh Golpayegani, and Nima Afraz. "xApp-Level Conflict Mitigation in O-RAN, a Mobility Driven Energy Saving Case". In: *IEEE INFOCOM 2025 - IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*. 2025, pp. 1–6. DOI: 10.1109/INFOCOMWKSHPS65812.2025.11153006.

[11] O-RAN Alliance. *O-RAN.WG3.TR.ConMit-R004-v01.00: Conflict Mitigation*. Technical Report. O-RAN Alliance, 2024.

[12] Pragya Sharma, Shihua Sun, Shachi Deshpande, Angelos Stavrou, and Haining Wang. "Towards xApp Conflict Evaluation with Explainable Machine Learning and Causal Inference in O-RAN". In: *2025 IEEE Conference on Network Function Virtualization and Software-Defined Networking (NFV-SDN)*. IEEE, Nov. 2025, pp. 1–7. DOI: 10.1109/NFV-SDN66355.2025.11349452. URL: http://dx.doi.org/10.1109/NFV-SDN66355.2025.11349452.

[13] Hakan Erdol, Xiaoyang Wang, Robert Piechocki, George Oikonomou, and Arjun Parekh. "xApp distillation: AI-based conflict mitigation in B5G O-RAN". In: *Computer Networks* 274 (2026), p. 111848. ISSN: 1389-1286. DOI: 10.1016/j.comnet.2025.111848. URL: https://www.sciencedirect.com/science/article/pii/S138912862500814X.

[14] Anastasios E. Giannopoulos, Sotirios T. Spantideas, George Levis, Alexandros S. Kalafatelis, and Panagiotis Trakadas. "COMIX: Generalized Conflict Management in O-RAN xApps—Architecture, Workflow, and a Power Control Case". In: *IEEE Access* 13 (2025), pp. 116684–116700. DOI: 10.1109/ACCESS.2025.3585774.

[15] M. Al Shami et al. *O-RAN xApps Conflict Management using Graph Convolutional Networks*. 2025. arXiv: 2503.03523 [cs.NI].

[16] I. Cinemre et al. *xApp Conflict Mitigation with Scheduler*. 2025. arXiv: 2504.06867 [cs.NI].

[17] CELTIC-NEXT. *6G-SMART: 6G Self-Organising and Managing Open Radio Access Networks*. Project leaflet. 2024.

[18] Can Kurtulan, Onur Kalinagac, Erdinc Ozdemir, Burak Gorkemli, Semih Aktas, and Mehmet Basaran. "ML-Based Proactive Conflict Detection and Resolution for 6G Open Radio Access Networks". In: *Proc. International Conference on Telecommunications (ICT 2026)*. Thessaloniki, Greece, May 2026.

[19] Orange OpenSource. *ns-O-RAN-flexric*. GitHub repository. 2024. URL: https://github.com/Orange-OpenSource/ns-O-RAN-flexric.

[20] Mosaic5G. *FlexRIC: A Software Development Kit for the Near-Real-Time RAN Intelligent Controller*. GitLab repository. 2024. URL: https://gitlab.eurecom.fr/mosaic5g/flexric.

[21] ns-3 Project. *ns-3 Model Library: Propagation Models*. ns-3 documentation. 2024. URL: https://www.nsnam.org/docs/models/html/propagation.html.

[22] ns-3 Project. *Spectrum Module and Two-Ray Ground Models*. ns-3 documentation. 2024. URL: https://www.nsnam.org/docs/models/html/spectrum.html.

[23] ns-3 Project. *Antenna Module: ThreeGpp AntennaModel and PhasedArrayModel*. ns-3 documentation. 2024. URL: https://www.nsnam.org/docs/models/html/antenna.html.

[24] ns-3 Project. *Mobility Module: RandomWalk2d and ConstantVelocity Models*. ns-3 documentation. 2024. URL: https://www.nsnam.org/docs/models/html/mobility.html.

AUTHORS

CAN KURTULAN received his B.S. degrees in electrical and electronics engineering and computer engineering (double major) from Koç University in 2025, following his graduation from VKV Koç High School in 2020. He has previously held internship positions as an avionics engineering intern at Turkish Technic, a software consultant intern at OPTiiM, and an AI/ML engineer intern at Odine. He is currently an AI/ML engineer at Odine. His professional focus centers on the architectural design of AI/ML-driven systems and the deployment of intelligent applications to solve complex industrial challenges, with a current specialization in telecommunication infrastructures. His research interests include 6G network optimization, AI/ML integration in Open RAN (O-RAN), and predictive modeling for intelligent network management.

ONUR KALINAGAC is an AI&ML engineer at OdineLabs, where he works on AI/ML-enabled telco systems with a focus on anomaly detection and machine learning orchestration for next-generation communication networks. He brings experience in backend engineering, cloud-native microservices, software-defined networking, and applied machine learning, having previously worked at Vestek, Argela, Zurich University of Applied Sciences, and Epicurus Labs. He received a B.S. degree in electrical and electronics engineering and an M.S. degree in computer engineering from Bogazici University, Istanbul, Turkey, in 2015 and 2022, respectively, and is the author or co-author of peer-reviewed publications in software-defined networking, UAV-assisted edge networks, microservice root-cause analysis, and AI/ML-enabled communication systems.

ERDINC OZDEMIR is working as product requirements engineer at OdineLabs, where he oversees the management for AI-based telco-cloud orchestration products. He brings over 15 years of experience in wireless technologies across multi-vendor environments, having held positions as LTE principal engineer, O-RAN technical project manager, and SON product design and delivery expert, with a domain spanning 2G/3G/4G/5G mobile network optimization and planning, O-RAN architecture, and presales activities. As a SON product development architect and mentor, he has led the delivery of LTE KPI acceptance, SON product acceptance, O-RAN trial deployments, and SON product design programmes, gaining first-hand insight into telecom-operator product needs and how to enhance and improve their network performance. Prior to joining OdineLabs, he served as technical project manager for O-RAN technology at Mavenir, where he led the O-RAN 7.2 LTE test and trial-deployment activities, the preparation of CIQ documents, and the technical coordination of the O-RAN proof-of-concept programme between R&D and customers.

Burak Gorkemli is product & innovation director at OdineLabs, where he leads the product strategy with a focus on leveraging AI to optimize and automate next-generation telco and enterprise cloud infrastructures, as well as services. He brings over 25 years of telecommunications experience spanning 3G/4G/5G, IP networking, xPON, SDN & NFV; and has coordinated more than ten research projects funded by TÜBİTAK, SSB, and the European Union. Dr. Gorkemli holds a Ph.D. degree from Koc University, Istanbul, and is the inventor on 17 patents and the author of numerous peer-reviewed papers. His current research interests include artificial intelligence, network optimization and automation, network intelligence, and next-generation communication systems.

Gokhan Karakus is director of big data & AI applications at OdineLabs, where he founded and leads the big data & AI competence area and drives the design of cloud-native AI platforms that integrate data lakes, AI/ML model lifecycles, and LLM-powered agentic workflows with end-to-end data governance. He brings over 20 years of experience in telecommunications, big data, and AI/ML platforms, and has previously held senior roles, including technology director and business unit director, at similar technology firms, leading multidisciplinary engineering teams to deliver high-concurrency distributed telco-grade applications, nationwide regulatory platforms, and high-performance deep packet inspection solutions serving millions of subscribers. He received a B.Sc. degree from Istanbul Technical University, an M.Sc. degree from Bogazici University, and an Executive MBA from Bahcesehir University. His current research interests include predictive maintenance, anomaly detection, and zero-touch automation for telecom operators and industrial enterprises.

Semih Aktas received B.Sc., M.Sc., and Ph.D. degrees in computer engineering from Middle East Technical University (METU). He is currently a 6G researcher with Turkcell, focusing on the integration of artificial intelligence into next-generation mobile networks. He has been actively involved in national and international research and development projects, with a particular emphasis on AI-native network architectures, intelligent resource management, and energy-efficient mobile network operations. His research interests include the application of machine learning, deep reinforcement learning, and data-driven optimization techniques to improve the performance, efficiency, and intelligence of wireless communication systems.

Mehmet Basaran received B.Sc. and M.Sc. degrees in electrical and electronics engineering from Istanbul University, Istanbul, Turkey, in 2008 and 2011, respectively, and a Ph.D. degree in electronics and communication engineering from Istanbul Technical University (ITU), Istanbul, in 2018, where he was a research assistant. Dr. Basaran was a visiting researcher with RWTH Aachen University, Aachen, Germany, in 2017 in the context of the FET EU Horizon 2020 Project. He was an R&D operations manager at ITU Vodafone Future Lab between 2018 and 2021. He was a 5G research professional with Siemens Turkey between 2021 and 2023. Dr. Basaran is currently principal 6G researcher with Turkcell's Frontier R&D Lab, "6GEN. LAB: 6G and AI Lab.", which he started in February 2023. His general research areas mainly include next-generation communication systems and signal processing for wireless communications.