Demonstration of LITE: Trajectory-Unaware CSI Estimation as an O-RAN xApp for CF-MaMIMO

David Góez*, Marco Piazzola†, Giulia Costa†
Nina Slamnik-Kriještorac*, Johann Marquez-Barja*, Miguel Camelo Botero*

*University of Antwerp - imec, IDLab, Belgium
†Spindox Labs SRL, Italy

***Abstract*—We present LITE, a lightweight trajectory-unaware CSI estimation framework implemented as an O-RAN xApp and integrated within a CF-MaMIMO emulator. LITE compresses high-dimensional CSI at the O-DU, transports a compact latent representation over the midhaul, and predicts short-horizon channel gains at the Near-RT RIC using a compact SE-BiLSTM model. The demo highlights end-to-end real-time operation, interactive visualization, fault injection, and fallback mechanisms under realistic impairments such as missing or delayed measurements. Results show stable and accurate per-AP predictions while meeting Near-RT latency constraints, demonstrating the feasibility of embedding bandwidth-aware intelligence in O-RAN-compliant RAN loops.**

***Index Terms*—CF-MaMIMO, O-RAN, xApp, CSI prediction, real-time RAN control, midhaul compression.**

I. INTRODUCTION

Cell-Free Massive Multiple-Input Multiple-Output (CF-MaMIMO) promises higher coverage and spectral efficiency by jointly serving User Equipments (UEs) with many distributed Access Points (APs). However, transporting high-dimensional Channel State Information (CSI) from radios toward control logic imposes significant midhaul overhead and tight timing constraints following the Open Radio Access Network (O-RAN) architecture. In practice, Near-Real-Time RAN Intelligent Controller (Near-RT-RIC) operation requires both bandwidth-aware representations of CSI and lightweight predictors that can close the control loop without violating latency budgets.

Trajectory-aware approaches rely on explicit position, trajectory, or map information, which may be unavailable, noisy, or privacy-sensitive. In contrast, trajectory-unaware forecasting leverages temporal and spatial regularities in radio observations alone, but is often evaluated offline or decoupled from deployable control stacks. As a result, the community lacks a demonstration that integrates midhaul-efficient CSI compression with on-the-loop prediction inside an O-RAN-compliant workflow that can be interacted with in real-time.

In this demonstration paper, we present the integration of LITE (Lightweight Intelligent Trajectory Estimator) [1] as an xApp in an O-RAN-enabled CF-MaMIMO emulator for real-time short-horizon CSI prediction. LITE compresses CSI at the O-RAN Distributed Unit (O-DU) using a lightweight 1-D convolutional autoencoder, transports a compact latent representation over the midhaul, and reconstructs and predicts future CSI values (e.g., gains, RX, etc.) at the Near-RT-RIC using a compact Squeeze-and-Excitation (SE)-Bidirectional Long Short-Term Memory (BiLSTM). The xApp exposes REST control APIs and a WebSocket telemetry stream, provides a minimal web User Interface (UI) for configuration and visualization, and implements data validation, fault injection, and fallback policies to demonstrate robustness under realistic impairments (e.g., missing or delayed measurements and AP-set changes).

This demo contributes:

- *End-to-end integration*: a Near-RT-RIC control loop connecting a CF-MaMIMO emulator to an O-RAN xApp, enabling live configuration, streaming measurements, and real-time prediction visualization.
- *Midhaul-efficient learning*: a convolutional autoencoder for CSI compression coupled with a lightweight SE-BiLSTM predictor, preserving forecasting fidelity under reduced transport overhead.
- *Robust and deployable design*: containerized deployment (Docker/Kubernetes), Taskfile-driven workflows, automated testing, and interactive fault injection with selectable fallback strategies (previous measurement vs. last prediction).

Together, these elements demonstrate a practical approach to embedding bandwidth-aware, Near-Real-Time intelligence within O-RAN systems, allowing attendees to interact with the system, observe runtime behavior, and assess trade-offs between compression, latency, and prediction accuracy.

II. LITE ARCHITECTURE

The LITE system architecture, illustrated in Fig. 1, enables lightweight CSI compression and prediction within an O-RAN-compliant CF-MaMIMO framework [1]. The design spans both the radio access domain and the Near-RT-RIC, while requiring no modifications to existing APs or O-RAN Radio Units (O-RUs), ensuring seamless integration with standard RAN processing pipelines.

The architecture is organized into four layers: (i) CF-MaMIMO radio access, (ii) O-DU processing, (iii) midhaul transport, and (iv) Near-RT-RIC execution.

ISBN 978-3-903176-82-9 © 2026 IFIP

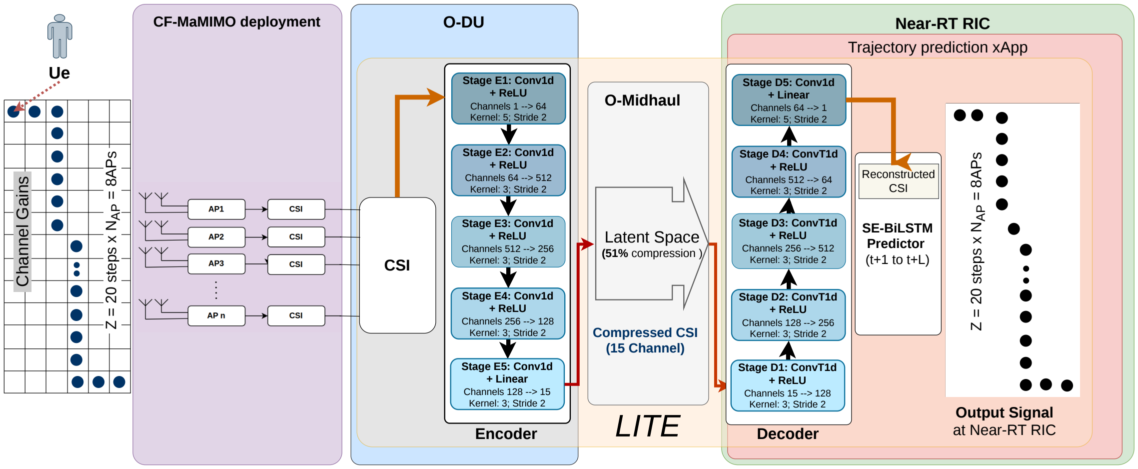


Fig. 1. Overview of the LITE system architecture, illustrating the four processing layers.

At the radio layer, uplink pilot-based estimation yields complex-valued CSI following the O-RAN 7.2x split, represented as

$$H(t) \in \mathbb{C}^{N_{\mathrm{AP}} \times N_{\mathrm{ant}} \times N_{\mathrm{sb}}} \tag{1}$$

Under flat-fading conditions, this representation is reduced to large-scale channel gains $\boldsymbol{\beta}(t) \in \mathbb{R}^{N_{\mathrm{AP}}}$, which are organized into a temporal window

$$S \in \mathbb{R}^{N_{\mathrm{AP}} \times Z} \tag{2}$$

serving as input to the LITE pipeline.

At the O-DU, S is normalized and encoded by a lightweight DNN-based autoencoder into a compact latent representation $L(t)$, with $|L| \ll |H|$, preserving key spatial-temporal features while significantly reducing dimensionality.

The latent representation is transported over the midhaul via the E2 interface, reducing bandwidth consumption and enabling efficient data exchange between distributed RAN components.

At the Near-RT-RIC, LITE performs three main operations:

1) **Decoding:** reconstruction of $\widetilde{S}(t)$ from $L(t)$.
2) **Prediction:** a compact SE-BiLSTM model predicts future channel gains $\widehat{\boldsymbol{\beta}}(t+1 \ldots t+L)$, leveraging channel-wise attention for efficient temporal modeling.
3) **Adaptation:** outputs are formatted as KPM-like vectors for visualization and potential integration with downstream RRM functions.

III. DEMO DESCRIPTION

The demonstration is based on the 6G-BRICKS CF-MaMIMO emulator, which synthesizes user equipment (UE) mobility patterns, radio propagation conditions, and Key Performance Measurement (KPM) streams within a dynamic indoor deployment scenario [2]. An illustrative APs layout and representative UE trajectory are presented in Fig. 2, while the overall emulator architecture is depicted in Fig. 3.

The emulator drives the LITE pipeline in a closed loop, where streaming KPMs are processed in real time by the xApp.

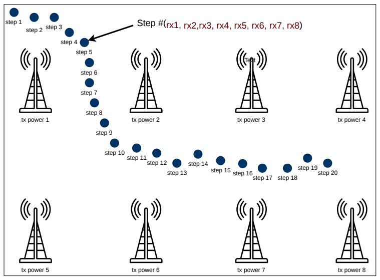


Fig. 2. Indoor CF-mMIMO deployment scenario with AP layout and UE trajectory.

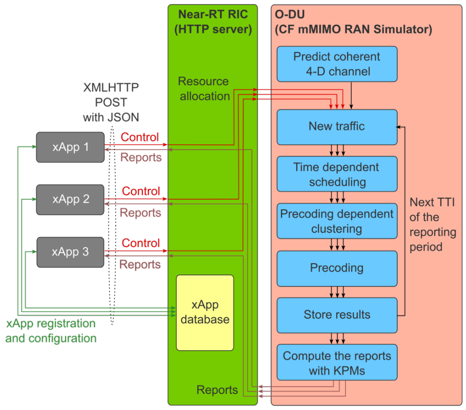


Fig. 3. 6G-BRICKS CF-mMIMO emulator architecture [2].

The end-to-end data path is depicted in Fig. 4. Incoming RX-power measurements are validated, normalized, and organized into windowed sequences, which are then processed by the decoding and prediction modules. The SE-BiLSTM model produces short-horizon per-AP channel-gain estimates, which are exposed via APIs and visualized live in the dashboard.

Dataset-aligned configuration: The emulator is configured to match the structure of the deployment used in [3], ensuring

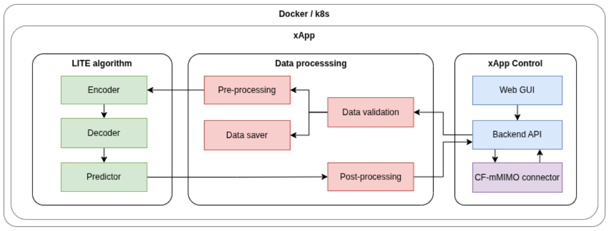


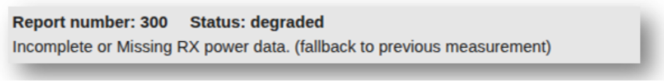
Fig. 4. LITE–simulator data path: KPM ingestion, validation and windowing, decoding, prediction, and real-time visualization.

Fig. 5. LITE xApp dashboard showing control interface, real-time per-AP measurements and predictions, and system metrics.

TABLE I

MEASURED VS. PREDICTED RX POWER ACROSS ACCESS POINTS (W).

AP	Measured	Predicted
AP1	3.2×10^{-8}	3.1×10^{-8}
AP2	1.42×10^{-7}	1.27×10^{-7}
AP3	9.0×10^{-8}	8.2×10^{-8}
AP4	0	2.7×10^{-8}
AP5	3.8×10^{-8}	3.1×10^{-8}
AP6	5.0×10^{-8}	5.3×10^{-8}
AP7	2.4×10^{-8}	2.6×10^{-8}


Report number: 300 **Status: degraded**

Incomplete or Missing RX power data. (fallback to previous measurement)

Fig. 6. Fault-handling banner: report #300 flagged as *degraded* due to missing RX power; the xApp applies the *previous measurement* fallback.

direct compatibility with the LITE pipeline. Streaming RX-power KPMs are mapped to the windowed representation $\mathbf{S} \in \mathbb{R}^{N_{\mathrm{AP}} \times Z}$ without additional reformatting. Key parameters are aligned, including AP count, PRB configuration, and carrier frequency (e.g., $N_{\mathrm{AP}}{=}8$, $N_{\mathrm{PRB}}{=}100$, $f{=}2.61\,\mathrm{GHz}$), with periodic reporting every $0.1\,\mathrm{s}$.

xApp interfaces and runtime: The system exposes three main interfaces: (i) a REST API for lifecycle management (`/register`, `/start`, `/stop`, `/config`), (ii) a WebSocket stream for real-time telemetry, and (iii) a lightweight dashboard (Fig. 5) for visualization and interaction. Internally, an asynchronous pipeline decouples ingestion, decoding, prediction, and publishing stages.

IV. KEY RESULTS

The demo provides live visibility into system behavior through three metrics: (i) end-to-end latency, (ii) real-time prediction outputs, and (iii) integrity counters with fallback activations. These metrics are continuously updated in the dashboard during the demonstration.

Table I summarizes RX power predictions across APs. The model closely tracks measured values, maintaining consistency across different signal levels. AP2 shows the largest deviation, slightly underestimating peak power, while AP4 predicts a non-zero value despite a zero measurement. These discrepancies stem from the temporal model's smoothing and reduced sensitivity at low-power levels, and remain within acceptable bounds for short-horizon forecasting under compressed inputs.

The system also includes real-time issue detection and mitigation. Fig. 6 shows an xApp alert for a degraded KPM report, displaying the report ID (e.g., 300), flagging the issue (missing RX power), and indicating the applied fallback (*previous measurement*). These events are surfaced in the dashboard, allowing to monitor integrity alongside prediction outputs.

In practice, the processing loop runs continuously under Near-RT constraints, achieving an end-to-end latency of $\sim$50 ms on a GPU-enabled node. When inputs are missing or delayed, the xApp applies the configured fallback (previous measurement or last prediction) and logs the event in the UI, enabling continuous monitoring of latency and integrity metrics, with per-AP prediction accuracy consistent with Table I.

V. CONCLUSION

We presented LITE, a lightweight trajectory-unaware CSI estimation xApp integrated within a CF-MaMIMO emulator. The demo validates real-time operation under Near-RT constraints, combining compressed CSI transport with accurate short-horizon prediction. Interactive visualization and fault-handling mechanisms highlight robustness and practical deployability, demonstrating the feasibility of embedding lightweight intelligence into O-RAN-compliant RAN loops.

ACKNOWLEDGMENT

This research is funded by the imec.icon project RAPIDNESS, which is co-financed by imec and Flanders Innovation and Entrepreneurship under project nr HBC.2024.0772

REFERENCES

[1] D. Goez, M. Piazzola, G. Costa, A. Colpaert, R. Martinez Alonso, E. Aycan Beyazıt *et al.*, "Lite: Lightweight channel gain estimation with reduced x-haul csi signaling in o-ran," in *2026 32nd International Conference on Telecommunications (ICT)*, Thessaloniki, Greece, 2026.

[2] 6G-BRICKS Consortium, "D3.1: Initial report on communication and sensing algorithms, cell-free distributed computation and synchronization," Smart Networks and Services Joint Undertaking (SNS JU), Horizon Europe, Project Deliverable D3.1, Jun. 2024, accessed: 2026-03-28. [Online]. Available: https://6g-bricks.eu/wp-content/uploads/2024/06/6G-BRICKS-D-3.1-Initial-report-on-communication-and-sensing-algorithms-cell-free-distributed-computation-and-synchronization-Final-CLEAN_20230404.pdf

[3] R. Martinez Alonso, R. Beerten, A. Colpaert, A. P. Guevara, and S. Pollin, "Trajectory-unaware channel gain forecast in a distributed massive mimo system based on a multivariate bilstm model," *IEEE Open Journal of the Communications Society*, vol. 5, pp. 5348–5363, 2024.