

# Intelligent Traffic Steering in O RAN: A Deep Proactive Throughput and Resource Co-optimization

Mrityunjay Gain<sup>\*</sup>, Avi Deb Raha<sup>\*\*</sup>, Gikum Fitihamlak Ejigu<sup>\*\*</sup>, Dong Uk Kim<sup>\*\*</sup>, and Choong Seon Hong<sup>\*\*</sup>

<sup>\*</sup>*Department of Artificial Intelligence, Kyung Hee University, Yongin-si, South Korea*

<sup>\*\*</sup>*Department of Computer Science and Engineering, Kyung Hee University, Yongin-si, South Korea*

gain@khu.ac.kr, avi@khu.ac.kr, girumfitex@khu.ac.kr, g9896@khu.ac.kr, cshong@khu.ac.kr

**Abstract**—Future cellular networks are moving toward disaggregated, cloud-native, and programmable architectures, with open radio access network (O-RAN) emerging as a key enabler. O-RAN introduces intelligent control and data-driven optimization into conventional RAN deployments. However, optimizing resource allocation alone is insufficient to meet the diverse and dynamic traffic demands. Traffic steering (TS), which intelligently guides user traffic to the most suitable radio resources, provides a more adaptive and performance-aware solution. In this paper, we present a proactive deep learning framework for traffic steering in O-RAN, using next-slot predictions of throughput, RB usage, and BLER. A BiLSTM model running in the non-real-time RAN intelligent controller (non-RT-RIC) forecasts these indicators to enable adaptive traffic allocation. We formulate a prediction-informed throughput resource co-optimization problem that maximizes throughput, enhances RB efficiency, and reduces BLER by intelligently distributing traffic across multiple RUs. We then propose a Gurobi-based linear programming to obtain a near-optimal allocation matrix. The proposed prediction and optimization framework is lightweight, scalable, and deployable within the non-RT-RIC. Experimental evaluations demonstrate that our proposed method significantly outperforms traditional equal-split and greedy baselines, achieving up to 13-62% higher throughput, 30-32% lower BLER, and 0.20-0.25% better RB utilization efficiency.

**Index Terms**—O-RAN, traffic steering, proactive, Gurobi, deep learning, non-RT-RIC, optimization.

## I. INTRODUCTION

The transition from 5G to 6G networks is set to enhance connectivity and expand the range of supported services. A key enabler of this evolution is the open radio access network (O-RAN), which promotes disaggregation, standardization, and openness to ensure vendor-neutral, interoperable environments [1], [2]. By decoupling hardware and software components and leveraging software-defined networking and virtualization, O-RAN enables scalable 5G deployment and real-time network optimization through its core components radio units (RUs), distributed units (DUs), and central units (CUs) [3]. AI and machine learning play a central role in O-RAN by enabling

intelligent resource management, such as beam-forming [4], CPU allocation [5] and traffic optimization across distributed base stations. However, managing heterogeneous services such as uRLLC, eMBB, and mMTC within a shared infrastructure presents significant challenges due to their conflicting latency, bandwidth, and reliability requirements [6]. RAN slicing helps address this by isolating resources across services, but dynamic and unpredictable traffic patterns demand more than static resource allocation. In this context, traffic steering (TS) emerges as a complementary approach that proactively guides user traffic toward the most suitable radio resources, thereby improving network adaptability, efficiency, and cost-effectiveness under evolving conditions.

The emergence of intelligent TS in O-RAN has led to several recent innovations aimed at optimizing resource utilization and user experience under dynamic network conditions. Lacava et al. [7] present a programmable ns-O-RAN-based TS framework that integrates convolutional neural networks (CNNs) with conservative Q-learning (CQL). Their approach enables per-user decision-making by learning optimal handover and TS strategies based on historical network states, significantly enhancing adaptability in real-time environments. Nguyen et al. [8] propose a holistic optimization framework that jointly addresses scheduling, congestion control, and flow-split distribution for multi-path TS. Their method leverages reinforcement learning to dynamically adjust traffic allocations, improving end-to-end flow performance under constrained resources. Tamim et al. [9] develop a machine learning-based system that proactively predicts congestion across network slices. By analyzing queuing delays and traffic patterns, their framework reroutes traffic to prevent bottlenecks and latency spikes, thereby improving quality of service (QoS). Kavehmadavani et al. [10] introduce the joint intelligent traffic prediction, flow-split distribution, dynamic user association, and radio resource management (JIFDR), which combines ML-based traffic forecasting with real-time dynamic resource provisioning. Their solution emphasizes predictive traffic awareness to reduce under- or over-provisioning across the RAN. Chatzistefanidis et al. [11] apply machine learning techniques to predict user-level QoS metrics for DU selection. Their model improves user experience by steering connections to optimal DUs based on predicted latency and throughput trends. Gain et al. [12] propose a deep learning-based TS framework that leverages channel quality estimation to forecast link

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government(MSIT) (No. RS-2023-00207816), (No. RS-2024-00352423), and supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government(MSIT) (No.2019-0-01287, Evolvable Deep Learning Model Generation Platform for Edge Computing) and supported by IITP- ITRC(Information Technology Research Center) grant funded by the Korea government(Ministry of Science and ICT)(IITP-2025-RS-2023-00258649) \*Dr. CS Hong is the corresponding author.

conditions and steer traffic accordingly. The model predicts key performance indicators and uses them for adaptive link selection in multi-connectivity environments.

While prior works have shown notable progress in optimizing handover policies, predicting congestion, and managing static traffic distribution, they often depend on indirect indicators such as queue lengths or operate under offline training assumptions. These approaches typically overlook the dynamic nature of physical-layer conditions and real-time adaptability. In contrast, our work directly forecasts next-slot physical-layer metrics namely throughput, resource block (RB) usage, and block error rate (BLER) using a BiLSTM-based deep learning model. These fine-grained predictions enable a forward-aware traffic steering optimization that adaptively allocates user traffic fractions across multiple RUs. By proactively anticipating these matrices, our predictive traffic steering framework improves throughput, enhances RB utilization, and reduces transmission errors under dynamic O-RAN scenarios. The main contributions of our paper are summarized as follows:

- We propose a deep learning-based proactive traffic steering framework for O-RAN architecture that enhances network throughput and efficiency by forecasting critical radio metrics such as RB usage, throughput, and BLER.
- To predict the next time slot metrics, we design a lightweight BiLSTM-based model within the non-RT RIC that accurately forecasts RB usage, throughput, and BLER across RUs. These forecasts are then used to optimize traffic steering decisions.
- We formulate a throughput-resource co-optimization problem that uses the predicted KPIs to dynamically allocate UE traffic fractions to the most suitable RUs under a multi-connectivity setup, and we propose Gurobi linear programming to solve it.
- Extensive experimental results demonstrate that our approach outperforms achieving up to 13-62% higher throughput, reducing BLER by 30-32%, and improving RB utilization efficiency by 0.20-0.25% compared to Greedy and Equal Split baselines.

The rest of the paper is organized as follows. In Section II, we illustrated the system model. Section III and IV represent the problem formulation and solution approach respectively. Section V presents the simulation settings and results, and finally, Section VI concludes the paper.

## II. SYSTEM MODEL

We consider a hierarchical O-RAN architecture comprising a single CU, denoted by  $C$ , connected to a set of  $k$  DUs,  $D = \{D_1, D_2, \dots, D_k\}$ . Each DU is connected to a set of  $m$  RUs, denoted by  $R = \{R_1, R_2, \dots, R_m\}$ , which serve a set of  $n$  UEs, denoted by  $U = \{U_1, U_2, \dots, U_n\}$ . The CU handles centralized intelligence and control, while the DUs provide latency-sensitive processing and act as intermediaries to the RUs. The system model is illustrated in Fig. 1 To describe the RU-UE connectivity, we define a binary association matrix  $\mathbf{A} \in \{0, 1\}^{m \times n}$ , where each element  $A_{j,i} = 1$  if UE  $U_i$  is associated with RU  $R_j$ , and  $A_{j,i} = 0$  otherwise. This matrix

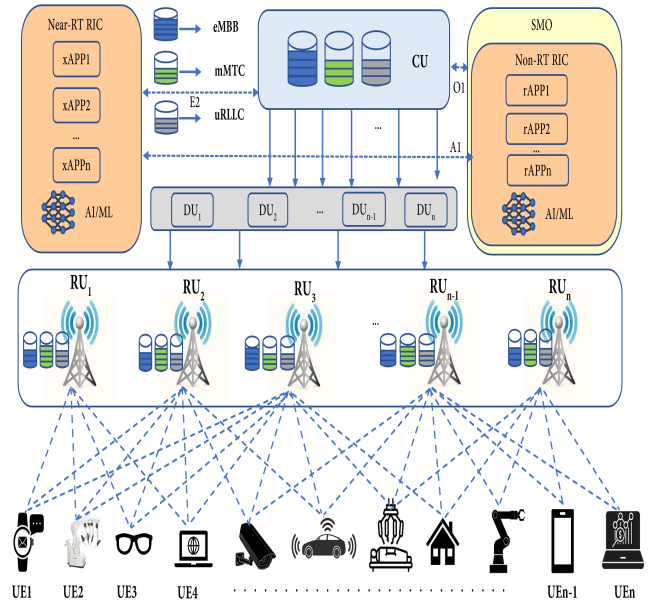


Fig. 1. System model for traffic steering in O-RAN systems. Each UE is connected to multiple RUs, and each RU is further connected to multiple DUs. Data from each UE adaptively selects the optimal RU-DU path to maximize throughput.

captures the partial connectivity between RUs and UEs. The set of RUs associated with UE  $U_i$  is denoted as:

$$\mathcal{R}_i = \{R_j \in R \mid A_{j,i} = 1\}, \quad (1)$$

and similarly, the set of UEs associated with RU  $R_j$  is defined as:

$$\mathcal{U}_j = \{U_i \in U \mid A_{j,i} = 1\}. \quad (2)$$

We define the UE-RU association as a binary matrix:

$$\mathbf{A} \in \{0, 1\}^{n \times m}, a_{i,j} = \begin{cases} 1 & \text{if UE } U_i \text{ is connected to RU } R_j, \\ 0 & \text{otherwise.} \end{cases} \quad (3)$$

The association matrix  $\mathbf{A}$  satisfies the following constraints:

$$\sum_{j=1}^m A_{j,i} \geq 1, \quad \forall U_i \in U, \quad (4)$$

which ensures that each UE is connected to at least one RU. This guarantees that every UE, regardless of the conditions, has a connection path to the network via an RU. This formulation allows for multi-connectivity, where each UE can be connected to more than one RU simultaneously, i.e.,  $|\mathcal{R}_i| \geq 1$  for each UE. Multi-connectivity is allowed, i.e.,  $|\mathcal{U}_j| \geq 1$ , enabling each UE to be simultaneously served by multiple RUs. The dynamic associations may vary over time, so we let  $\mathbf{A}(t)$  represent the RU-UE association matrix at timestep  $t \in \{1, 2, \dots, T\}$ . Each RU  $R_i$  have fixed RBs, obtain a throughput and also some block error. In every time slot  $t$  each RU have a vector containing these three.

$$\hat{y}_i(t+1) = [\hat{r}_i^{t+1}, \hat{\theta}_i^{t+1}, \hat{\beta}_i^{t+1}] \in \mathbb{R}^3, \quad (5)$$

where  $\hat{r}_i(t+1) \in \mathbb{R}_+$  is the RBs usage for  $R_i$  at time  $t+1$ ,  $\hat{\theta}_i(t+1) \in \mathbb{R}_+$  is the predicted throughput  $R_i$  at time  $t+1$ , and  $\hat{\beta}_i(t+1) \in [0, 1]$  is the predicted BLER for  $R_i$  at time  $t+1$ .

By accurately predicting these vectors for the upcoming time slot, scheduling of UE packets across RUs, can be dynamically adjusted thereby enabling improved average throughput, BLER, and more efficient RBs utilization across all RUs.

### III. PROBLEM FORMULATION

The objective is to allocate the traffic of each UE to RUs in such a way that the throughput is maximized the block error rate to ensure reliable communication and RBs is efficient without exceeding the RU's capacity. The optimization is formulated as:

$$\begin{aligned} \max_{w_{ij}} \quad & \sum_{j=1}^n \sum_{i=1}^m (\hat{\theta}_i^{t+1} - \lambda_1 \cdot \hat{\beta}_i^{t+1} - \lambda_2 \cdot \hat{r}_i^{t+1}) \cdot w_{ij} \\ \text{s. t.} \quad & \sum_{i=1}^m w_{ij} = 1 \quad \forall j \in n, \end{aligned} \quad (6a)$$

$$0 \leq w_{ij} \leq 1 \quad \forall i \in m, \forall j \in n, \quad (6b)$$

$$\sum_{j=1}^n w_{ij} \cdot \hat{r}_i^{t+1} \leq \alpha_i \quad \forall i \in m, \quad (6c)$$

$$\sum_{i=1}^m w_{ij} \cdot \hat{\beta}_i^{t+1} \leq \kappa \quad \forall j \in n, \quad (6d)$$

$$\sum_{i=1}^m w_{ij} \cdot \hat{\theta}_i^{t+1} \geq \gamma_j \quad \forall j \in n, \quad (6e)$$

where  $w_{ij}$  is the fraction of traffic allocated to RU  $i$  for UE  $j$ ,  $\hat{\theta}_i(t+1)$  is the predicted throughput for RU  $i$  at time  $t+1$ ,  $\hat{\beta}_i(t+1)$  is the predicted block BLER for RU  $i$  at time  $t+1$ ,  $\hat{r}_i(t+1)$  is the predicted resource block usage for RU  $i$  at time  $t+1$ ,  $\lambda_1$  and  $\lambda_2$  are the weight for the BLER and RBs usage respectively. Constraint (6a) and (6b) indicates each UE's traffic will be fully allocated to the available RUs, ensuring that the fraction of traffic for each UE for each RU will be within 0-1. Constraint (6c) indicates the total RBs used by each RU should be maximum within the capacity, ensuring efficient use of resources. To minimize the BLER for each UE's traffic allocation, we impose the constraint (6d). Constraint (6e) indicate the maximization of the throughput for each UE while distributing traffic based on the predicted throughput at each RU.  $\alpha_i$ ,  $\kappa$ , and  $\gamma_j$  are the resource block usage threshold, BLER threshold and throughput threshold respectively.

### IV. PROPOSED SOLUTION APPROACH

To address the proposed formulation problem, we introduce a BiLSTM-based prediction model to forecast throughput, BLER, and RBs usage for all RUs in the upcoming time slot. Leveraging these predicted metrics, we then employ a Gurobi-based linear programming optimization framework to maximize throughput, minimize BLER, and ensure efficient

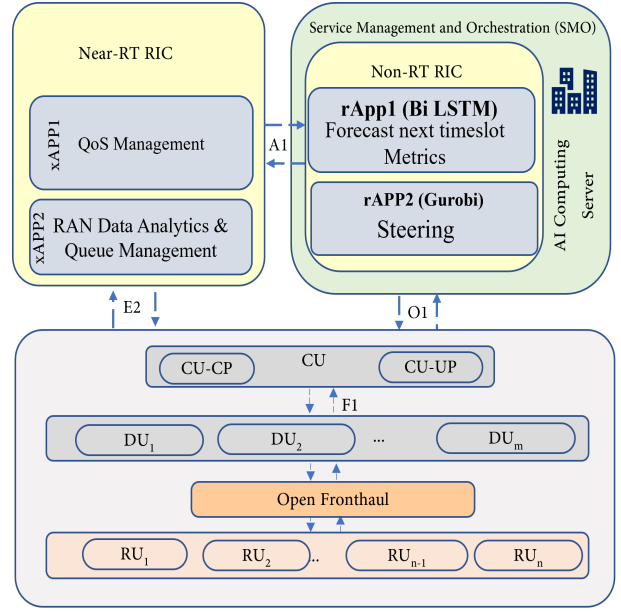


Fig. 2. Proposed deep learning oriented proactive traffic steering solution model for O-RAN. A BiLSTM in the non RT RIC predicts next slot performance metrics, and a Gurobi optimizer allocates UE traffic routes to maximize throughput.

RB utilization. The proposed solution model is illustrated in Fig. 2. Each step of the proposed approach is described in detail in the following sections.

#### A. Joint Resource Prediction via BiLSTM

For each RU  $R_i$ , we consider a temporal sequence of multivariate features over the past  $T$  timesteps. This feature matrix is denoted by:

$$\mathbf{X}_i = [\mathbf{x}_i(t-T+1), \mathbf{x}_i(t-T+2), \dots, \mathbf{x}_i(t)]^T \in \mathbb{R}^{T \times d}, \quad (7)$$

where  $\mathbf{x}_i(\tau) \in \mathbb{R}^d$  represents the feature vector at time  $\tau$  containing  $d$  attributes, such as historical RB usage, downlink throughput, BLER, SINR, CQI, and control signaling meta-data. To capture bidirectional temporal dependencies, we use a BiLSTM network  $f_\phi$  parameterized by  $\phi$ . The BiLSTM processes  $\mathbf{X}_i$  in both forward and backward directions:

$$\begin{aligned} \vec{\mathbf{h}}_\tau &= \text{LSTM}_{\text{fwd}}(\mathbf{x}_i(\tau), \vec{\mathbf{h}}_{\tau-1}), \\ \overleftarrow{\mathbf{h}}_\tau &= \text{LSTM}_{\text{bwd}}(\mathbf{x}_i(\tau), \overleftarrow{\mathbf{h}}_{\tau+1}). \end{aligned} \quad (8)$$

The hidden state at time  $\tau$  is computed by concatenation:

$$\mathbf{h}_\tau = [\vec{\mathbf{h}}_\tau; \overleftarrow{\mathbf{h}}_\tau] \in \mathbb{R}^{2h}, \quad (9)$$

where  $h$  is the dimensionality of each directional hidden state. The final context vector  $\mathbf{h}_i^*$ , used for prediction, is derived from a temporal aggregation of all hidden states. We use attention-based pooling:

$$\alpha_\tau = \frac{\exp(\mathbf{v}^T \tanh(\mathbf{W}_a \mathbf{h}_\tau))}{\sum_{\tau'=1}^T \exp(\mathbf{v}^T \tanh(\mathbf{W}_a \mathbf{h}_{\tau'}))}, \quad \mathbf{h}_i^* = \sum_{\tau=1}^T \alpha_\tau \cdot \mathbf{h}_\tau, \quad (10)$$

**Algorithm 1** Algorithm for intelligent proactive traffic steering in O RAN using BiLSTM based prediction aware throughput resource co optimization and Gurobi linear programming for solving the optimization problem.

**Require:** Time-series feature data  $\mathbf{X}_i \in \mathbb{R}^{T \times d}$  for each RU  $R_i$ , thresholds  $\kappa$ ,  $\gamma_j$ , and capacity limits  $\alpha_i$

**Ensure:** Optimal traffic allocation matrix  $[w_{ij}]$  for all RUs and UEs

- 1: **for** each RU  $R_i$ ,  $i = 1, 2, \dots, m$  **do**
- 2: Construct feature sequence  $\mathbf{X}_i = [\mathbf{x}_i(t - T + 1), \dots, \mathbf{x}_i(t)]^\top$
- 3: Compute BiLSTM hidden states by (8).
- 4: Concatenate hidden states:  $\mathbf{h}_\tau = [\mathbf{h}_\tau; \mathbf{h}_\tau]$
- 5: Apply attention pooling to get context vector by (10).
- 6: Predict next-slot resource metrics by (12).
- 7: **end for**
- 8: Construct utility coefficients for each  $(i, j)$  pair:

$$c_{ij} = \hat{\theta}_i^{t+1} - \lambda_1 \cdot \hat{\beta}_i^{t+1} - \lambda_2 \cdot \hat{r}_i^{t+1}.$$

- 9: Define decision variable vector  $\mathbf{w} = \text{vec}([w_{ij}]) \in \mathbb{R}^{mn}$ .
- 10: Formulate and solve the following Linear Program using Gurobi by (16).
- 11: **return** Optimal traffic allocation  $w_{ij}$  for all RU-UE pairs

where  $\mathbf{W}_a \in \mathbb{R}^{k \times 2h}$  and  $\mathbf{v} \in \mathbb{R}^k$  are trainable parameters of the attention mechanism. The prediction of the next-slot performance vector for RU  $R_i$  is obtained via a fully connected regression head:

$$\hat{\mathbf{y}}_i(t+1) = \mathbf{W}_p \cdot \mathbf{h}_i^* + \mathbf{b}_p = \begin{bmatrix} \hat{r}_i^{t+1} \\ \hat{\theta}_i^{t+1} \\ \hat{\beta}_i^{t+1} \end{bmatrix} \in \mathbb{R}^3, \quad (11)$$

where  $\hat{r}_i^{t+1}$  is the predicted RBs usage,  $\hat{\theta}_i^{t+1}$  is the predicted downlink throughput,  $\hat{\beta}_i^{t+1}$  predicted BLER, and  $\mathbf{W}_p \in \mathbb{R}^{3 \times 2h}$ ,  $\mathbf{b}_p \in \mathbb{R}^3$  are learnable output layer parameters.

The model is trained in a supervised fashion using the ground truth targets  $\mathbf{y}_i(t+1) \in \mathbb{R}^3$  from the real network traces. The loss function is defined as a weighted Mean Squared Error (MSE):

$$\mathcal{L}_{\text{pred}} = \sum_{i=1}^M \|\mathbf{W}_r \cdot (\hat{\mathbf{y}}_i(t+1) - \mathbf{y}_i(t+1))\|_2^2, \quad (12)$$

where  $\mathbf{W}_r = \text{diag}(\omega_r, \omega_\theta, \omega_\beta)$  controls the importance of RB usage, throughput, and BLER in the learning objective. This predictive module serves as the foundation for traffic steering decisions, enabling resource-aware and quality-optimized UE-RU assignments in real time.

### B. Prediction-Aware Traffic Steering via Optimization

After predicting the future RB usage  $\hat{r}_i(t+1)$ , throughput  $\hat{\theta}_i(t+1)$ , and BLER  $\hat{\beta}_i(t+1)$  for each RU  $R_i$  using the BiLSTM model, we formulate a linear program to steer the downlink traffic between RUs and UEs in an optimal manner.

Let  $w_{ij} \in [0, 1]$  denote the fraction of UE  $U_j$ 's traffic that is allocated to RU  $R_i$ . Our objective is to maximize average throughput, minimize the average BLER, maximize the average proper use of RBs across all RUs. Once the next timeslot's resource state  $(\hat{r}_i(t+1), \hat{\theta}_i(t+1), \hat{\beta}_i(t+1))$  is jointly predicted using the BiLSTM model (as described in Section IV-A), the traffic steering task is formulated as a linear programming (LP) problem to optimally distribute UE traffic across RUs while satisfying system constraints.

We define  $\mathbf{w} = [w_{ij}] \in \mathbb{R}^{m \times n}$  be the traffic allocation matrix where  $w_{ij} \in [0, 1]$  indicate the fraction of traffic of UE  $j \in n$  allocated to RU  $i \in m$ , and define the flattened decision vector:

$$\mathbf{w} = \text{vec}([w_{ij}]) \in \mathbb{R}^{m \cdot n}. \quad (13)$$

This vector is the optimization variable in the LP. We aim to maximize the total weighted utility of the system, which includes utility coefficient for each pair  $(i, j)$  as:

$$c_{ij} = \hat{\theta}_i^{t+1} - \lambda_1 \cdot \hat{\beta}_i^{t+1} - \lambda_2 \cdot \hat{r}_i^{t+1}. \quad (14)$$

Then, the overall objective becomes:

$$\max_{\mathbf{w}} \sum_{j=1}^n \sum_{i=1}^m c_{ij} w_{ij} = \mathbf{c}^\top \mathbf{w} \quad (15)$$

where  $\mathbf{c} \in \mathbb{R}^{m \times n}$  is the stacked vector of coefficients  $c_{ij}$ . We consider linear constraints. Each UE must receive 100% of its traffic  $\sum_{i=1}^m w_{ij} = 1$ ,  $\forall j \in n$ . The total RBs served by each RU must not exceed its predicted capacity  $\sum_{j=1}^n w_{ij} \cdot \hat{r}_i(t+1) \leq \alpha_i$ ,  $\forall i \in m$ . Each UE must receive traffic such that the aggregate BLER is within a tolerable range:  $\sum_{i=1}^m w_{ij} \cdot \hat{\beta}_i(t+1) \leq \kappa_{\text{th}}$ ,  $\forall j \in n$ . Each UE must receive sufficient throughput  $\sum_{i=1}^m w_{ij} \cdot \hat{\theta}_i(t+1) \geq \kappa_j$ ,  $\forall j \in n$  and fraction of traffic for each RU and UE will lie within range 0-1 ( $0 \leq w_{ij} \leq 1$ ,  $\forall i \in m, \forall j \in n$ ). We rewrite the entire problem as a standard LP:

$$\begin{aligned} \max_{\mathbf{w}} \quad & \mathbf{c}^\top \mathbf{w} \\ \text{s.t.} \quad & A_{\text{eq}} \cdot \mathbf{w} = \mathbf{b}_{\text{eq}}, \\ & A_{\text{ineq}} \cdot \mathbf{w} \leq \mathbf{b}_{\text{ineq}}, \\ & \mathbf{w} \geq \mathbf{0}, \end{aligned} \quad (16)$$

where  $A_{\text{eq}}$ ,  $A_{\text{ineq}}$ ,  $\mathbf{b}_{\text{eq}}$ , and  $\mathbf{b}_{\text{ineq}}$  encode the constraints above.

This LP can be solved using high-performance commercial solvers like Gurobi or CPLEX. In this work, we utilize Gurobi, which guarantees globally optimal solutions for convex problems and supports large-scale real-time optimization. This prediction-aware linear program is executed in each timestep with updated predictions from the BiLSTM, enabling real-time adaptation of UE-RU assignments for optimal network performance. The whole procedure is illustrated in Algorithm 1.

## V. SIMULATIONS SETTINGS AND RESULTS

### A. Dataset and Environment Settings

For our experiment, we utilize the O-RAN experimental evaluation dataset's [13] "energy" folder, available through

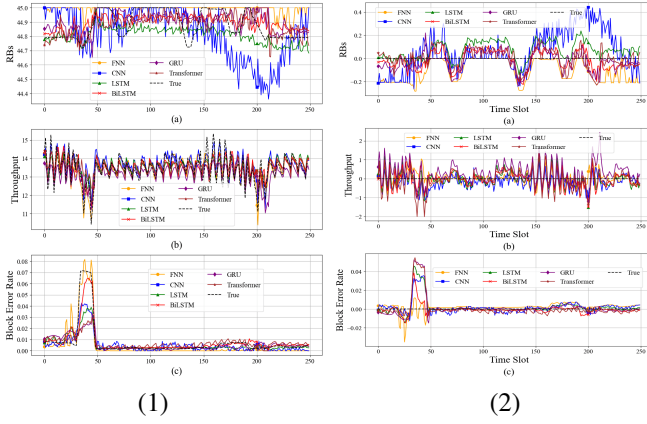


Fig. 3. Visualization of predicted RB usage, throughput, and BLER versus ground truth and baselines. Subfigure (1) shows predictions across models, while Subfigure (2) presents their corresponding prediction errors.

the IEEE Dataport. The dataset includes two key setups: ‘ul’ and ‘dlul’, where we focus on the ‘dataset\_dlul’ configuration, which provides essential traffic and system performance metrics. This specific setup contains a total of 32,798 traffic samples, which are split into 80% for training and 20% for testing to ensure robust evaluation. For model development and training, we utilize a computational environment built on Python 3.10, with PyTorch serving as the deep learning framework. The model optimization is performed using the Adam optimizer with a learning rate of 0.001. Our network architecture consists of a 2-layer BiLSTM with two hidden layer size of 128 units. During training, we set the batch size to 32 and employ  $\lambda_1$  and  $\lambda_2$  value of 0.1 and 0.05 respectively. We set the constraint threshold  $\gamma_j$ ,  $\alpha_i$ , and  $\kappa$  value of 10, 45, and 0.01 respectively to ensure high network performance. We consider 3 RUs and 5 UEs for our experiment. every UEs is connected to every RUs. For evaluating the steering performance we consider two approach ‘greedy’ and ‘equal split’.

**Greedy Approach:** The Greedy approach allocates each UE’s traffic entirely to the RU with the highest predicted throughput. This method aims to maximize data transmission efficiency by prioritizing RUs that are expected to deliver the best performance. It is straightforward and effective in scenarios where maximizing instantaneous throughput is the primary objective.

**Equal Split Approach:** The Equal Split strategy distributes each UE’s traffic equally among all its connected RUs. This simple approach ensures fairness and avoids overloading any particular RU. However, it fails to consider dynamic network conditions like varying throughput, RB availability, and BLER, which can lead to suboptimal performance in terms of both efficiency and reliability.

## B. Experimental Results

In this section, we evaluate the performance of our proposed method against several baseline models. For next time slot prediction of throughput, BLER, and required RBs, we com-

TABLE I  
COMPARISON OF FORECASTING PERFORMANCE BETWEEN THE PROPOSED BiLSTM MODEL AND BASELINE MODELS.

| Model       | Train MSE     | Parameters  | Test MSE    | Test MAE    |
|-------------|---------------|-------------|-------------|-------------|
| FNN         | 0.0002        | 2230275     | 0.05        | 0.14        |
| CNN         | 0.0002        | <b>8628</b> | 0.05        | 0.12        |
| LSTM        | 0.0002        | 20653       | 0.13        | 0.08        |
| BiLSTM      | <b>0.0001</b> | 25753       | <b>0.01</b> | <b>0.05</b> |
| GRU         | 0.0003        | 15528       | 0.12        | 0.15        |
| Transformer | 0.0003        | 68739       | 0.07        | 0.12        |

pare our BiLSTM based forecasting model with other widely used deep learning architectures including feedforward neural network (FFN), one dimensional CNN, standard LSTM, gated recurrent unit (GRU), and transformer.

In Tab. I, we present the comprehensive results of train mean squared error (MSE) loss, number of parameters, test MSE and test mean absolute error (MAE) for our proposed method compared to the five baselines. From the table we find among all the models, the BiLSTM achieves the best overall performance, recording the lowest train MSE (0.0001), lowest test MSE (0.10), and lowest test MAE (0.05). While the CNN offers the smallest parameter footprint (8628), it lags behind BiLSTM in terms of predictive accuracy. Although models like LSTM and Transformer exhibit strong individual traits such as the Transformer’s relatively low test MSE (0.07) they either suffer from higher MAE or require significantly more parameters. The BiLSTM strikes the optimal balance between accuracy and model complexity, making it the most suitable choice for the prediction-aware traffic steering framework in O-RAN scenarios.

In Fig. 3, we present a visual comparison between the predicted and ground truth values of throughput, RBs usage, and BLER, along with their corresponding prediction errors. The left side of the figure displays the predicted values alongside the actual ground truth, while the right side depicts the prediction errors. For both sides, subfigures (a), (b), and (c) represent RBs, throughput, and BLER, respectively. As shown in Fig. 3(1), the proposed Bi-LSTM model accurately predicts the required RBs, throughput, and BLER for the next time slot, closely aligning with the ground truth and outperforming other baseline models. In Fig. 3(2), the BiLSTM model consistently outperforms the other models across all three metrics. For RB prediction (a), BiLSTM closely follows the true trend after the initial time steps, capturing key fluctuations with relatively small errors. In throughput prediction (b), it achieves high accuracy with errors near zero, aside from a few minor spikes that remain negligible compared to the other methods. For BLER prediction (c), BiLSTM maintains near-zero error for most of the time, closely tracking the ground truth. Although a noticeable spike appears around time steps 40–50, its overall performance remains superior, highlighting its effectiveness over baseline models.

In Fig. 4, we evaluate the traffic steering performance of the proposed method against two baseline strategies, Equal Split and Greedy, across multiple time slots, using throughput, RB

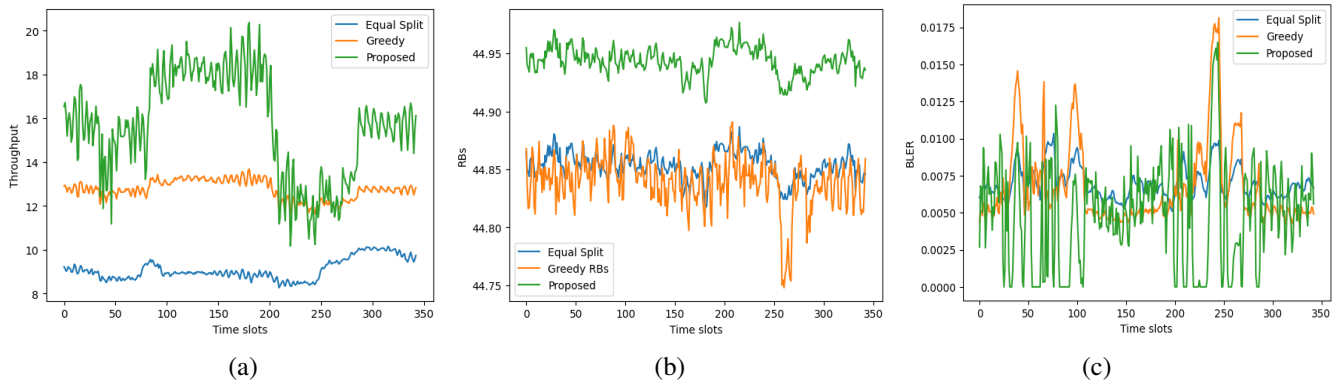


Fig. 4. Performance comparison in terms of average throughput, resource block usage, and BLER between the proposed method and baseline approaches. Higher throughput and RB utilization denote better performance, while lower BLER indicates more efficient network management.

utilization, and BLER as performance metrics. As shown in Fig. 4(a), the proposed method consistently achieves higher throughput than both baselines. Equal Split delivers low but stable performance, while Greedy shows moderate throughput with notable fluctuations. In Fig. 4(b), our method demonstrates more efficient RB utilization, consistently allocating maximum usable resources than the baselines. Equal Split maintains a uniform allocation, and Greedy exhibits variable usage. Fig. 4(c) shows that our method achieves the lowest BLER, frequently reaching zero but with occasional spikes, indicating a highly adaptive though sometimes volatile response to changing conditions. Overall, the proposed strategy outperforms both Equal Split and Greedy across all key metrics.

## VI. CONCLUSION

In this study, we propose a deep learning-based intelligent proactive traffic steering framework tailored for O-RAN environments. The framework formulates a throughput-resource co-optimization problem aimed at enhancing network efficiency by leveraging predicted required RBs, throughput, and BLER for each RU in the upcoming time slot. To achieve the forecasting outcome, a BiLSTM-based forecasting model is deployed in the non-real-time RIC to estimate key radio metrics proactively. These predictions are then used by the optimization module, which employs linear programming via Gurobi to dynamically allocate user traffic to RUs. The objective is to maximize throughput and resource utilization while minimizing the BLER. The experimental evaluation, conducted using real O-RAN traffic traces, demonstrates that the proposed approach achieves approximately up to 13% and 62% higher throughput compared to the Greedy and Equal Split baselines, respectively. Additionally, it reduces the BLER by approximately 30–32%, and improves RBs utilization efficiency by 0.20–0.25% over these baselines. These results highlight the effectiveness of our framework in delivering adaptive, efficient, and reliable traffic management for next-generation disaggregated RAN systems.

## REFERENCES

- [1] M. Gain, A. D. Raha, A. Adhikary, S. S. Hassan, and C. S. Hong, "Fortifying lifelong security for o-ran ecosystem: An incremental learning framework for nextg seamless networking," in *International Conference on Information Networking (ICOIN)*, Chiang Mai, Thailand, 2025, pp. 396–401.
- [2] M. Gain, A. D. Raha, A. Adhikary, K. Kim, and C. S. Hong, "Open ran embracing continual learning: Towards nextg adaptive traffic analysis," in *IEEE Network Operations and Management Symposium (NOMS)*, Seoul, Korea, May 2024.
- [3] M. Gain, A. D. Raha, S. K. Dam, A. Amirjon, K. Kim, and C. S. Hong, "Ai-driven proactive handover optimization for nextg o-ran systems," in *25th Asia-Pacific Network Operations and Management Symposium (APNOMS)*, Kaohsiung, Taiwan, 2025.
- [4] A. D. Raha, M. Gain, H. Kim, Y. Qiao, Z. Han, E.-N. Huh, and C. S. Hong, "Adversarial attacks and defenses for vision-based mmwave beam prediction in 6g uav networks," *Authorea Preprints*, 2025.
- [5] M. Gain, R. Avi Deb, A. Adhikary, H. Zhu, and C. S. Hong, "Continual adaptation and dynamic number of devices management for resource provisioning in nextg o-ran," in *IEEE International Conference on Communications (ICC)*, Montreal, Canada, Jun. 2025.
- [6] M. Gain, K. Kim *et al.*, "A deep incremental framework for multi-service multi-modal devices in nextg ai-ran systems," *arXiv preprint arXiv:2504.01730*, 2025.
- [7] A. Lacava, M. Polese, R. Sivaraj, R. Soundararajan, B. S. Bhati, T. Singh, T. Zugno, F. Cuomo, and T. Melodia, "Programmable and customized intelligence for traffic steering in 5g networks using open ran architectures," *IEEE Transactions on Mobile Computing*, vol. 23, no. 4, pp. 2882–2897, Apr. 2023.
- [8] V.-D. Nguyen, T. X. Vu, N. T. Nguyen, D. C. Nguyen, M. Juntti, N. C. Luong, D. T. Hoang, D. N. Nguyen, and S. Chatzinotas, "Network-aided intelligent traffic steering in 6g o-ran: A multi-layer optimization framework," *IEEE Journal on Selected Areas in Communications*, vol. 42, no. 2, pp. 389–405, Nov. 2024.
- [9] I. Tamim, S. Aleyadeh, and A. Shami, "Intelligent o-ran traffic steering for urllc through deep reinforcement learning," in *Proceedings of the IEEE International Conference on Communications (ICC)*, Rome, Italy, Jun. 2023, pp. 112–118.
- [10] F. Kavehmadavani, V.-D. Nguyen, T. X. Vu, and S. Chatzinotas, "Intelligent traffic steering in beyond 5g open ran based on lstm traffic prediction," *IEEE Transactions on Wireless Communications*, vol. 22, no. 11, pp. 7727–7742, Mar. 2023.
- [11] I. Chatzistefanidis, N. Makris, V. Passas, and T. Korakis, "MI-based traffic steering for heterogeneous ultra-dense beyond-5g networks," in *Wireless Communications and Networking Conference (WCNC)*, Scotland, UK, Mar. 2023.
- [12] M. Gain, A. D. Raha, A. Adhikary, H. Kim, A. Amirjon, and C. S. Hong, "Optimal corridor selection: An intelligent traffic steering framework for nextg o-ran based on channel quality estimation," in *Korea Software Congress (KSC)*, Yeosu-Expo, Korea, Dec. 2024, pp. 1052–1054.
- [13] J. X. Salvat Lozano, J. A. Ayala-Romero, L. Zanzi, A. Garcia-Saavedra, and X. Costa-Perez, "O-ran experimental evaluation datasets," *IEEE Dataport*, 2022.