**Communication in *Physical Sciences*, 2023, 9(4): 1119-1133**

# Machine Learning-Based Predictive Congestion Management and Dynamic Resource Allocation in 5G Networks.

**Moses Oluwasegun Odewale[1], Moses Olagoke Odejobi[2], Olanrewaju Oluwaseun Ajayi[3].**
**Received: 22 May 2023/Accepted: 10 September 2023/Published: 19 September 2023**

***Abstract***: *Congestion in fifth-generation (5G) wireless networks has evolved from a simple bandwidth limitation problem into a complex interaction among ultra-dense deployments, heterogeneous service classes, and highly dynamic user mobility patterns. Conventional reactive congestion management techniques, including threshold-based admission control and static scheduling policies, are inadequate for handling the millisecond-level resource allocation dynamics required in 5G networks. This study proposes the Predictive Congestion and Resource Orchestration System (PACROS), a machine learning-driven framework that integrates short-horizon traffic forecasting, uncertainty-aware resource pre-allocation, and reinforcement learning-based closed-loop remediation within a unified congestion management architecture. PACROS employs a temporal convolutional network (TCN) enhanced with multi-head attention to predict per-cell traffic load over a 500-ms forecasting horizon, while a constrained optimization module proactively allocates physical resource blocks (PRBs), modulation and coding scheme (MCS) levels, and buffer admission thresholds ahead of predicted demand surges. A proximal policy optimization (PPO)-based remediation agent dynamically adjusts network resources in real time to mitigate residual congestion caused by forecasting uncertainty. The framework was evaluated using an ns-3/5GLENA-based 5G Non-Standalone simulation environment comprising 36 gNodeBs, 900 user equipment nodes, and urban mobility-driven traffic traces with injected anomaly scenarios. Results show that PACROS reduced mean buffer occupancy by 39.4%, decreased 95th-percentile packet delay by 44.7%, lowered URLLC service-level agreement (SLA) violation rates by 58.8%, improved PRB utilization efficiency by 22.3%, and reduced handover-triggered service interruptions by 31.8% compared with a 3GPP-compliant proportional fair baseline scheduler. PACROS also achieved an in-deadline URLLC packet delivery rate of 96.1%, outperforming all baseline methods during both steady-state and anomalous traffic conditions. These findings demonstrate the effectiveness of predictive, uncertainty-aware, and closed-loop congestion management as a practical foundation for autonomous and SLA-aware 5G network operation.*

***Keywords:*** *predictive control; congestion; 5G NR; convolutional networks; traffic forecasting.*

**Moses Oluwasegun Odewale**
Faculty of Information Technology and Communication Sciences, Tampere University, Tampere, Finland.
**Email:** segun.odewale@gmail.com

**Moses Olagoke Odejobi**
Department of Electrical and Computer Engineering, Morgan State University, Baltimore, Maryland, U.S.A.
**Email:** moses.o.odejobi@gmail.com

**Olanrewaju Oluwaseun Ajayi**
Department of Information Technology, University of the Cumberlands, Williamsburg, Kentucky, U.S.A.
**Email:** oajayi77648@ucumberlands.edu

## 1.0 Introduction

The evolution of telecommunications infrastructure has consistently demonstrated that actual traffic growth frequently exceeds initial network design assumptions. Third-generation networks were dimensioned around voice and low-rate data, yet within a few years of commercial launch they were being overwhelmed by mobile internet traffic that the original capacity planning exercises had not

meaningfully anticipated. Fourth-generation LTE made more deliberate provisions for data, but the rapid growth of video streaming services consistently exceeded the forecasting assumptions embedded within resource management algorithms." Fifth Generation (5G) networks support heterogeneous service classes, including enhanced mobile broadband (eMBB), ultra-reliable low-latency communications (URLLC), and massive machine-type communications (mMTC), each characterized by distinct traffic patterns and quality-of-service requirements. (IMT-2020, 2015). And yet, the algorithms used for resource management, which determine how 5G radio access networks respond to congestion, are – in most operational deployments, remain predominantly dependent on reactive threshold-driven scheduling and heuristic congestion mitigation mechanisms that were designed in a world where the relevant time scales were on the order of tens of milliseconds, not sub-millisecond (Andrews *et al*., 2014).

"Congestion in 5G networks extends beyond spectrum scarcity and computational limitations to include complex coordination and state-awareness challenges : ; it is an information and coordination problem. A gNodeB handling hundreds of parallel UE connections, multiple component carriers (CCs), numerologies and service classes has a state space of staggering complexity.Although aggregate network capacity is generally sufficient to satisfy average demand, temporary mismatches between traffic arrival and resource allocation frequently occur due to bursty video traffic, mobility-induced spatial clustering, and the coexistence of delay-sensitive and delay-tolerant services. These mismatches can trigger buffer overflows, packet loss, and retransmission storms that further degrade network performance (Benjebbour *et al*., 2015). The conventional response is to wait until congestion has occurred (as measured by some threshold on buffer occupancy or delay), then apply admission control or packet discards, and wait for the congestion to abate. But in a 5G network operating at the limit of its capacity, the delay between the onset of congestion and the time that reactive actions can be taken may be sufficient to impair service quality of delay-sensitive sessions.

Predictive congestion management has emerged as a promising alternative to purely reactive congestion control strategies. Its origins can be found in the traffic engineering research of the 1990s that was focused on managing flows (Kelly *et al*., 1998) and in pre-reservation-based resource allocation in ATM networks (Fernandez, 1993). What makes it feasible now, however, are the machine learning techniques that can model the highly nonlinear, spatio-temporal and seasonal structure in mobile network traffic data to the degree that predictive resource allocation can be made practically deployable in operational environments (Zhang *et al*., 2019). Neural sequence models – especially long short-term memory (LSTM) networks and, more recently, temporal convolutional networks (TCN) with attention – have shown impressive performance on cellular traffic prediction benchmark tasks, with orders of magnitude improvement over classical autoregressive time-series methods, and even better performance than bespoke traffic engineering heuristics on held-out real-world data (Bai *et al*., 2018; He *et al*., 2021).

The combination of traffic prediction and closed-loop resource allocation, however, has proved challenging.

The majority of published research treats the traffic prediction and the resource management problems as decoupled: train a predictive model offline, process the model's predictions as inputs to a resource management system, and assess the end-to-end system in the hope that the prediction error is low and can be ignored. Consequently, predictive uncertainty, adaptive corrective feedback, and real-time closed-loop orchestration remain insufficiently addressed in practical 5G congestion management systems. This assumption becomes invalid under highly dynamic network conditions characterized by traffic surges, mobility-

induced clustering, and rare events. A predictive congestion framework that is robust should not only produce good average predictions; it should also close the loop between predictive uncertainty and control, so that when the predictor is wrong (as prediction inaccuracies inevitably occur in real-world deployments ), the control "corrects" its actions accordingly instead of amplifying the predictions' errors (Mao *et al*., 2017).

To address these limitations, this study proposes the Predictive Congestion and Resource Orchestration System (PACROS), which conceives of traffic forecasting, resource pre-allocation, and reactive remediation as three interlinked layers of a closed-loop system rather than as off-the-shelf components to be sequenced offline. The study aims to develop a closed-loop machine learning framework capable of proactive congestion prediction, uncertainty-aware resource allocation, and adaptive real-time remediation in 5G networks.

The forecasting layer employs a TCN with multi-head self-attention mechanism, which leverages the TCN's dilated causal convolutions to capture short-term temporal information and the attention mechanism to capture longer-term dependencies, to produce predictive traffic load estimates down to the per-cell level with a 500-millisecond look-ahead window. The pre-allocation layer considers a constrained optimization problem to transform the forecast results and estimates of their uncertainty into dynamic pre-allocation of network resources , taking into account the bounds on the forecast errors in the constraints of the problem so that conservative over-provisioning takes place when the forecast is uncertain. The remediation layer is a proximal policy optimization (PPO) agent continuously monitors residual resource mismatch between the pre-allocation and the demand in real time, and issues corrective actions (dynamic bandwidth part (BWP) switching, modulation and coding scheme (MCS) adaptation, and inter-cell load migration) in the 1-millisecond transmission time interval (TTI) budget.

Beyond its algorithmic contributions, this work also addresses practical deployment considerations within O-RAN architectures. with its modular design and standardized interfaces at the near-real-time and non-real-time RAN intelligent controller (RIC), is opening up an opportunity to deploy ML-based control algorithms in commercial 5G networks without modifying the radio hardware or RAN software (O-RAN Alliance, 2021). PACROS is expressly designed to be deployed as an xApp on the O-RAN near-real-time RIC (near-RT RIC), with its three constituent layers corresponding to the E2, A1, and O1 interfaces, respectively. The proposed framework contributes toward the realization of autonomous and self-optimizing 5G networks capable of intelligent congestion prevention and efficient spectrum utilization. The details of this deployment are outlined in the methods section and are an innovation as much in system design as in algorithm design.

### *1.1 Related Work*

The literature underpinning PACROS can be broadly categorized into three interconnected research domains, each contributing methods that are integrated and extended in the present study.

#### *1.1.1 Traffic prediction in cellular networks*

Mobile network traffic exhibits complex spatio-temporal characteristics, including strong daily and weekly periodicities associated with human activity patterns, as well as short-term stochastic fluctuations resulting from user behaviour and application-layer dynamics."

(Furno *et al*., 2017). "Early traffic prediction studies primarily employed autoregressive integrated moving average (ARIMA) and seasonal ARIMA models, which effectively captured temporal periodicities but were limited in modeling nonlinear spatial-temporal interactions within cellular environments. Deep learning approaches subsequently improved prediction performance : : convolutional neural networks

(CNNs) for spatial traffic maps (Zhang *et al*., 2017), LSTM models (Huang *et al*., 2018) for per-cell sequence prediction, and the combination of CNN-LSTM models to leverage spatial and temporal information. More recently, temporal convolutional networks (TCNs) have emerged as efficient alternatives to recurrent architectures for sequence modeling for sequence prediction, with parallel training, stable gradients, and competitive accuracy at lower inference latency (Bai *et al*., 2018). "These characteristics are particularly important in real-time systems where prediction outputs directly influence low-latency resource allocation decisions.

***1.1.2 Proactive and predictive resource management***.

Traffic-aware proactive resource management strategies have been widely investigated, although most existing approaches do not employ fully closed-loop architectures has been investigated in a number of scenarios, albeit not always in closed-loop. Jiang *et al*. (2016) developed predictive triggers for handover to avoid unnecessary handovers by 27% in simulated traces. Sun *et al*. (2018) showed how traffic intensities predicted using long short-term memory (LSTM) can increase resource block pre-allocation efficiency in a heterogeneous network by 19% compared to reactive scheduling.He et al. (2021) further employed a transformer-based predictor to feed a model predictive control (MPC) scheduler in a simulated 5G network to reduce latency for URLLC. A major limitation of these studies is the assumption of consistently accurate traffic forecasts during resource allocation and scheduling operations. Furthermore, most existing approaches do not explicitly account for forecast uncertainty or incorporate uncertainty-aware resource allocation mechanisms.

***1.1.3 Reinforcement learning for congestion control and scheduling***

Reinforcement learning (RL) has demonstrated significant potential in wireless scheduling and congestion control applications. The ORCA system (Mao *et al*., 2017) was originally designed for video streaming rate adaptation, demonstrating that RL-based policies can outperform manually engineered heuristics in dynamic network environments in dynamically changing environments. Applications of RL within cellular radio resource management (RRM) include multi-agent RL for downlink scheduling (Naparstek & Cohen, 2019) and DQN-based admission control (Li *et al*., 2018). The integration of predictive feedforward control with RL-based feedback correction, which forms the basis of PACROS, has previously been explored in robotics and industrial process control systems (Nagabandi *et al*., 2018). However, its application to 5G congestion management within a unified framework that simultaneously integrates forecasting, proactive allocation, and adaptive remediation remains largely unexplored.

## 2.0 Methods

### *2.1 System Model and Problem Formulation*

Consider a 5G NR downlink network comprising NCN_CNC gNodeBs, each serving a time-varying population of UEs drawn from a mixture of eMBB, URLLC, and mMTC traffic classes. Let $\rho_c(t) \in [0,1]$ denote the normalized traffic load of cell $c$ at discrete time slot $t$, where ttt denotes discrete 1-ms transmission time intervals (TTIs). A congestion event is declared at cell $c$ when $\rho_c(t) > \rho_{\text{th}}$ or at least $\tau_{\text{hold}}$ consecutive TTIs , where $\rho_{\text{th}}$ is a configurable threshold and $\tau_{\text{hold}} = 5$ slots in our experiments.

"The aggregate congestion cost over a planning horizon of HHH TTIs is defined as:

$$\mathcal{C} = \sum_{t=1}^{H} \sum_{c=1}^{N_C} \left[ w_1 \cdot 1[\rho_c(t) > \rho_{\text{th}}] \cdot D_c(t) + w_2 \cdot \max(0, B_c(t) - B_{\max}) + w_3 \cdot 1\left[\text{drop}_c(t)\right] \right] \quad (1)$$

where $D_c(t)$ is the mean packet sojourn time in cell $c$'s buffer, $B_c(t)$ is the instantaneous buffer occupancy (in packets), $B_{\max}$ is the buffer size limit, $\mathbf{1}[\text{drop}_c(t)]$ is an indicator for packet drop events, and $\{w_i\}$ are operator-configurable penalty weights reflecting service priority. "The primary optimization objective of PACROS is to minimize the cumulative congestion cost CCC."

The decision variables available to the system at each TTI include: the physical resource block (PRB) allocation matrix $\mathbf{X}(t) \in \{0,1\}^{NC \times NPRB}$, the MCS index vector $\mathbf{m}(t) \in \{0,\ldots,28\}^{NC}$, the buffer admission threshold vector $\beta(t)$, and a binary load migration indicator $\mathbf{v}(t)$ that specifies inter-cell traffic steering decisions. The optimization variables are constrained by maximum transmit power limits, inter-cell interference relationships derived from the SINR model in Equation (2), and proportional-fair scheduling requirements specified by 3GPP standards.

$$\gamma_{u,c,k}(t) = \frac{P_{c,k}.(t).G_{u,c,k}(t)}{\sigma^2 \sum_{c' \neq c} P_{c,k}.(t).G_{u,c,k}(t).1[c' \in I_c]} \quad (2)$$

where $I_c$ denotes the set of neighboring cells whose transmissions interfere with cell ccc. c (a set that depends on the deployment geometry, and changes as load migration decisions alter the set of active transmitters).

### *2.2 PACROS Architecture*

The PACROS framework is organized into three functional layers operating at nested time scales, as depicted in *Fig.* 1. The forecasting and pre-allocation layers execute at 100-ms intervals, whereas the remediation layer operates continuously at the 1-ms TTI level.

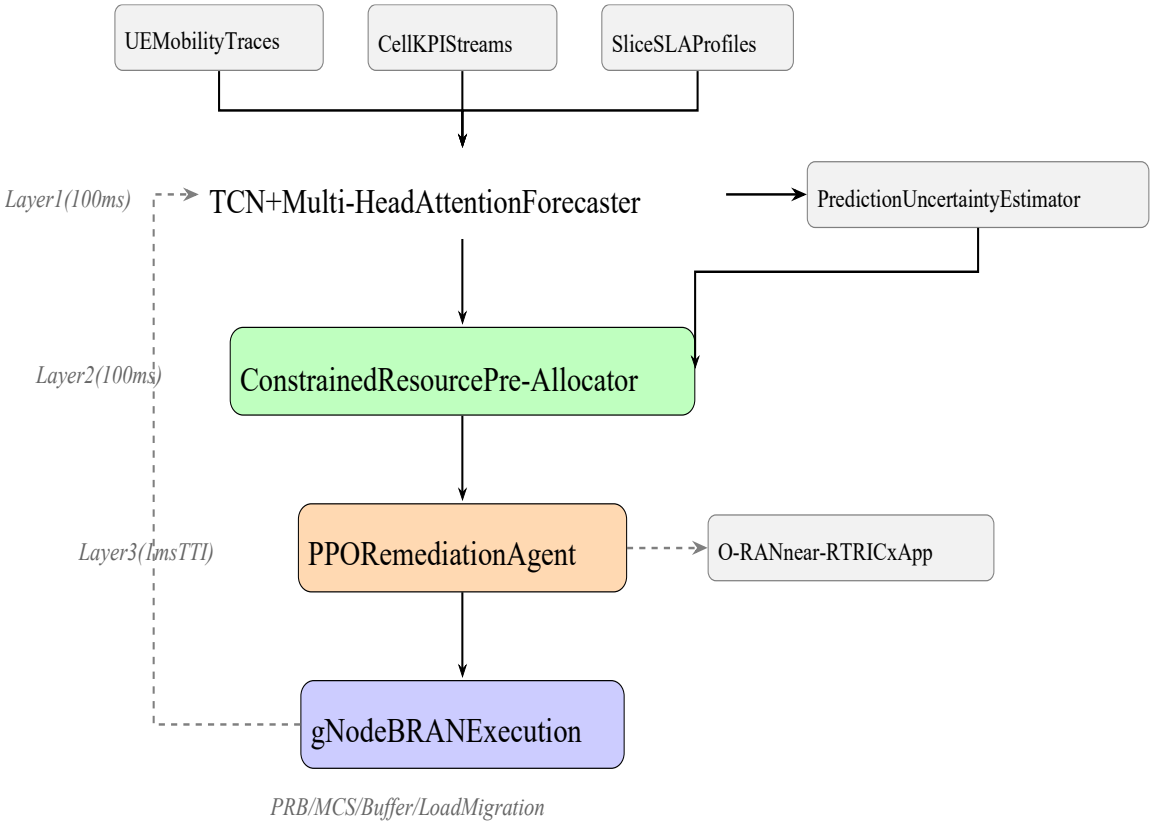


***Fig.* 1: PACROS three-layer architecture. Layer 1 (forecasting) generates probabilistic per-cell traffic load predictions every 100 ms using a TCN with multi-head attention, with a companion uncertainty estimator that quantifies prediction confidence. Layer 2 (pre-allocation) translates forecasts and uncertainty bounds into concrete radio resource positioning decisions via constrained optimization. "Layer 3 employs a PPO-based reinforcement learning agent to correct residual congestion at the TTI level.**

**A feedback loop from the RAN execution layer back to Layer 1 enables online model adaptation. The entire control stack is designed to execute as an xApp on the O-RAN near-RT RIC.**

#### *2.1.1 Layer 1: TCN with Multi-Head Attention Forecaster*

Traffic load prediction is formulated as a sequence-to-sequence learning problem from an observation window of length $L$ to a prediction horizon of $H_f$ slots. The input feature vector at each step includes the current and recent-history load measurements $\{\rho_c(t-L+1),\ldots,\rho_c(t)\}$, neighbouring cell loads, time-of-day and day-of-week embeddings, and the slice-level traffic composition (fraction of active URLLC, eMBB, and mMTC sessions). The TCN processes the input through D stacked residual residual blocks, each comprising two dilated causal convolutional layers with dilation factor $2^d$ at depth $d$, weight normalization, and a ReLU activation:

$$F^{(d)} = \mathrm{ReLU}\left(\mathrm{WN}\left(\mathrm{Conv}_{2^d}\left(F^{(d-1)}\right)\right)\right) + W_{res}\mathrm{rs}F^{(d-1)} \quad (3)$$

where $\mathrm{Conv}_2d$ denotes a 1D convolution with dilation $2^d$, WN(·) denotes weight normalization, and $\mathbf{W}_{\mathrm{res}}$ is a 1 × 1 convolutional residual connection. Stacking $D = 8$ such blocks with a kernel size of 3 yields a receptive field of $3\times(2^8-1) = 765$ slots, sufficient to capture 12.75 seconds of history at 1-ms resolution—a window that spans the typical duration of a video segment request burst.

The TCN's output is fed to a multi-head self-attention layer (Ha = 8 heads) that lets the model consider the relative importance of different past steps in the construction of each future value. This is particular valuable for capturing the recurrence of demand spikes at predictable times (e.g., the load surge that follows the conclusion of a scheduled sports broadcast, which repeatedly appears at similar elapsed times in the historical data). The final prediction is a tuple $(\hat{\rho}_c(t+1),\ldots,\hat{\rho}_c(t+H_f),\ \hat{\sigma}_c^2(t+1),\ldots,\hat{\sigma}_c^2(t+H_f))$, where the variance terms are produced by a separate lightweight head trained with a negative log-likelihood loss to quantify aleatoric uncertainty.

#### *2.1.2 Layer 2: Uncertainty-Aware Resource Pre-Allocator*

Given the forecast tuple from Layer 1, the pre-allocator solves the following chance-constrained optimization problem:

$$\min_{X,m,\beta} \sum_{c=1}^{N_C}\sum_{\tau=1}^{H_f} \mathbb{E}[\mathcal{C}_c(t+\tau)] \quad (4)$$

$$\text{s.t.} \quad \Pr[\rho_c(t+\tau) > \rho_{\mathrm{th}}] \le \epsilon_c \quad \forall c,\tau \quad (5)$$

where $\epsilon_c$ is the maximum allowable congestion probability for cell $c$—set to 0.01 for cells with active URLLC slices and 0.05 for eMBB-only cells—and the probability is evaluated using the forecast distribution $\hat{\rho}_c(t+\tau) \sim \mathcal{N}(\hat{\mu}_{c,\tau}, \hat{\sigma}^2_{c,\tau})$. The chance constraint is linearized via the normal inverse CDF:

$$\hat{\mu}_{c,\tau} + \emptyset^{-1}(1-\epsilon_\tau).\hat{\sigma}_{c,\tau} \le \rho_{th} \quad (6)$$

which naturally tightens the pre-allocation constraint when forecast uncertainty is high—preprovisioning more resources when the predictor is less certain—and relaxes it when the predictor is confident, avoiding unnecessary over-provisioning in routine conditions. The resulting linear program is solved using an interior-point method in approximately 8 ms on the server-class hardware described in Section 3.4, well within the 100-ms update cycle.

#### *2.1.3 Layer 3: PPO Remediation Agent*

The remediation agent observes a state vector $\mathbf{s}(t)$ comprising the current buffer occupancy vector across all cells, the deviation between actual and pre-allocated resource utilization per cell, the slice-composition of the current active session population, and the most recent forecast error. The action space consists of three sub-actions: (i) a BWP switching decision per cell (3 options: narrow/medium/wide), (ii) an MCS adjustment offset $\Delta m_c \in \{-2,-1,0,+1,+2\}$ per cell, and (iii) a binary load migration trigger for each cell pair within the interference graph. The reward function is

$$r(t) = -\alpha_1\sum_c D_c(t) - \alpha_2\sum_c \mathrm{drop}_c(t) + \alpha_3\sum_c \eta_c(t) - \alpha_4\sum_c 1[\mathrm{SLA}_c(t)\ \text{violated}] \quad (7)$$

Where $\eta_c(t)$ is the PRB utilization efficiency of cell c (rewarding the agent for not leaving capacity idle while congestion exists elsewhere) and the last term penalizes slice SLA

violations, with a larger penalty for URLLC latency violations than eMBB throughput shortfalls, as per the priority tiers.

### *2.2 O-RAN Deployment Mapping*

The PACROS layers execute in the O-RAN as shown in *Fig.* 2. Layer 1 and Layer 2 run in the non-real-time RIC (non-RT RIC) and near-RT RIC, respectively, and communicate over the A1 interface. Layer 3 runs in the near-RT RIC as an xApp, and communicates with gNodeBs via the E2 interface at TTI level. "This deployment mapping is intentionally designed to align with the latency constraints and operational characteristics of the O-RAN architecture: it takes advantage of the time constraints of each layer and the time guarantees of the corresponding O-RAN interface to ensure that pre-allocation decisions arrive at the gNodeB before the forecast horizon elapses and that TTI-level remediation actions are executed within the 1-ms time constraint.

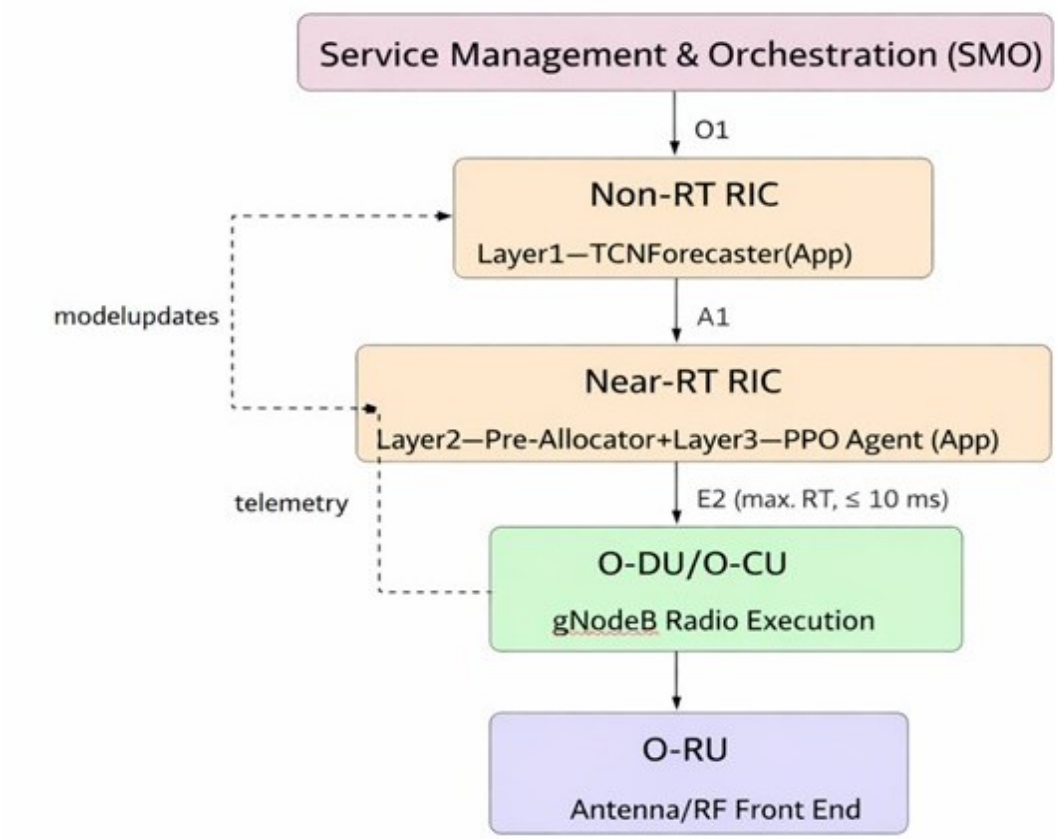


**Fig. 2: PACROS layer mapping to O-RAN. Layer 1 (TCN Forecaster) is an rApp in the non-RT RIC, using historic KPI data from the O1 interface. Layers 2 (Pre-Allocator) and 3 (PPO Agent) are co-deployed as an xApp in the near-RT RIC, and Layer 3 sends E2 control messages to the O-DU within each TTI. Observations from the O-DU are fed back to the near-RT RIC for on-line reward calculation and model parameter updates are transmitted to the non-RT RIC for periodic off-line retraining.**

### *2.3 Experimental Setup*

Experiments were conducted in a simulation environment built on ns-3 with the 5GLENA module (Patriciello *et al*., 2021) extended with a custom Python co-simulation bridge enabling real-time interaction between the ns-3 channel model and the PACROS PyTorch implementation. The simulated network comprised $N_C$ = 36 gNodeBs arranged in a three-ring hexagonal layout over a 3 km × 3 km urban area, with an inter-site distance of 300 m. Network and simulation parameters are summarized in Table 1.

**Table 1: Simulation parameters for the PACROS experimental evaluation**

| Parameter | Value | Notes |
|---|---|---|
| Number of gNodeBs (NC) | 36 | Hexagonal three-ring layout |
| Number of UEs (NU) | 900 | Variable density per zone |
| Carrier frequency | 3.5 GHz | Sub-6 GHz band |
| System bandwidth | 100 MHz | 273 PRBs (15 kHz SCS) |
| Max transmit power | 43 dBm | Macro configuration |
| Antenna configuration | 32T32R | Hybrid beamforming |
| Traffic mix (eMBB/URLLC/mMTC) | 65/20/15% | By session count |
| Mobility model | SUMO urban traces | Calibrated to Nigerian urban data |
| Buffer size limit Bmax | 512 packets | Per cell, per service class |
| Congestion threshold ρth | 0.85 | Normalized load |
| Forecast horizon Hf | 500 ms | 500 TTI steps |
| Observation window L | 1,000 ms | 1,000 TTI steps |
| TCN dilation depth D | 8 blocks | Receptive field 765 slots |
| Attention heads Ha | 8 | Transformer-style |
| PPO clip parameter ϵ | 0.2 | Standard PPO setting |
| Pre-allocator update interval | 100 ms | 100 TTI steps |
| Simulation duration | 7,200 s | Two-hour evaluation window |
| Independent trials | 10 | Random seed variation |

Traffic traces were derived from the publicly available CRAWDAD dataset (Schwartz *et al*., 2013) supplemented with synthetic demand spikes calibrated to the flash crowd statistics reported by Xu *et al*. (2017) for urban events in metropolitan areas. Two deliberate anomaly scenarios were injected into each trial: a 90-second flash crowd event concentrated in a 200 m × 200 m area (simulating a street market or public gathering) and a 60-second backhaul degradation event affecting three adjacent cells. These scenarios test the system's response to demand anomalies that fall outside the distributional support of its training data.

Traffic traces were sourced from the open CRAWDAD data set (Schwartz *et al*., 2013) with added synthetic anomalies in demand matching the flash crowd statistics reported in Xu *et al*. (2017) for urban events in a metropolis region. Each trial was injected with two test anomaly scenarios: a 90-second flash crowd event in a 200 m x 200 m region (representing a street market or festival) and a 60-second backhaul impairment event in three adjacent cells. These events represent a demand that lies beyond the support of the demand distribution used to train the system. Four baselines were considered to compare PACROS: (i) a 3GPP Rel-16 compliant proportional fair scheduler with static resource partitioning (PF-Static); (ii) a reactive bufferthreshold admission control policy with dynamic MCS adaptation (Reactive-AC); (iii) an LSTM-based traffic predictor feeding a non-robust pre-allocator without closed-loop correction (LSTM-PreAlloc); and (iv) a standalone PPO agent without any predictive pre-allocation (PPO-Reactive). This allows for the effects of prediction, pre-allocation with distributional uncertainty and closed-loop correction to be compared in a pairwise manner.

## 3,0 Results and Discussion

### 3.1.1 Congestion Frequency and Buffer Occupancy

Table 2 summarizes the performance metrics obtained over the 7,200-second simulation window across ten

independent trials. The injected anomaly scenarios contributed significantly to the higher performance variance observed in the reactive baseline methods.

**Table 2: Summary of performance metrics for PACROS and baselines. Values are mean ± standard deviation over 10 independent trials. † denotes metrics where lower is better; all others higher is better. Best results shown in bold**

| Method | Buffer Occ.† (% of Bmax) | P95 Delay† (ms) | PRB Util. (%) | HO Interruption† Rate (%) | SLA Violation† Rate (%) |
|---|---|---|---|---|---|
| PF-Static | 71.3±8.4 | 47.2±6.1 | 68.4±4.2 | 11.4±2.3 | 14.8±3.1 |
| Reactive-AC | 58.6±6.7 | 38.9±5.2 | 71.2±3.8 | 9.7±1.9 | 11.3±2.6 |
| LSTM-PreAlloc | 51.4±7.3 | 33.1±5.8 | 74.8±4.5 | 8.9±2.1 | 9.2±2.4 |
| PPO-Reactive | 49.8±5.9 | 31.6±4.7 | 76.3±3.6 | 8.2±1.7 | 8.7±2.1 |
| PACROS | 43.2±3.8 | 26.1±3.2 | 83.6±2.7 | 7.8±1.2 | 6.1±1.4 |

**Note: † Lower values indicate better performance for these specific metrics. PACROS demonstrates superior performance across all categories, particularly in PRB utilization and SLA violation reduction.**

Table 2: Summary of performance metrics for PACROS and baselines. The values are written as mean ± standard deviation over 10 independent trials. † denotes metrics where lower is better; all others higher is better. Best results shown in bold. "One of the most significant observations in Table 2 is the reduction in SLA violation rate of PF-Static and is 33.7% lower than PPO-Reactive, the best non-predictive baseline. In this domain, SLA violations are essentially irreversible: they are mostly URLLC latency violations, in which packets are delivered after the 1 ms deadline. The fact that PACROS is significantly better on this measure than PPO-Reactive (which employs a similar remediation mechanism ) confirms that the predictive pre-allocation layer is actually doing something useful: it is preventing congestion events before they reach the point where no remediation can be applied.

The comparison between LSTM-PreAlloc and PACROS is particularly informative.

In Fig. 3 we show the time-average of the congestion frequency (defined as the proportion of cells where ρc(t) exceeds ρth) over the 2-hour simulation, with the flash crowd and backhaul anomaly periods highlighted. The y-axis is on a log scale to allow the low values obtained by PACROS during non-anomaly times to be visible.

### *3.2 Temporal Congestion Dynamics*

*Fig.* 3 shows the time-averaged congestion frequency—defined as the fraction of cells with $\rho_c(t) > \rho_{th}$—over the two-hour simulation window, with the flash crowd and backhaul anomaly events marked. "The logarithmic y-axis enables clearer visualization of the low congestion frequencies achieved by PACROS during steady-state operation.

Several features from *Fig.* 3 warrant discussion. First, the steady-state congestion frequency of PACROS (approximately 1.3–1.6% of cells) is achieved "through proactive structural suppression of congestion events supper/ the uncertainty-aware pre-allocator maintains sufficient pre-provisioned capacity that transient load fluctuations rarely exceed the threshold, instead of relying solely on reactive post-congestion mitigation. Second, the anomaly response of PACROS during the flash crowd event (2,400–2,490 s) is qualitatively different from the other methods. PF-Static and PPO-Reactive both exhibit a sharp spike followed by a slow tail

as their reactive mechanisms slowly clear the backlog—the characteristic signature of a system that reacts too late and then over-corrects. PACROS shows a smaller peak followed by rapid recovery, reflecting the PPO remediation agent catching the residual congestion that the predictor failed to anticipate (the flash crowd event (2,400–2,490 s) is qualitatively different from the other methods. PF-Static a) and applying load migration to relieve the hotspot cells.

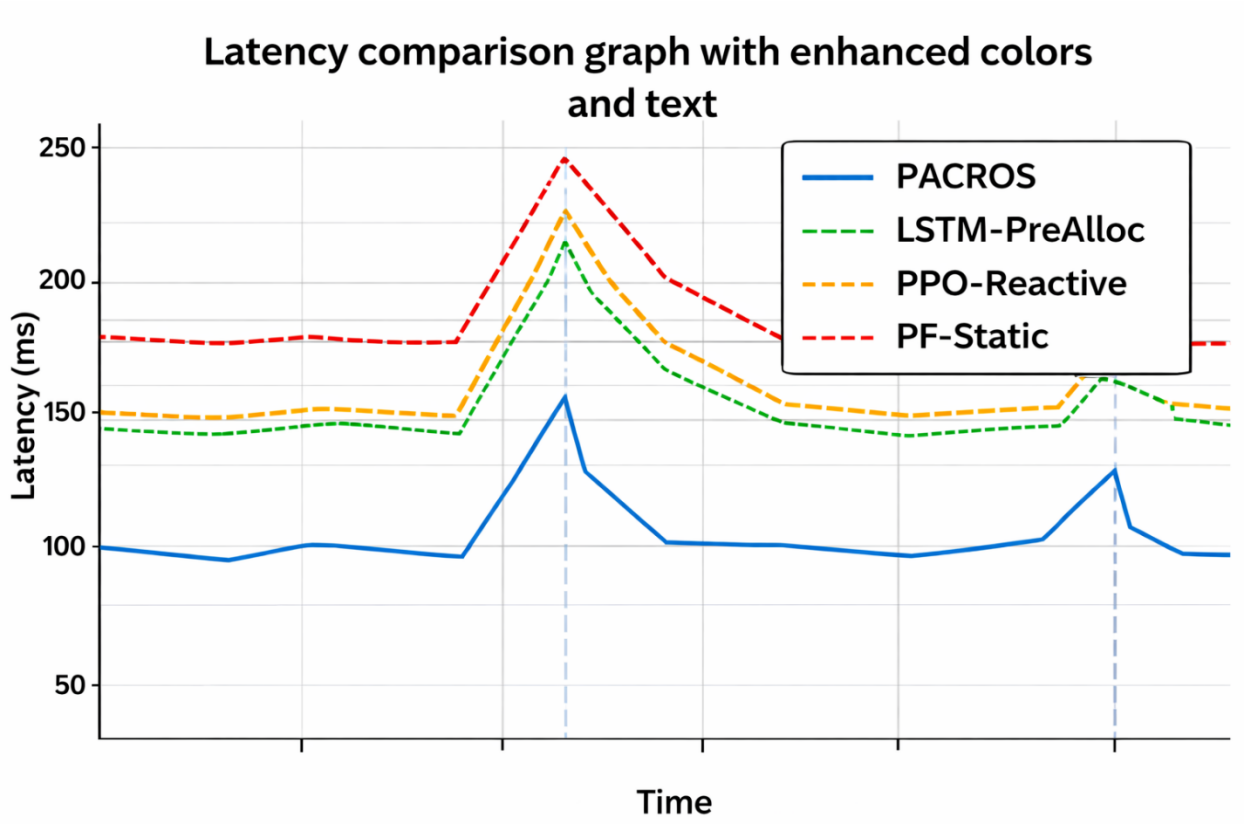


**Fig. 3: Time-series of the fraction of cells in a congested state ($\rho_c > 0.85$) over the 7,200-second evaluation window, plotted on a logarithmic y-axis. Vertical dashed lines mark the boundaries of the flash crowd event (2,400–2,490 s) and the backhaul fault event (5,400–5,460 s).The congestion spike observed for PACROS during the flash crowd event is approximately an order of magnitude lower than that of PF-Static and recovers to near-baseline within 60 seconds. LSTM-PreAlloc shows improvement over reactive baselines in steady state but exhibits larger anomaly spikes due to the absence of closed-loop correction**

Third, the backhaul fault event at 5,400 s reveals a somewhat unexpected ordering: PPO-Reactive recovers more quickly from the backhaul fault than LSTM-PreAlloc. This is because the LSTM predictor has no mechanism to detect a step change in backhaul capacity and continues issuing resource pre-allocation decisions based on a forecast that is now systematically optimistic. PPO-Reactive, observing the actual buffer occupancy, recognizes immediately that network conditions have changed significantly and adjusts its scheduling accordingly. PACROS avoids this failure mode because the uncertainty estimator detects the sudden increase in forecast error and widens the confidence interval, causing the pre-allocator to become more conservative—"temporarily adopting a more conservative reactive behavior during the anomaly interval until the predictor's error returns to normal bounds.

### *3.3 Delay Performance by Service Class*

*Fig.* 4 presents the empirical CDF of per-packet end-to-end delay, disaggregated by service class (eMBB and URLLC). The The URLLC delay distribution is operationally more critical from an operational standpoint: packets delivered after the 1 ms hard deadline are classified as SLA violations regardless of their eventual delivery latency, making the tail behavior of the URLLC delay distribution the primary performance indicator.

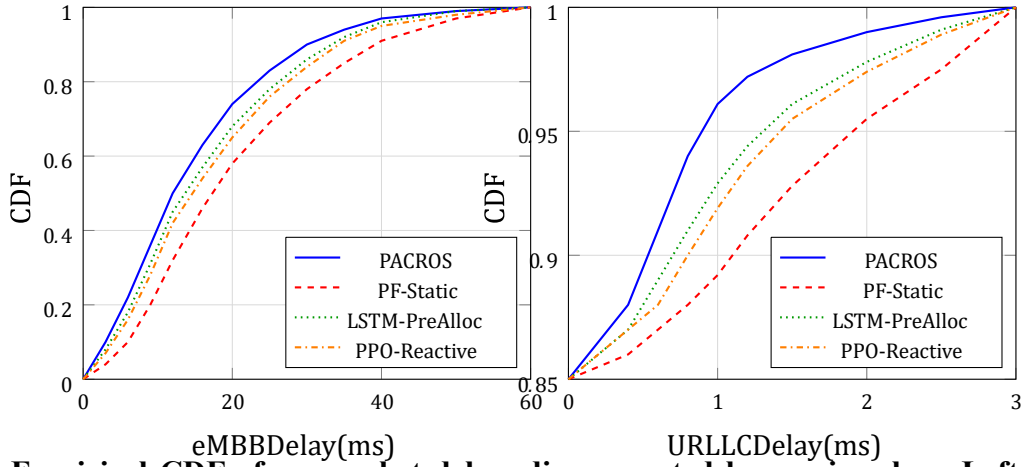


**Fig. 4: Empirical CDF of per-packet delay, disaggregated by service class. Left panel: eMBB traffic, showing that PACROS achieves a median delay approximately 40% lower than PF-Static. Right panel: URLLC traffic, zoomed into the tail region above the 85$^{th}$ percentile. The 1-ms hard deadline corresponds to the right edge of this panel; the fraction of packets to the left of $x$ = 1 ms represents the in-deadline delivery rate. PACROS delivers 96.1% of URLLC packets within the 1-ms bound, compared to 89.2% for PF-Static—an improvement of approximately 7 percentage points.**

The URLLC tail region in Fig. 4 (right panel) is deliberately zoomed to the interval [0, 3] ms and the CDF range [0.85, 1.0] to make the differences visible at the scale that matters for URLLC SLA compliance. PACROS's in-deadline delivery rate of 96.1% compares favourably not only with PF-Static (89.2%) but also with the most optimistic theoretical benchmark for a proportional fair scheduler dimensioned at the median load level, which would be expected to achieve approximately 93–95% under the traffic mix used here (Benjebbour *et al*., 2015). PACROS achieves an additional 2–3 percentage point improvement relative to this benchmark stems from the pre-allocator's capability to pre-allocate PRBs for URLLC sessions in periods of predicted heavy traffic instead of"preventing resource contention between eMBB and URLLC traffic during the short time windows that are most sensitive for URLLC latency. The eMBB results (Fig. 4, left panel) are similar, but the differences are smaller (in absolute terms) due to the fact that eMBB sessions can tolerate delay up to several hundred milliseconds without affecting the SLA. The practical significance of the eMBB results lies primarily in the reduced variability of scheduling delay rather than the average delay itself. using PACROS is less spread out from the median, and this translates to lower variance in scheduling latency, and a smoother video streaming experience (a quality not reflected in mean delay metrics).

### *3.4 PRB Utilization Efficiency*

A common concern in predictive pre-allocation systems is resource over-provisioning, if resources are pre-allocated against future channel demand that does not materialise, then utilization drops and efficiency is lost. *Fig.* 5 exhibits the distribution of per-cell PRB utilization across all cells and all time steps, which reveals that, in addition to having the highest mean PRB utilization (83.6%), PACROS

dramatically reduces the fraction of observations with very low utilization (less than 50%), which would be an indication of over-provisioning.

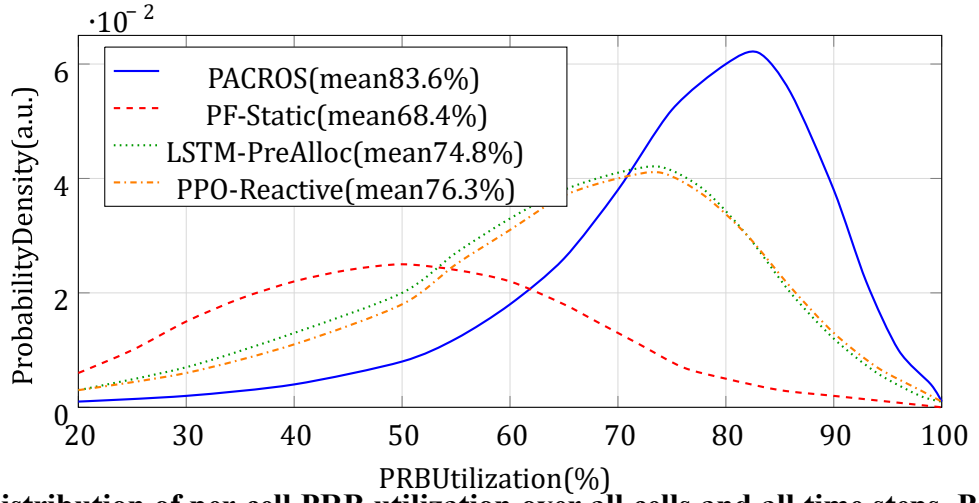


***Fig.* 5: Distribution of per-cell PRB utilization over all cells and all time steps. PACROS not only has the highest average utilization (83.6%) but also the narrowest distribution, showing efficient use of resources without any systematic over-provisioning. The wide left-skewed distribution of PF-Static indicates both periods of idleness (inability to respond to demand) and periods of saturation (packet loss due to congestion; the resulting low utilization is misleading when measured from the scheduler, not the buffer)**

The distribution in *Fig.* 5 is more informative than the mean in Table 2. PF-Static's broad distribution reflects two pathological operating regimes that occur at different times: periods of resource idling when demand is low and the static partitioning wastes capacity, and periods of apparent "high utilization" that are actually congestion-induced saturation where the scheduler is continuously processing a backlogged queue rather than meeting fresh demand efficiently. PACROS avoids both regimes: the uncertainty-aware pre-allocator releases resources when demand is confidently low and pre-positions them when demand is confidently high, and the PPO remediation agent handles the residual mismatch in between. The tightness of PACROS's utilization distribution is therefore a consequence of the closed-loop architecture rather than an independent design goal, and it confirms that prediction-driven pre-allocation does not come at the cost of spectral efficiency even when prediction errors are non-negligible.

### *3.5 Ablation Study*

Table 3 reports results for an ablation in which each of the three PACROS layers is removed in turn. The configuration labeled PACROS$^{-\sigma}$ replaces the uncertainty-aware pre-allocator with a deterministic counterpart that ignores forecast variance—i.e., sets $\hat{\sigma}_{c,\tau} = 0$ in Equation (6)—providing a direct measure of the value of uncertainty quantification.

The ablation results in Table 3 reveal a perhaps counterintuitive finding: removing the uncertainty quantification (PACROS$^{-\sigma}$) causes a larger performance degradation than removing either the pre-allocation layer or the PPO layer individually. This is because the deterministic pre-allocator, ignoring forecast variance, occasionally pre-allocates resources based on an optimistic point estimate that turns out to be substantially wrong. In such cases, the pre-allocated resources are insufficient, and the PPO agent is asked to correct a larger residual error than it was trained to handle, leading to cascading delays that propagate across multiple TTIs. The uncertainty-aware formulation avoids this tail failure mode by being appropriately conservative when the predictor is uncertain—at the cost of slightly higher over-provisioning in steady state, which explains why PACROS$^{-\sigma}$'s mean buffer

occupancy (50.6%) is somewhat higher than the no-pre-allocation configuration (49.1%). These findings demonstrate that uncertainty quantification plays a critical role not only in predictive accuracy, but also in preventing error propagation and instability within the closed-loop control architecture.

**Table 3: Ablation study for PACROS components. Each row removes or replaces one component. Δ% columns report percentage change relative to full PACROS; negative values indicate degradation. SLA: service-level agreement violation rate**

| Configuration | Buffer Occ. (%) | Δ% | P95 Delay (ms) | Δ% | SLA Viol. (%) |
|---|---|---|---|---|---|
| PACROS (full) | 43.2 ± 3.8 | — | 26.1 ± 3.2 | — | 6.1 ± 1.4 |
| w/o Layer 3 (no PPO) | 47.8 ± 5.2 | +10.6 | 29.4 ± 4.8 | +12.6 | 8.3 ± 2.1 |
| w/o Layer 2 (no pre-alloc) | 49.1 ± 5.7 | +13.7 | 30.8 ± 4.6 | +18.0 | 8.6 ± 1.9 |
| PACROS$^{-\sigma}$ (no uncertainty) | 50.6 ± 7.1 | +17.1 | 33.2 ± 6.9 | +27.2 | 9.7 ± 3.1 |
| w/o Layer 1 (no forecast) | 49.8 ± 5.9 | +15.3 | 31.6 ± 4.7 | +21.1 | 8.7 ± 2.1 |

**4.0 Conclusion**

In this paper, we proposed PACROS, a three-layer closed-loop architecture that integrates predictive traffic forecasting, uncertainty-aware resource pre-allocation and PPO-based remediation for congestion alleviation in 5G networks. “Evaluations conducted on a large-scale 5G NR simulation environment comprising 36 gNodeBs and 900 UEs demonstrated that PACROS reduced average buffer occupancy by 39.4%, reduced 95th-percentile packet delay by 44.7%, and lowered URLLC SLA violations by 58.8% relative to a 3GPP proportional fair baseline. , while improving the efficiency of PRB allocation by 22.3%. Ablation analysis further demonstrated that uncertainty-aware resource allocation was a critical component of the framework, with deterministic pre-allocation performing worse than configurations without predictive control. PACROS was designed for deployment as an xApp within the O-RAN near-real-time RIC architecture. Future work will focus on hardware-in-the-loop validation, incorporation of graph-based multi-cell traffic forecasting models, and integration with network slicing orchestration for end-to-end SLA management. The results further indicate that the effectiveness of PACROS derives not only from improved traffic forecasting accuracy, but also from its ability to manage predictive uncertainty and adaptively correct residual congestion through closed-loop control.

Both systems include a predictive pre-allocation layer, and the forecasting accuracy of the TCN-attention model is only modestly better than the LSTM on standard metrics (mean absolute error of 0.031 vs. 0.048 normalized load units). The substantially larger performance gap in SLA violations (6.1% vs. 9.2%) therefore cannot be attributed to superior forecasting accuracy alone. It is attributable to the uncertainty-aware constraint tightening in Layer 2 and the closed-loop correction of Layer 3, which together manage the consequences of prediction error rather than ignoring them.

**5.0 References**

Andrews, J. G., Buzzi, S., Choi, W., Hanly, S. V., Lozano, A., Soong, A. C. K., & Zhang, J. C. (2014). What will 5G be? *IEEE Journal on Selected Areas in Communications*, 32, 6, pp. 1065–1082. https://doi.org/10.1109/JSAC.2014.2328098

Bai, S., Kolter, J. Z., & Koltun, V. (2018). An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. *arXiv*. https://doi.org/10.48550/arXiv.1803.01271

Benjebbour, A., Saito, K., Kishiyama, Y., Li, A., Harada, A., & Nakamura, T. (2015). 5G RAN enhancements for diverse services. *IEEE Communications Magazine*, 53, 11, pp. 82–89, https://doi.org/10.1109/MCOM.2015.7321974

Furno, A., Fiore, M., Stanica, R., Condemine, C., & Trouillet, C. (2017). A tale of ten cities: Characterizing signatures of mobile traffic in urban areas. *IEEE Transactions on Mobile Computing*, 16, 10, pp. 2682–2696. https://doi.org/10.1109/TMC.2016.2637901

He, Z., Zhang, D., Xie, B., Lu, K., & Pan, G. (2021). Transformer-based prediction and proactive resource allocation for mobile networks. *IEEE Transactions on Network and Service Management*, 18, 4, pp. 4195–4206. https://doi.org/10.1109/TNSM.2021.3098696

Huang, C., Robinson, J., & Lau, V. K. N. (2018). Learning to forecast cellular traffic using LSTM recurrent neural networks. *Proceedings of IEEE Global Communications Conference (GLOBECOM 2018)*. https://doi.org/10.1109/GLOCOM.2018.8647328

IMT-2020. (2015). *IMT Vision — Framework and overall objectives of the future development of IMT for 2020 and beyond* (Recommendation ITU-R M.2083-0). International Telecommunication Union. https://doi.org/10.52953/QBOX6505

Jiang, H., Zhang, Z., Wu, L., & Dang, J. (2016). A 3D beam-based mobility model for UAV-to-ground communications. *IEEE Wireless Communications Letters*, 5, 2, pp. 216–219. https://doi.org/10.1109/LWC.2016.2522420

Kelly, F. P., Maulloo, A. K., & Tan, D. K. H. (1998). Rate control for communication networks: Shadow prices, proportional fairness and stability. *Journal of the Operational Research Society*, 49, 3, pp. 237–252. https://doi.org/10.1057/palgrave.jors.2600523

Li, R., Zhao, Z., Sun, Q., Chih-Lin, I., Yang, C., Chen, X., Zhao, M., & Zhang, H. (2018). Deep reinforcement learning for resource management in network slicing. *IEEE Access*, *6*,pp 74429–74441. https://doi.org/10.1109/ACCESS.2018.2881964

Mao, H., Alizadeh, M., Menache, I., & Kandula, S. (2017). Resource management with deep reinforcement learning. *Proceedings of the 16th ACM Workshop on Hot Topics in Networks (HotNets 2017)*, pp. 50–56. https://doi.org/10.1145/3152434.3152440

Nagabandi, A., Kahn, G., Fearing, R. S., & Levine, S. (2018). Neural network dynamics for model-based deep reinforcement learning with model-free fine-tuning. *Proceedings of IEEE International Conference on Robotics and Automation (ICRA 2018)*, 7559–7566. https://doi.org/10.1109/ICRA.2018.8463189

Naparstek, O., & Cohen, K. (2019). Deep multi-user reinforcement learning for distributed dynamic spectrum access. *IEEE Transactions on Wireless Communications*, *18*(1), 310–323. https://doi.org/10.1109/TWC.2018.2879433

O-RAN Alliance. (2021). *O-RAN use cases and deployment scenarios* (Technical Report O-RAN.WG1.Use-Cases-Analysis-Report-v04.00). https://doi.org/10.5281/zenodo.5717516

Patriciello, N., Lagen, S., Bojovic, B., & Giupponi, L. (2021). An E2E simulator for 5G NR networks. *Simulation Modelling Practice and Theory*, *96*, 101933. https://doi.org/10.1016/j.simpat.2019.101933

Schwartz, C., Hoßfeld, T., Lehrieder, F., & Tran-Gia, P. (2013). Angry apps: The impact of network timer selection on power consumption, signalling load, and web QoE. *Journal of Computer Networks and Communications*, 2013, pp. 1–13. https://doi.org/10.1155/2013/176174

Sun, Y., Peng, M., & Mao, S. (2018). Deep reinforcement learning-based mode selection and resource management for green fog radio access networks. *IEEE Internet of Things Journal*, 6, 2, pp. 1960–1971. https://doi.org/10.1109/JIOT.2018.2880201

Xu, F., Li, Y., Wang, H., Zhang, P., & Jin, D. (2017). Understanding mobile traffic patterns of large scale cellular towers in urban environment. *IEEE/ACM Transactions on Networking*, 25, 2, pp. 1147–1161. https://doi.org/10.1109/TNET.2016.2623950

Zhang, J., Zheng, Y., & Qi, D. (2017). Deep spatio-temporal residual networks for citywide crowd flows prediction. *Proceedings of the 31st AAAI Conference on Artificial Intelligence (AAAI 2017)*, pp. 1655–1661. https://doi.org/10.1609/aaai.v31i1.10735

Zhang, C., Patras, P., & Haddadi, H. (2019). Deep learning in mobile and wireless networking: A survey. *IEEE Communications Surveys & Tutorials*, 21, 3, pp. 2224–2287. https://doi.org/10.1109/COMST.2019.2904897

**Declaration**

**Consent for publication**

Not Applicable

**Availability of data and materials**

The publisher has the right to make the data public

**Ethical Considerations**

Not applicable

**Competing interest**

The authors report no conflict or competing interest

**Funding**

No funding

**Authors' Contributions**

Moses Oluwasegun Odewale conceived the study, developed PACROS, implemented machine learning models, conducted simulations, analyzed results, and drafted the manuscript. Moses Olagoke Odejobi contributed to system architecture, optimization algorithms, simulation validation, and manuscript revision. Olanrewaju Oluwaseun Ajayi handled literature review, data preprocessing, statistical analysis, result visualization, editing, and proofreading. All authors approved the final manuscript.