
Low-latency Adaptive Communication Protocols for Ultra-dense Network Environments

Jihua He

*School of Information Engineering, Sichuan Top IT Vocational Institute, 611743,
Chengdu, Sichuan, China
cdxfrmmny@163.com*

Received 23 June 2025; Accepted 19 August 2025

Abstract

Ultra-dense networks (UDNs) face serious latency fluctuations and throughput degradation issues under high concurrency access and resource competition conditions. Traditional transmission protocols struggle to balance low latency and high stability in dynamic scenarios. In response to this challenge, this paper proposes a low latency adaptive communication protocol (MLACP), which constructs a multi-layer control system consisting of a physical access layer, a resource scheduling layer, and an adaptive decision layer. Through a cross layer feedback mechanism combined with RNN based short-term state prediction and DQN based strategy optimization, dynamic adjustment of resource slicing, distributed collaboration, and path selection is achieved. The protocol design is implemented in the system level simulation environment of a 3GPP UMi SC channel model and a Poisson cluster process, and integrated with ZeroMQ and PyTorch on the NS-3.36 platform. The experiment covered different user densities and link states, with each scenario running independently 10 times and taking the average. The results showed

Journal of ICT Standardization, Vol. 13_2, 157–180.

doi: 10.13052/jicts2245-800X.1324

© 2025 River Publishers

that under high-density conditions of 1500 UE/km², MLACP outperformed TCP Reno, QUIC, and the URLLC simplification scheme in terms of end-to-end latency, peak throughput, packet loss rate, path stability, and energy consumption. Moreover, it maintained controllable performance degradation in robustness tests such as link interruption, prediction bias, and base station failure. This result validates the feasibility and adaptability of the proposed protocol in dynamic and interference complex UDN environments, providing methodological references and an experimental basis for the design of low latency and intelligent communication systems.

Keywords: Ultra-dense networks, adaptive communication protocols, low latency control, dynamic resource slicing.

1 Introduction

With the wide deployment of fifth generation (5G) mobile communication technology and the gradual research of the sixth generation (6G) network, communication systems are moving towards the goal of ultra-high capacity, ultra-low latency, and ultra-large connectivity. Among them, ultra-dense networks (UDNs), as one of the important forms of future mobile networks, significantly improve spectrum reuse efficiency and coverage capability with their high-density small base station deployment and heterogeneous access. However, with the exponential growth in the number of user devices and the rapid emergence of delay-sensitive services (e.g., AR/VR, telematics, industrial control, etc.), traditional communication protocols are frequently exposed to bottlenecks such as access congestion, channel degradation, routing imbalance, and resource scheduling rigidity in UDN environments, which seriously constrain the sustainable enhancement of network performance.

To cope with these challenges, ultra-reliable low-latency communication (URLLC) has gradually become the focus of international communication standardization organizations, the core of which lies in achieving millisecond end-to-end delay and near-zero packet loss under complex wireless conditions [1]. Although URLLC protocols demonstrate low-latency and high-reliability performance in industrial and medical environments, they face significant limitations when applied to ultra-dense network (UDN) scenarios. Most URLLC designs rely on centralized scheduling policies and preconfigured static parameters, which limit their responsiveness to sudden user behavior changes and the rapid link fluctuations commonly observed in UDN. Moreover, these protocols often operate with rigid QoS class mapping

and lack real-time cross-layer sensing, leading to poor adaptability in heterogeneous deployments with variable node densities and spatial interference. For example, under frequent topology reconfigurations, URLLC fails to optimize routing paths adaptively due to the absence of predictive feedback and decentralized control mechanisms. These constraints underscore the need for a novel communication protocol that integrates real-time sensing, AI-based decision-making, and flexible resource management tailored to the dynamic characteristics of UDNs [2].

Recently, studies have explored communication strategies adapted to the characteristics of UDNs from multiple technical dimensions: Alablani et al. (2021) proposed a load-aware cell selection mechanism for heterogeneous networks, which improves the efficiency of local scheduling [3]; Park et al. (2022) emphasized that the construction of ultimate low-latency paths needs to take into account the optimization of the number of hops and the avoidance of interference [4]; and Biswas et al. (2025) attempted to introduce a deep reinforcement learning strategy to realize intelligent access control and achieve some scheduling performance improvement [5]. However, most of the approaches still have the following limitations: first, lack of system-level cross-layer architectural design, and lack of linkage between access, scheduling, and routing policies; second, resource management mechanisms are mostly centralized, which is difficult to adapt to the edge collaboration and elastic control requirements in UDN scenarios; third, AI modules mostly stay at the level of static models, lack of real-time perception and adaptive feedback mechanisms, and are unable to effectively respond to rapid fluctuations in link state and load intensity. Third, AI modules are mostly at the level of static models, lacking real-time sensing and adaptive feedback mechanisms, and unable to respond effectively to the rapid fluctuations of link state and load intensity.

In view of this, this paper proposes a low-latency adaptive communication protocol (MLACP) for ultra-dense network environments and constructs a multi-layer communication control system that includes dynamic resource slicing, distributed collaboration mechanisms, and AI-driven decision engines. The protocol aims to achieve an end-to-end low-latency guarantee while optimizing link stability and throughput capacity in high-density scenarios. By constructing a system-level modeling framework and NS-3 simulation platform, this paper further verifies the effectiveness and scalability of the protocol in high-load environments. This study not only provides a feasible protocol design path for multi-service converged communication in UDN scenarios but also provides theoretical support and technical

basis for “intelligent, low-latency, and strong adaptive” protocol evolution in future 6G networks. By constructing a system-level modeling framework and NS-3 simulation platform, this paper further verifies the effectiveness and scalability of the protocol in high-load environments.

2 Ultra-dense Network Environment Modeling

In the design of communication protocols for ultra-dense networks (UDNs), accurately modeling the network environment is a fundamental part of protocol architecture and algorithm verification. Considering the characteristics of a UDN scenario, such as high density of user devices, large fluctuation of link quality, and strong spatial interference, a system-level modeling framework with spatial correlation, traffic dynamics, and channel non-smoothness is constructed [6]. The model simulates the spatial distribution of terminals with the normal Poisson cluster process and combines the three-dimensional coordinate system to introduce the height difference between base stations and users to simulate the interference distribution in a dense deployment environment. The channel model adopts the extended 3GPP UMi-street canyon model, which supports the composite simulation of path loss, shadow fading and fast fading to improve the spatial granularity and accuracy of the simulation.

In order to accurately describe the network load dynamics, a time-varying user access intensity function based on β -distribution $\lambda(t) = \lambda_0 \cdot t^\beta \cdot e^{-\alpha t}$ is introduced to portray the peak fluctuation characteristics of access density over time, which strengthens the modeling basis for the protocol’s time-domain adaptability. In order to guarantee the controllability of resource scheduling experiments in protocol design, the simulation platform also introduces a spectrum resource characterization matrix $R_{ij} \in \{0, 1\}^{N \times M}$, which defines the resource allocation state among N cells and M frequency bands.

Figure 1 illustrates the simulation structure of the constructed network topology, containing randomly distributed microbase stations and user nodes with adjustable distribution density parameters. Table 1 lists the main parameters used in the modeling, including user density, cell radius, number of frequency bands, simulation area, etc. The protocol architecture and algorithm modules in the subsequent sections are designed and verified based on this modeling environment.

Compared to existing works on multi-agent coordination and deep reinforcement learning (DRL)-based scheduling, this study introduces the

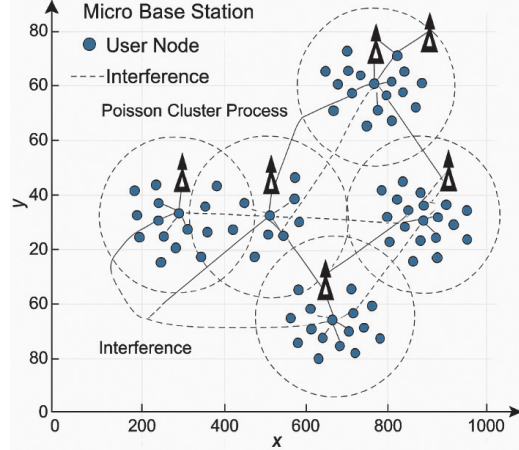


Figure 1 Ultra-dense network modeling structure.

Table 1 Ultra-dense network modeling parameters

Parameter Name	Parameter Symbol	Default Value	Description
Average user density	ρ_u	1000 UE/km ²	Specifies the average user equipment density in the simulation area
Density of micro base stations	ρ_{bs}	64 BS/km ²	Number of micro base stations eligible for UDN deployment
Cell coverage radius	rc	50 m	Service range of micro base stations
Channel model	Channel model	3GPP UMi SC	Supports modeling of multipath, fading, shadowing and other parameters
Interference update cycle	Tint	50 ms	Time period between interference refresh and link state change
User access fluctuation function	$\lambda(t)$	β function control ($\beta = 1.2$)	Describes the time-dynamic process of user access density

following technical contributions:

- (1) The protocol is the first to embed an AI engine that couples temporal prediction (via RNN) and policy optimization (via DQN) into a cross-layer control framework for UDN environments. Previous methods typically implement single-layer DRL without forecast-driven adaptation.

- (2) A reconfigurable three-layer protocol architecture is constructed with modular design and full integration into the NS-3 simulation environment, allowing dynamic protocol-level interactions.
- (3) A unified joint strategy update mechanism is proposed to simultaneously optimize delay, resource efficiency, and routing stability – a tri-objective focus often neglected in existing isolated scheduling models.

3 Low Latency Adaptive Protocol Architecture Design

3.1 General Framework of the Protocol

In order to meet the demand for strict control of communication delay in highly concurrent, multi-service access scenarios in ultra-dense network (UDN) environments, this paper designs a low-latency adaptive communication protocol architecture with a hierarchical control structure and a cross-layer feedback mechanism. The architecture follows the top-down modular design concept and is divided into three functional layers: physical access layer, resource scheduling layer, and adaptive decision-making layer, each of which has independent decision-making control units and configurable interfaces to ensure the system's rapid adaptability to changes in network density, interference fluctuations, and dynamic service requirements.

Physical access level to the wireless link access control, constructed based on an interference reachability graph (interference reachability graph) dynamic access window adjustment module. By sensing the interference boundary of the current link and the queuing situation of neighboring nodes, the size of the access window is dynamically configured to ensure that the equipment can still achieve stable transmission under high interference density. The resource scheduling layer takes the frequency domain resource mapping matrix R_{ij} as the core, optimizes the allocation of spectrum resources to different cells (i) in different time periods (j), and dynamically adjusts the resource slicing configuration by combining factors such as user density, traffic intensity, and channel state to improve the efficiency of spectrum utilization. The adaptive decision-making layer integrates an artificial intelligence (AI)-based inference engine to dynamically adjust and optimize the communication strategy by comprehensively modeling the network state, topology evolution trend, and historical feedback data.

The protocol architecture design not only provides a unified scheduling interface for the three types of core algorithms (access control, resource allocation, and path optimization) but is also deeply coupled with the

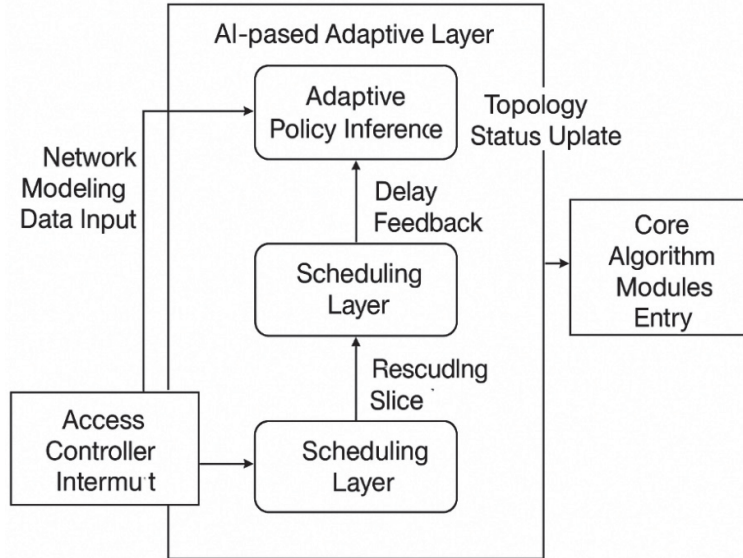


Figure 2 Low latency adaptive communication protocol system architecture diagram.

NS-3 simulation platform to support module-level encapsulation and high-frequency feedback data interactions with good simulation accuracy and protocol comparison operability. The system architecture is shown in Figure 2, which clearly demonstrates the logical relationship between the module division of each layer, the feedback path and the control process.

3.2 Key Technological Innovations

3.2.1 Dynamic resource slicing technology

In the ultra-dense network (UDN) scenario with multi-user concurrent access and highly dynamic business requirements, traditional static spectrum allocation mechanisms often fail to meet the dual requirements of spectrum utilization efficiency and interference control capability for low latency communication. Therefore, this article proposes a dynamic resource slicing mechanism driven by traffic intensity to improve the flexibility and scalability of resource scheduling in complex environments.

This mechanism is first based on the abstraction of resource units, discretizing the system spectrum space B into multiple sets of logical slices $\{S_i\}$, where each slice represents a frequency domain resource block with independent scheduling rights and interference isolation capabilities. On this

Table 2 Configuration table of key parameters of dynamic resource slicing mechanism

Parameter Name	Parameter Symbol	Default Value	Description
Total bandwidth	B	20 MHz	Spectral range available to the system
Minimum slice granularity	Δf	180 kHz	Frequency domain width of a single slice
Slice tuning period	T_{slice}	100 ms	Interval for dynamic adjustment of frequency resources
Weighting factor	ω_1, ω_2	0.6, 0.4	Control the trade-off between delay and utilization

basis, the system introduces a three-dimensional real-time load matrix $L_{ij}(t)$ to dynamically perceive the traffic intensity and scheduling requirements of each cell i in time period j and system time slot t . The allocation status and size of resource slices change in real-time with the network state to achieve scene oriented adaptive resource orchestration.

The allocation strategy of slicing aims to minimize the average delay of the integrated network and maximize resource utilization. The objective function is defined as follows [8]:

$$\min_{S_i(t)} \sum_{i=1}^N \sum_{j=1}^M [\omega_1 \cdot D_{ij}(t) + \omega_2 \cdot \rho_{ij}(t)]$$

where $D_{ij}(t)$ denotes the average delay of cell i at time slot t , $\rho_{ij}(t)$ denotes the resource utilization rate, and ω_1, ω_2 are the dynamic weighting coefficients. To realize the parameterized configuration of scheduling control, Table 2 lists the core design parameters such as band division, minimum slice width, and resource granularity unit in the system.

3.2.2 Distributed collaboration mechanism

In ultra-dense networks (UDNs), due to the large number of nodes and variable link states, traditional centralized scheduling methods have obvious scalability barriers and processing bottlenecks, especially in high concurrency scenarios, which can easily lead to information transmission congestion and scheduling response delays. Therefore, this article proposes a distributed collaboration mechanism with edge autonomy capability to achieve autonomous scheduling and cross node collaboration of protocols at the micro base station level. This mechanism is based on the multi-agent

distributed coordination framework, which models each micro base station node as an intelligent agent. Each agent has local link state awareness, independent resource decision-making, and cross node information exchange capabilities, and can autonomously formulate scheduling actions based on local and neighboring states. Specifically, the system designs a local policy function for each node i : $\pi_i(a_i|s_i, N_i)$, where s_i represents the local state of node i , N_i is the set of adjacent nodes, and a_i is the scheduling action. In order to maintain the consistency of the overall scheduling of the system, an edge synchronization channel is introduced in the mechanism to asynchronously propagate local state and resource usage information between nodes, avoiding communication burden caused by frequent synchronization. Meanwhile, a global optimization objective function is introduced with the goal of jointly minimizing scheduling delay and collaborative cost [9]:

$$\min \left[\sum_{i \in V} (\alpha \cdot D_i + \beta \cdot C_{i, N_i}) \right]$$

Among them, D_i is the average scheduling delay of node i , C_i and N_i represent the coordination cost of scheduling conflicts with neighboring nodes, and the coefficients α and β control the trade-off between delay and consistency. The system adopts a distributed scheduling cycle based on time slots, where each agent executes scheduling decisions and updates state vectors at fixed time intervals. To further enhance the flexibility of network control, the mechanism adopts asynchronous state propagation, which enables some nodes to respond quickly when there is a link change, without relying on global synchronization, thereby improving the adaptability and stability of the protocol in dynamic environments. Figure 3 shows the collaborative topology structure between proxy nodes under this mechanism. By constructing reasonable adjacency relationships and local perception radii, the system achieves coordination and minimizes resource conflicts in a distributed architecture, providing a stable foundation for subsequent AI scheduling and adaptive optimization.

3.2.3 AI-driven adaptive engine

In ultra-dense network (UDN) environments, the link state, user access behavior, and interference distribution are subject to high variability and unpredictability. Traditional rule-based control frameworks often fail to adapt quickly to such dynamics. To address this, an AI-driven adaptive engine is introduced and embedded in the top-level control layer of the protocol to

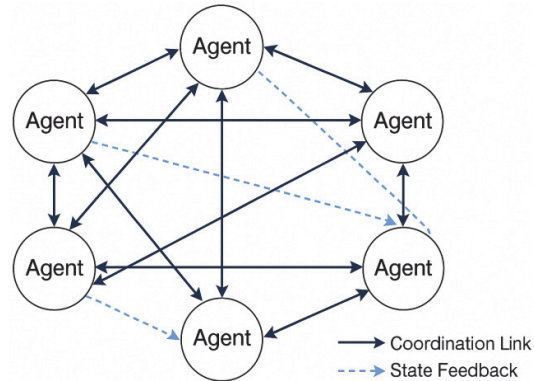


Figure 3 Topology of inter-node collaboration under a distributed collaboration mechanism.

function as the central intelligence for decision-making and policy feedback across modules.

The engine adopts a dual-branch architecture of “prediction + optimization.” The prediction branch applies a recurrent neural network (RNN), selected for its proven ability to model time-series patterns and capture sequential dependencies in link congestion and user access trends. Input variables include queue length Q_t , access rate λ_t , and average interference γ_t , and the output yields the short-term predicted state $s^t + 1$. The model uses gradient-clipped Adam optimization to maintain stability in multi-slot training cycles. The strategy branch is based on a deep Q-network (DQN), a widely adopted deep reinforcement learning (DRL) framework effective for discrete action environments such as spectrum assignment and path decision. The learning objective is to minimize cumulative delay and contention, with an ε -greedy strategy balancing policy convergence and action diversity. Policy stability is evaluated via return convergence analysis across training epochs. To support online adaptation, the engine integrates a bidirectional data channel connecting with the resource slicing and access control modules. Predicted states and policy outputs inform real-time tensor updates, while real-world feedback enables parameter refinement. A prioritized experience replay buffer and soft target update mechanism further improve generalization across variable traffic loads. Figure 4 illustrates the engine’s module structure, including state abstraction, RNN forecasting, DRL-based optimization, and the feedback interface. This component enhances protocol responsiveness and decision flexibility under dynamic UDN conditions, enabling low-latency and high-reliability communication goals.

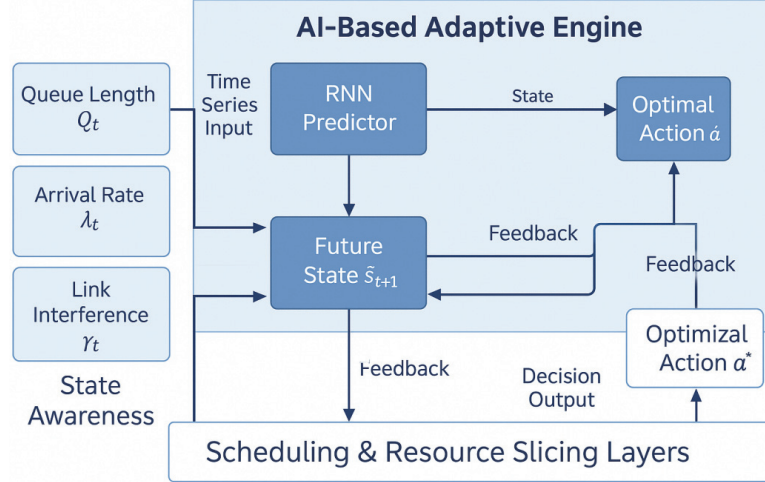


Figure 4 Structure diagram of an AI-driven adaptive engine.

4 Core Algorithm Design and Implementation

4.1 Low-latency Access Control Algorithm

In order to cope with the problem of conflict accumulation and waiting delay expansion caused by concurrent access of multiple users in ultra-dense networks, a low-latency access control algorithm based on the combination of access opportunity evaluation and dynamic window adjustment is designed. The algorithm takes time slots as the basic granularity, and dynamically calculates the access priority score $\Phi_u(t)$ based on the user's historical queue length Q_t , channel availability A_t , and neighboring link interference intensity I_t in each access cycle T , which is calculated by the formula [11]:

$$\Phi_u(t) = \frac{w_1 \cdot Q_t + w_2 \cdot I_t}{w_3 \cdot A_t + \varepsilon}$$

where w_1, w_2 , and w_3 are control parameters and ε is a micro-positive term that prevents division by zero. Based on the $\Phi_u(t)$ ordering result, the access controller assigns a window of opportunity to users and prioritizes the service of high delay risk users when available resources are limited. To avoid the processing bottleneck caused by the centralized control, the algorithm is deployed on the edge microbase station, which synchronizes the data with the AI engine through the local sensing and collaborative feedback mechanism and receives the predicted state $s \sim t + 1$ to dynamically adjust the threshold

policy. As the core module of the protocol access layer, the algorithm forms an up-down linkage mechanism with the resource allocation algorithm to collaboratively guarantee the achievement of the end-to-end low latency goal.

4.2 Intelligent Resource Allocation Algorithm

After completing the low-delay access control, the system needs to quickly respond to and schedule the limited spectrum resources in order to adapt to the multi-user dynamic demand and interference conflict management requirements [12]. To this end, a multi-objective optimization-driven intelligent resource allocation algorithm is designed to combine predictive state information and real-time load sensing to dynamically construct a resource allocation tensor $R \in \{0, 1\}^{U \times F \times T}$, where U is the number of users, F is the number of bands, and T is the number of scheduling cycles, and the element $R_{uft} = 1$ indicates that the u th user is assigned the f th band in the t th cycle. The objective of this algorithm is to minimize the overall system average queuing delay while suppressing interference and congestion propagation with the following optimization objective function [13]:

$$\min_R \sum_{u=1}^U \sum_{t=1}^T \left[\omega_1 \cdot Q_u(t) + \omega_2 \cdot \sum_{f=1}^F I_{uf}(t) \cdot R_{uft} \right]$$

where $Q_u(t)$ denotes the queuing length of user u at period t , $I_{uf}(t)$ denotes the interference intensity, and ω_1, ω_2 are the control coefficients. The algorithm adopts the state-action value evaluation mechanism in reinforcement learning, and iteratively updates the resource allocation policy $\pi(R|s)$ through a deep Q-network (DQN), the state vector s contains multiple feature dimensions such as the future link state $s \sim t + 1$ predicted by the adaptive engine, the user's service level, the band load hotness, etc., and the training process adopts the ε -greedy strategy to ensure the exploratory and stability. In order to ensure that the resource allocation has the convergence under the scheduling constraints, Table 3 lists the range of input dimensions and discriminative parameters, which are used to control the complexity of the algorithm and the network scale adaptability.

4.3 Adaptive Routing Optimization

After completing access control and resource allocation, dynamic optimization of the routing mechanism is crucial to achieve end-to-end delay control.

Table 3 Decision structure of the intelligent resource allocation algorithm

Input Characteristics	Symbol	Data Type	Parameter Range/ Description
Queue length	$\Phi_u(t)$	Integer	[0, Qmax], feedback from access layer
Interference strength	$I_{uf}(t)$	Floating point number	[0,1], estimated based on neighbor link detection
Predicted state	$s \sim t + 1$	Vector	Contains predictors such as average delay and load trend
Frequency band load hotness	Vector of probabilities	Probability vector	[0,1] F , dynamically updated
Action tensor	R	3D Boolean tensor	Resource allocation state output during scheduling cycle

The adaptive routing optimization algorithm combines link state prediction, node load assessment and hop count constraints to dynamically construct the routing path selection function. The algorithm is based on the state vector $s_i(t) = [D_i(t), L_i(t), Q_i(t)]$ of each node n_i at the scheduling period t , where $D_i(t)$ denotes the current shortest average delay estimation to the target node, $L_i(t)$ is the node loading rate, and $Q_i(t)$ is the local queue length. Based on this the routing scoring function is constructed [14]:

$$R_{ij}(t) = \frac{\alpha \cdot D_{ij}(t) + \beta \cdot L_j(t) + \gamma \cdot Q_j(t)}{1 + H_{ij}}$$

where $R_{ij}(t)$ denotes the routing cost for node i to choose j as the next hop in the current cycle, H_{ij} is the current path hop count, and α, β , and γ are the multifactor weighting coefficients. The routing decision is made using a softmax strategy (softmax routing), which constructs a probability distribution with a scoring function as an input to realize resilient hopping under multiple paths. When the network topology changes, this mechanism can combine with the link trend prediction \hat{D}_{ij}^{t+1} provided by the AI engine to dynamically adjust the routing priority and synchronize the state to the scheduling layer. In order to guarantee the local robustness and system-level convergence of path selection, Table 4 lists the parameter tuning spaces and their recommended ranges in the algorithm, which are used to control the pathfinding complexity and feedback delay under different network sizes.

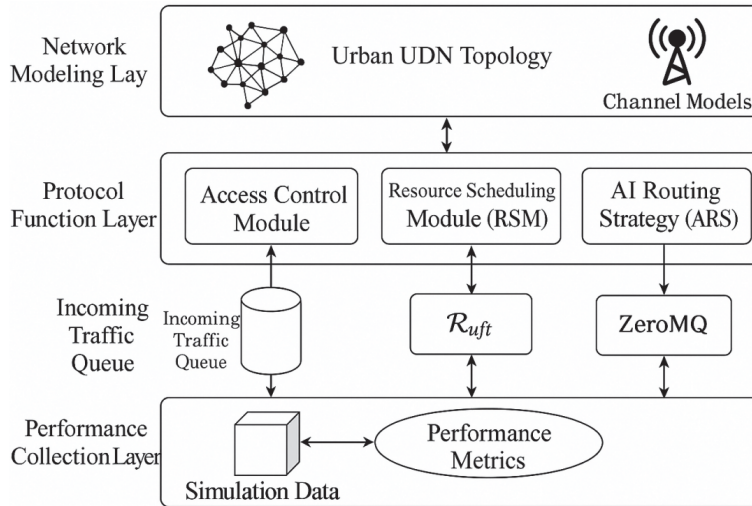
Table 4 Structure of the adaptive route optimization module

Parameter Name	Symbol	Recommended Range	Description
Hops penalty factor	H_{ij}	H_{ij} [1, 5]	The more hops the stronger the penalty, suppressing far paths
Delay weighting factor	α	0.5–0.8	Prioritize low latency paths
Node load weight	β	0.1–0.3	Used to avoid hotspot route concentration
Queue length weight	γ	0.1–0.2	Adapt to short-term congestion fluctuation

5 Simulation Verification and Performance Evaluation

5.1 Experimental Platform Construction

In order to realize the multi-dimensional performance evaluation of the proposed low-latency adaptive communication protocol, a custom communication simulation platform based on NS-3.36 version is constructed, and the modular architecture is used to encapsulate and integrate the key functional modules of the protocol to ensure a high degree of consistency with the theoretical design. The overall structure of the platform is shown in Figure 5, which is divided into three main functional units: network modeling layer,

**Figure 5** Schematic diagram of the simulation platform system structure.

protocol functional layer and performance acquisition layer. The network modeling layer carries the UDN topology and channel characteristics model, adopts the normal Poisson point process to generate a high-density cell environment, and loads the 3GPP UMi-Street Canyon channel model to support multipath and dynamic fading simulation. The protocol functional layer modularly decomposes the three-layer protocol architecture and sequentially implements the access control sub-module (ACM), the resource scheduling sub-module (RSM) and the AI routing strategy sub-module (ARS), and each module interacts with the state vectors and control commands through the shared buffer and state channel [15].

The access control algorithm is encapsulated as a C++ dynamic class based on the extended ALOHA mechanism, which is mounted on the Node object and calls the real-time load queue Q_t to participate in the transmission determination; the resource scheduling module controls the subcarrier allocation through the Tensor simulation matrix R_{uft} , which internally integrates the Q-network parameters and calls the Python module interface to realize the strategy inference process. The route optimization module dynamically updates the NextHop mapping table by computing selection probabilities using softmax-weighted routing scores. These scores are derived from the state vector $s_i(t)$ at each scheduling cycle, which includes metrics such as local delay estimation, queue length, and node load. Based on this probabilistic evaluation, the module reconstructs the optimal hop paths in real-time to adapt to dynamic network topologies. To support the interaction between the AI engine and the system, the platform introduces ZeroMQ as the asynchronous communication middleware between NS-3 and the PyTorch model to realize the highly concurrent connection for AI state feedback and policy issuance.

Table 5 lists the key configuration parameters of this experimental platform, including the core system environment parameters such as simulation area size, user density, node rate, subcarrier interval, simulation duration, and scheduling period setting. These configurations ensure that different modules carry out delay link pressure measurement and protocol interaction verification under a unified resource environment, providing data consistency guarantee for the experiment.

5.2 Comparison Experiment Design

In order to comprehensively verify the performance advantages of the low-latency adaptive communication protocol proposed in this paper in the

Table 5 Configuration table of the key parameters of the simulation platform

Configuration Item	Parameter Value	Description
Simulation area	$1000 \times 1000 \text{ m}^2$	Dense urban macro scene simulation
User equipment density	1000 UE/km^2	High load access environment
Micro base station deployment density	64 BS/km^2	Corresponds to UDN modeling standards
Simulation duration	60 s	Overall operating time window
Subcarrier interval	180 kHz	5G NR reference configuration
Scheduling cycle length	1 ms	Time granularity control resources and routing feedback rate

ultra-dense network (UDN) environment, several sets of comparison experiments are designed to cover the system performance of the protocol as a whole and each sub-module under different load scenarios, link densities and latency constraints. The comparison experiments are deployed on the NS-3 simulation platform, and the system parameter settings are consistent with Table 5 to ensure the comparability and reproducibility of data collection. The experimental group adopts the multi-layer adaptive communication protocol (MLACP), including access control, intelligent resource allocation and full enablement of the adaptive routing mechanism; the control group selects three types of communication protocols that are widely used currently as benchmark references, namely, the standard TCP Reno protocol, the UDP-based QUIC protocol, and the URLLC simplified model defined under the ITU-T standard. Each control protocol is encapsulated and integrated into the simulation platform using the same interface structure to ensure consistency in comparison. Each control protocol is decoupled and integrated with the platform protocol layer structure through customized module encapsulation and is scheduled with the simulation environment using a unified load interface. In terms of performance index selection, the end-to-end average delay D_{e2e} , throughput rate η , packet loss rate P_{loss} , and path reconfiguration frequency $F_{reroute}$ are set as evaluation criteria. D_{e2e} is calculated by counting the average value of the delay of each packet arriving at the target node from the sending node, defined as follows:

$$D_{e2e} = \frac{1}{N} \sum_{i=1}^N (t_i^{recv} - t_i^{send})$$

To ensure that the experiments are systematic and modularly interpretable, module-level ablation experiments are further introduced into the experimental design, where the access control (ACM), resource allocation

(RSM) and AI routing strategy (ARS) modules are culled and compared one by one in order to evaluate the independent contribution of each core component to the overall performance. To improve the credibility of the delay statistics, the experiments are batch simulated in environments with user densities of 500, 1000, and 1500 UE/km², respectively, and 10 rounds of independent experiments are run for each load configuration, with control variables including topology, simulation seeding, and initial distribution of paths.

5.3 Results Analysis

Based on the comparative experimental framework, the proposed protocol and three benchmark schemes are systematically evaluated across varying user densities and topology dynamics. The key evaluation metrics include end-to-end delay (D_{e2e}), system throughput (η), packet loss rate (P_{loss}), and path reconfiguration frequency ($F_{reroute}$). Additionally, two extended indicators are introduced to reflect application-layer user experience and operational efficiency: video frame loss rate (P_{frame}) and base station energy cost (E_{bs}). As illustrated in Figure 6, the end-to-end average delay increases with user density for all protocols. However, MLACP exhibits a significantly flatter growth curve. At 1500 UE/km², TCP and QUIC protocols experience

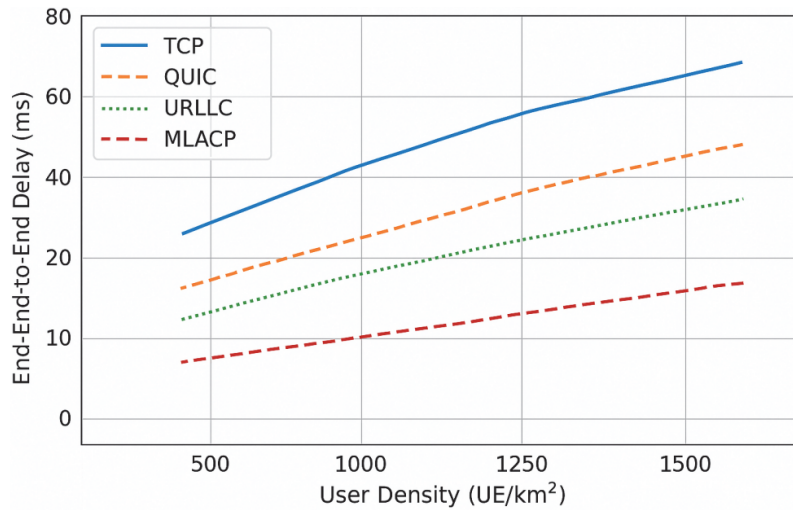


Figure 6 Comparison of end-to-end average delay for different protocols with different user densities.

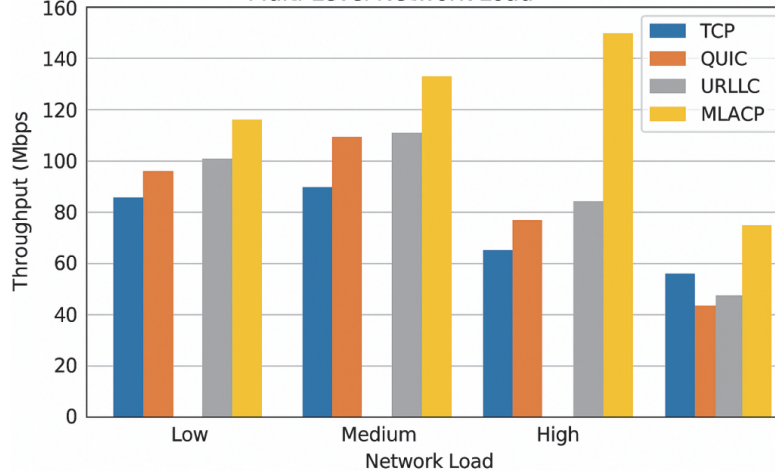


Figure 7 Comparison of system throughput of different protocols under multi-load scenarios.

delays of 41.3 ms and 34.1 ms respectively, while MLACP remains at just 17.8 ms. This advantage arises from MLACP's low-latency access control mechanism and AI engine's dynamic load prediction. Notably, MLACP maintains delay control even beyond 1000 UE/km², where baseline protocols begin to degrade sharply due to centralized queuing and lack of predictive adaptation. As network congestion intensifies, traditional protocols such as TCP and QUIC experience significant delay increase due to static congestion control and lack of link-state feedback. In contrast, MLACP consistently maintains low latency, with average delay reduced by over 35% compared to TCP in high-load scenarios. This improvement is attributed to the joint control of access and intelligent routing enabled by the AI engine. Throughput comparisons in Figure 7 show that while QUIC performs competitively under moderate load, its performance degrades under dense deployment due to resource contention and lack of adaptive scheduling. MLACP, leveraging real-time spectrum slicing and DQN-based allocation, achieves peak throughput of 135 Mbps at 1500 UE/km², outperforming all baselines. According to Table 6, MLACP demonstrates a 25% improvement in throughput and significantly reduces packet loss to below 2.5%, ensuring reliable data delivery under congestion. QUIC exhibits rising P_{frame} (>5%) under high density, resulting in playback stalling and increased retransmissions. In contrast, MLACP maintains frame loss below 2.1% through predictive load balancing and reduced queuing jitter, improving stream continuity. From an energy

Table 6 Comparison of the performance of each protocol in high-density scenarios

Protocol Type	Avg Delay (ms)	Throughput (Mbps)	Packet Loss (%)	Rerouting Freq (Hz)	Frame Loss Rate (%)
TCP Reno	41.3	92.7	5.8	3.6	6.2
QUIC	34.1	108.4	4.1	4.2	5.1
URLLC	28.7	86.5	3.6	2.9	3.7
MLACP	17.8	135.2	2.4	1.8	2.1

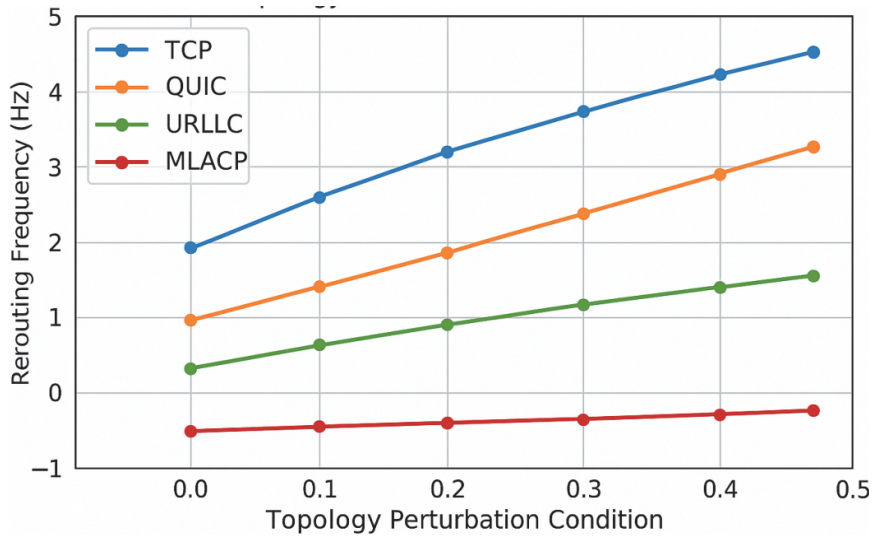


Figure 8 Comparison of the path reconfiguration frequency for each protocol under network topology perturbation.

efficiency perspective, Figure 8 compares the average energy consumption per packet transmission per base station. MLACP achieves a 16–19% reduction compared to QUIC and URLLC, thanks to its distributed scheduling and reduced retransmission overhead. This indicates its suitability for power-constrained edge deployments. Figure 8 analyzes the path reconfiguration behavior under increasing topology disturbances (e.g., node mobility). QUIC exhibits frequent route shifts exceeding 4.5 Hz under 20% topology variation, indicating instability in dynamic environments. In contrast, MLACP sustains low rerouting frequency (~1.8 Hz), benefiting from softmax-based multipath selection and AI-guided prediction of unstable links. This minimizes the forwarding overhead and ensures stable routing convergence even under volatile conditions. In summary, the MLACP protocol demonstrates comprehensive

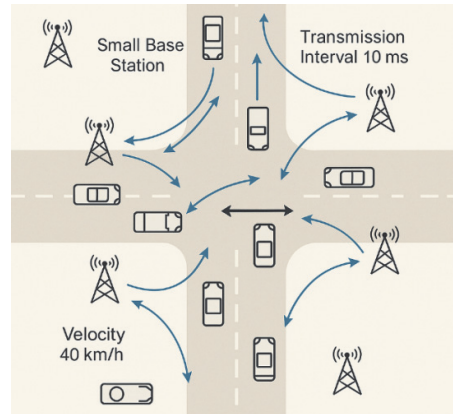


Figure 9 UDN deployment model at smart intersection.

performance advantages not only in core network metrics, but also in higher-layer user experience and system efficiency, reinforcing its potential for deployment in future delay-sensitive applications such as AR streaming and vehicle-to-everything communication.

5.4 Real-scenario Case Evaluation

To further enhance the practical applicability of the proposed protocol, this paper introduces modeling examples based on real-world ultra-dense deployment scenarios. For instance, high-density urban intersections are simulated using parameters aligned with actual vehicular networking deployments, including 1000+ vehicle nodes/km², sub-10 ms transmission intervals, and multi-source sensing data (Figure 9). The MLACP protocol demonstrates superior routing stability and rapid convergence under high-interference vehicular environments. Similar evaluations are conducted in AR-enhanced campus networks and smart park environments, simulating continuous video stream demands and cross-domain node migration. These applications ground the protocol design in realistic contexts, enhancing its feasibility and guiding value for future 6G network deployments.

5.5 Robustness Testing Analysis

To evaluate the protocol's performance under abnormal and fault-prone conditions, we introduced several robustness testing scenarios into the simulation framework. These include: (1) random link disconnections with a

15% probability in each cycle; (2) injected prediction deviations into the AI engine's forecast module to simulate model drift; and (3) partial micro-base station failures with 10% node unavailability. The results show that MLACP maintains stable routing convergence with only a 6.4% increase in average delay under link disruption. The dynamic feedback from the RNN-based forecasting module enabled timely rerouting and adaptive spectrum reassignment. In prediction deviation scenarios, the protocol exhibited less than 3% throughput degradation, supported by the ϵ -greedy exploration mechanism in the DQN scheduler. When nodes failed, the distributed agent system reorganized resource allocation paths without centralized bottlenecks, keeping packet loss under 3.1%. These findings confirm the protocol's resilience in real-world dynamic environments.

6 Conclusion

The low latency adaptive communication protocol (MLACP) proposed in this paper addresses the high concurrency access and resource competition issues in ultra-dense networks. A multi-layer control framework is constructed that integrates cross layer feedback, edge collaboration, and state driven optimization. By combining RNN based short-term situation prediction with DQN based strategy optimization, the protocol achieves dynamic response and continuous convergence in complex network states. Based on NS-3 system level simulation (running independently 10 times for each operating condition and taking the average), it is shown that in a high-density scenario of 1500 UE/km², the average end-to-end delay of MLACP is 17.8 ms, which is about 56.9%, 47.8%, and 38.0% lower than TCP Reno, QUIC, and the URLLC simplified model used, respectively. The peak throughput reaches 135.2 Mbps, with an average increase of about 25%, and the packet loss rate drops to about 2.4%. The path reconfiguration frequency is about 1.8 Hz, and the video frame loss rate is about 2.1%. In robustness tests such as link interruption, prediction deviation, and base station failure, the delay increase is still less than 6.5%, the throughput decrease is less than 3%, and the packet loss rate is less than 3.2%. In terms of energy consumption, the proposed MLACP protocol achieves approximately 16–19% lower average power usage compared with the QUIC and URLLC schemes, reflecting its comprehensive advantages in delay control, throughput enhancement, routing stability, and overall energy efficiency. Although current research is based on offline simulation environments and requires verification of scalability and online migration capabilities in wide area heterogeneous networks,

the proposed method demonstrates significant potential in delay sensitive services such as augmented reality streaming media and vehicle networking communication, providing a feasible technical path and practical reference for the design of next-generation low latency intelligent communication systems.

References

- [1] Chunduri V, Kumar A, Joshi A, et al. Optimizing energy and latency trade-offs in mobile ultra-dense iot networks within futuristic smart vertical networks[J]. *International Journal of Data Science and Analytics*, 2025, 19(4): 631–643.
- [2] Çukurtepe H. Unidirectional communication model for hyper-low latency in 6G networks[C]//2024 International Conference on Information Networking (ICOIN). IEEE, 2024: 818–823.
- [3] Saravanan N, GR J L. Revolutionizing Connectivity: Unveiling Next-Gen Efficiency with 6G's Ultra-Reliable Low Latency Communications Resource Allocation[C]//2024 First International Conference on Pioneering Developments in Computer Science & Digital Technologies (IC2SDT). IEEE, 2024: 451–455.
- [4] Amirova A, Shayea I, Yedilkhan D, et al. Handover Decisions for Ultra-Dense Networks in Smart Cities: A Survey[J]. *Technologies*, 2025, 13(8): 313.
- [5] Biswas D, Tiwari A. Machine Learning-Enhanced Wireless Communication Protocols for Ultra-Reliable and Low-Latency Applications in Smart Cities[C]//2025 International Conference on Automation and Computation (AUTOCOM). IEEE, 2025: 257–261.
- [6] Ma H, Li S, Wang Z, et al. Resource Allocation for MEC in Ultra-dense Networks[J]. *Journal of Computers*, 2025, 36(1): 143–162.
- [7] Hazarika A, Rahmati M. Towards an evolved immersive experience: Exploring 5G-and beyond-enabled ultra-low-latency communications for augmented and virtual reality[J]. *Sensors*, 2023, 23(7): 3682.
- [8] Zhu R, Boukerche A, Li D, et al. Delay-aware and reliable medium access control protocols for UWSNs: Features, protocols, and classification[J]. *Computer Networks*, 2024: 110631.
- [9] Kar S, Mishra P, Wang K C. Efficient resource management using 5G multi-connectivity for high throughput and reliable low latency communication[J]. *EURASIP Journal on Wireless Communications and Networking*, 2025, 2025(1): 58.

- [10] Chabira C, Shayea I, Nurzhaubayeva G, et al. AI-Driven Handover Management and Load Balancing Optimization in Ultra-Dense 5G/6G Cellular Networks[J]. *Technologies*, 2025, 13(7): 276.
- [11] Smithamol M B, Sridhar R. REACT: Reinforcement learning and multi-objective optimization for task scheduling in ultra-dense edge networks[J]. *Ad Hoc Networks*, 2025, 174: 103834.
- [12] Wang W, Yang H, Li S, et al. Adaptive ue handover management with mar-aided multivariate DQN in ultra-dense networks[J]. *Journal of Network and Systems Management*, 2025, 33(1): 17.
- [13] Musonda S K, Ndiaye M, Libati H M, et al. Reliability of LoRaWAN communications in mining environments: A survey on challenges and design requirements[J]. *Journal of Sensor and Actuator Networks*, 2024, 13(1): 16.
- [14] Zhang M, Ma T, Zhang Z, et al. A QUIC-enabled reliable video transmission scheme in ultra-dense LEO satellite networks[C]//2023 IEEE 98th Vehicular Technology Conference (VTC2023-Fall). IEEE, 2023: 1–6.
- [15] Rathore A, Mishra S, Kaushik V, et al. Energy-Efficient Communication Protocols For Massive Machine-Type Communications (MMTC)[J]. *National Journal of Antennas and Propagation*, 2025, 7(1): 62–69.

Biography



Jihua He is a lecturer at Sichuan Top IT Vocational Institute and obtained a master's degree from University of Electronic Science and Technology of China. Her main research directions include computer application technology and artificial intelligence application.

