Original Article	*Curr. Appl. Sci.*, **2025**, *3*(2):119-136 DOI: 10.22034/cas.2026.245730	Current Applied Sciences

AI-Driven Predictive Handover in Ultra-Dense 6G Networks: A Federated Graph Reinforcement Learning Approach

Hamed Fehri[1]*, Mostafa Monemizadeh[2]

[1] *Department of Electrical Engineering, Faculty of Engineering, University of Zabol, Zabol, Iran*

[2] *Department of Electrical Engineering, University of Neyshabur, Neyshabur, Iran*

Corresponding author's e-mail: *hamedfehri@uoz.ac.ir*

Article Information

Received: 16 July 2025
Revised: 01 September 2025
Accepted: 03 September 2025
Published online: 21 September 2025

Keywords

6G
Handover optimization
Ultra-dense networks
Reinforcement learning
Federated learning
Graph neural networks
Mobility prediction
Privacy-preserving AI
URLLC

Abstract

Traditional handover (HO) methods encounter critical challenges in satisfying the Tbit/s throughput and sub-millisecond latency requirements of ultra-dense 6G networks. Inability to achieve ultra-reliable and low-latency communication (URLLC), excessive overhead of signaling and ping-pong effects are critical issues arising from applying traditional 3GPP-compliant HO methods in ultra-dense networks (UDNs) with device densities exceeding 700 UEs/km². To tackle these limitations, this paper proposes a new AI-based predictive HO architecture for pre-planning HO timing and target BS in advance up to 500 ms. Our proposed architecture is a hybrid one based on federated graph reinforcement learning (FGRL) and can provide an intelligent and proactive mobility management while preserving privacy in 6G networks. In the proposed FGRL method, federated learning addresses the privacy concerns, graph neural networks model the spatial relationships between BSs, and Multi Agent Deep Q Networks (MADQN) provide distributed decision making. To assess the performance of the proposed method, extensive simulations are conducted showing that the proposed method significantly improves HO latency, ping-pong rates, and throughput compared to 3GPP-standard-compliant A3/A6 event-based algorithms. We also compare centralized, distributed, and federated designs and demonstrate that the proposed federated design achieves competitive performance to the centralized design while preserving user privacy and considerably decreasing uplink overhead.

1. Introduction

Sixth-generation (6G) wireless networks are being designed to deliver a transformative leap in connectivity, targeting peak data rates in the terabit/s regime, end-to-end latencies as low as 0.1 ms, and device densities

exceeding 10^7 devices/km² [1, 2]. Beyond sheer capacity, 6G is expected to enable entirely new application classes – holographic telepresence, multisensory extended reality, pervasive digital twins, unmanned aerial vehicle (UAV) swarms, and autonomous vehicular networks – each imposing stringent demands on reliability, mobility support, and latency [2, 3]. A defining characteristic of 6G is the shift from reactive to proactive, intelligent network operations. Cloud-native, AI-driven architectures are envisioned where machine learning is embedded into every control loop, from the physical layer to resource management, enabling networks that self-optimize, self-heal, and adapt in real time [3]. In ultra-dense network (UDN) deployments, where small-cell densities may reach 40 cells/km² or more at mmWave frequencies, traditional handover mechanisms designed for sparser 4G/5G layouts break down: excessive measurement signaling, ping-pong effects, and handover failures become prevalent, severely jeopardizing ultra-reliable low-latency communications. The need to maintain seamless connectivity under high mobility --- a challenge also emphasized by recent waveform studies highlighting the vulnerability of conventional schemes to Doppler spread [4] --- makes intelligent mobility management a cornerstone of 6G. In this context, adopting AI-driven, predictive handover is not merely an optimization but a fundamental requirement to unlock the full potential of UDNs in 6G.

Recently, researchers [5] have supported the idea that AI/ML approaches could manage handover based on real-time network conditions prediction, user mobility, and interference levels. Reinforcement learning algorithms, in particular, have emerged as key enablers for optimizing handover decisions by learning optimal base station selection policies through continuous interaction with the network environment [6]. Furthermore, exploiting historical data proves useful in deep learning results that access user-mobility predictive profiles and facilitate subsequent advanced decisions for handover [7]. Yet thus far, all theoretically offered solutions require centralized AI development, which adds challenges of privacy as well as latency since any user/historical data must first be sent to that central processing location before anything can be completed. Thus, we address the following research problems in AI-driven handover for 6G UDNs in this paper:

1. **Spatial-temporal modeling challenges for handover policies**: Most reinforcement learning works assume a flat topology of the network; however, spatial relationships play an important role in base station (BS) communication and thus should be included for optimal handover performance.
2. **Centralized versus distributed solutions:** No known implementation exists that simultaneously achieves high performance, low latency, and privacy preservation.
3. **Privacy-preserving methods:** Training centralized AI with user-supplied data inherently introduces privacy challenges, which have yet to be suitably addressed.

To address these problems, this paper introduces a Federated Graph Reinforcement Learning (FGRL) system for predicting handover in 6G UDNs. The main contributions of this paper are:

1) We design a Graph Attention Network (GAT)-based spatial model to understand the relationships among base stations. This enables more accurate handover prediction based on network topology.

2) We deploy a multiagent reinforcement learning method at the network edge, maintaining coordination across neighbouring cells. This enables distributed decision-making within each cell.

3) We develop a federated learning (FL) method to protect privacy. Base stations can learn a general handover prediction model without sharing user data. This addresses privacy issues that can arise with accurate prediction models.

Our simulations demonstrate that the FGRL framework outperforms the traditional 3GPP handover method

and other state-of-the-art AI approaches in terms of handover delay, ping-pong rate, and network throughput. The rest of the paper is organised as follows. Section II reviews related work, Section III describes the system model, Section IV details the FGRL framework, Section V presents the performance evaluation, Section VI discusses the results and practical implications, and Section VII concludes the paper.

2. Related Work

2.1 *6G networks: requirements, AI-native intelligence, and mobility challenges*

Comprehensive 6G roadmaps, such as the extensive survey by Wang et al. [2], establish a clear set of key performance indicators (KPIs) that go well beyond 5G capabilities: peak data rates of 1 Tbps, user-experienced rates of 10 Gbps, air-interface latencies below 0.1 ms, connection densities up to 10^8 devices/km², and mobility support exceeding 1000 km/h. These KPIs are driven by ambitious application scenarios including immersive extended reality, holographic communication, wireless brain-computer interfaces, and integrated sensing and communication (ISAC) [2]. Importantly, the 6G vision is not confined to terrestrial communication; it extends to space-air-ground-sea integrated networks, forming a three-dimensional, multi-layer architecture that imposes unprecedented mobility management challenges [2, 3].

A central theme in the 6G literature is the concept of AI-native networking. Yang et al. [3] propose an AI-enabled intelligent architecture that layers sensing, data mining, intelligent control, and smart applications, allowing 6G systems to perform knowledge discovery, automated resource allocation, and self-optimization. In this architecture, AI techniques – deep reinforcement learning (DRL), deep learning, and supervised/unsupervised learning – are directly applied to mobility and handover management, spectrum management, and mobile edge computing. The work demonstrates that DRL, in particular, can learn optimal handover policies by interacting with highly dynamic environments, including UAV-enabled networks, thereby reducing handover failures and latency while preserving reliable connectivity [3]. However, that study stops short of addressing the spatial interdependencies among base stations or the privacy implications of centralized training.

In parallel, physical-layer research highlights the importance of robust transmission in high-mobility scenarios. Sheikh-Hosseini et al. [4] compare OFDM, GFDM, and OTFS modulation under doubly-dispersive fading channels with impulsive noise, showing that traditional waveforms such as OFDM degrade sharply under Doppler spread, whereas OTFS excels in such conditions. This finding reinforces the necessity of mobility-aware designs across all protocol layers, including handover. While AI-based approaches are increasingly proposed to address mobility management in dense networks, most existing solutions either ignore the spatial graph structure of base station placements, rely on centralized data processing with associated privacy risks, or have not been validated in the ultra-dense, multi-tier settings envisioned for 6G. These gaps directly motivate our FGRL framework, which combines (i) graph neural networks to model the spatial relationships among cells, (ii) multi-agent deep reinforcement learning for distributed decision-making at the edge, and (iii) FL to collaboratively train models without exposing raw user data.

2.2 *Traditional handover methods*

Conventional handover methods adjust parameters like hysteresis margins, time-to-trigger, and offset values to limit unnecessary handovers [8]. Alhammadi et al. proposed a smart, coordinated, self-adjusting handover scheme for mixed 4G/5G networks. It uses fuzzy logic to change these parameters based on the state of the network [9].

Hwang et al. developed a flexible handover decision tool using fuzzy logic, specifically for 5G UDNs [10]. While these methods improve upon basic 3GPP procedures, they remain reactive rather than predictive and struggle with the density of 6G UDNs.

2.3 *AI-based handover*

Current studies use AI to tackle handover problems in dense networks. Zhang et al. created a handover method for LTE-R systems that estimates handover settings using clustering [7]. Mal et al. addressed mobility robustness through radio link failure prediction [11]. Chaudhuri et al. introduced a self-organizing method for handover performance optimization in LTE-Advanced networks [12].

Reinforcement learning has become a main way to handle handovers. Guo et al. applied multiagent DRL to optimize handover control and power assignment at the same time [6]. Wang et al. created a satellite handover plan for satellite communication using DRL [13]. Castro-Hernandez and Paranjape adjusted handover settings for indoor LTE/LTE A using reinforcement learning [14]. Hegazy et al. applied fuzzy Q-learning to LTE networks to make handovers based on users acts [15]. Recent work on learning-based network control by Tan et al. [16] showed that machine learning can control interference in dense networks by using coexistence plans for LAA-LTE networks.

2.4 *Network management with FL*

The FL is getting looked at as a way to train AI while still saving on privacy, instead of having everything in one spot, namely centralized AI. Liu et al. studied FL for 6G communication, noting that it can predict movement while saving privacy [17]. Al-Quraan et al. studied edge-native intelligence for 6G communication powered by FL and identified the main trends and challenges [18]. Recent work by Noman et al. showed that FL can be applied to handover management in UDNs, which seems to improve throughput and cut down on unneeded handovers [19].

2.5 *Research gaps*

These improvements have not solved all existing problems in the field. Some gaps are as follows:

1. Almost all AI-based handover methods do not consider the spatial relations between base stations. This matter is important in UDNs because the cell boundaries change frequently.
2. The centralized AI architectures introduce additional latency and privacy concerns that are not compatible with 6G requirements.
3. The trade-offs between centralized, distributed, and federated AI implementations for handover management have not been thoroughly investigated.

Based on the above, our work tries to fix these gaps through a novel FGRL framework specifically designed for 6G UDNs.

3. System Model

3.1 *Network architecture*

We consider an ultra-dense 6G network consisting of three tiers:

1. Macro-tier: Traditional macro base stations providing wide-area coverage

2. Small-cell tier: Dense deployment of small cells (40 cells/km²) operating at 28 GHz (mmWave)
3. Edge layer: Mobile Edge Computing (MEC) servers co-located with small cells

The network serves 700 UEs/km² with heterogeneous mobility patterns (pedestrian: 1-2 m/s, vehicular: 10-30 m/s). Each UE periodically reports channel quality indicators (CQIs), reference signal received power (RSRP), and reference signal received quality (RSRQ) to the serving base station. Figure 1 provides an overview of this three-tier architecture and the placement of the FGRL functional blocks. As illustrated in the figure, UE measurement reports (CQI, RSRP, RSRQ) are first collected by the serving small-cell BS and processed at the co-located MEC server. The GAT module extracts spatial features from adjacent cells, and the MADQN agent makes the actual handover decision locally. All measurement data remains at the edge; only the trained model parameters are periodically uploaded to the NWDAF for federated aggregation. Once the global model is computed, it is broadcast back to the MEC servers, where the next round of local training continues.

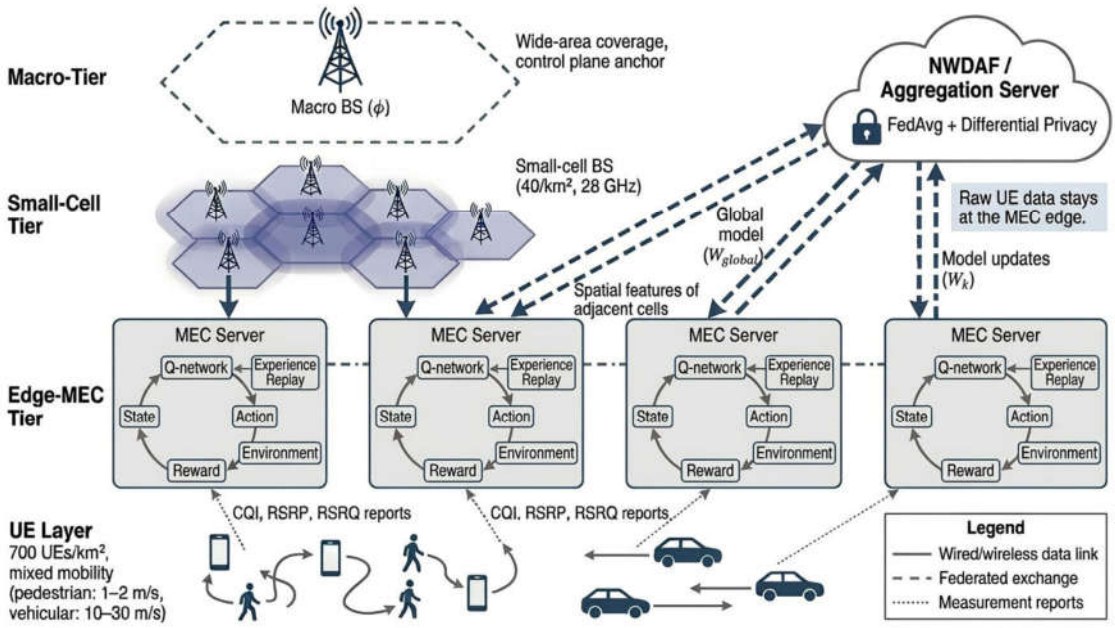

Figure 1. Proposed three-tier network architecture for ultra-dense 6G handover management, illustrating the placement of the FGRL components. The macro-tier provides wide-area coverage, the small-cell tier handles dense mmWave access, and the edge tier hosts the GAT-based spatial model and the MADQN agents. Federated aggregation is performed at the NWDAF in the core network, ensuring privacy-preserving collaborative learning.

3.2 *Handover process model*

The handover process in our system consists of four phases:

1. **Measurement:** UEs measure signal quality from neighboring cells, continuously.
2. **Decision:** The network determines whether a handover is necessary.
3. **Execution:** The handover command is issued and executed.
4. **Verification:** Post-handover performance is monitored.

Traditional 3GPP handover decisions are based on the A3 event (neighbor cell becomes offset better than serving cell) with hysteresis and time-to-trigger parameters. In contrast, our predictive framework anticipates handover needs before signal degradation occurs.

The detailed interplay among the graph spatial model, the multi-agent learning, and the federated aggregation is outlined in Section 4.5, where the complete information flow during training and inference is presented.

3.3 *Problem formulation*

Let $G = (V, E)$ define the network topology where V is the set of base stations and E shows which cells are next to each other. Each UE u has a state $\boldsymbol{s}_u(t) = [x_u(t), y_u(t), v_u(t), \theta_u(t), \text{RSRP}_u(t)]$ that describes its position, velocity, direction, and signal quality at time t.

We can model the handover decision as a Markov Decision Process (MDP) $\langle \mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \gamma \rangle$ where:

- $\mathcal{S}$ is the state space which includes UE mobility patterns and network conditions
- $\mathcal{A}$ is the action space consisting of potential target base stations
- $\mathcal{P}(s'|s, a)$ is the state transition probability
- $\mathcal{R}(s, a)$ is the reward function which is balancing throughput, latency, and handover frequency
- γ is the discount factor

The objective is to find a policy $\pi^*(a|s)$ that maximizes the expected cumulative reward $E[\sum_{t=0}^{\infty} \gamma^t \mathcal{R}(s_t, a_t)]$.

Notation convention: Throughout the paper, bold uppercase letters denote matrices, bold lowercase letters denote vectors, and italic symbols denote scalar quantities.

4. Proposed FGRL Framework

4.1 *Movement behavior function*

Our FGRL framework contains three main components:

1. **The Graph Spatial Modeling Layer**: employs GATs to detect how base stations relate to each other.
2. **The Distributed Reinforcement Learning Layer**: uses Multiagent Deep Q-Networks (MADQN) to enable localized decision-making.
3. **The FL Orchestration Layer**: This layer coordinates updates across base stations while protecting user privacy.

The system operates through base station data processing while exchanging model parameters only. The method decreases both the amount of signaling data and protects user privacy. The overall pipeline is illustrated in Figure 2, which depicts the sequential interplay among the graph spatial model, the distributed reinforcement learning agents, the federated aggregation cycle, and the predictive handover mechanism.

4.2 *Graph spatial modeling*

In order to address the spatial relationships between base stations, the network topology in UDNs is represented as a graph structure. Thus, we have:

- Nodes $v_i \in V$ represent base stations
- Edges $e_{ij} \in E$ represent adjacency relationships (cells with overlapping coverage)

The adjacency matrix $\boldsymbol{A}$ is defined as:

$$\boldsymbol{A}_{ij} = \begin{cases} 1 & \text{if cells } i \text{ and } j \text{ have overlapping coverage} \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

Let $\mathcal{N}_i = \{j \in V \mid e_{ij} \in E\}$ denote the set of neighbouring cells of BS i. The GAT layer computes attention coefficients between neighbouring cells as follows:

$$\alpha_{ij} = \frac{\exp(\text{LeakyReLU}(\boldsymbol{a}^T[\boldsymbol{W}\boldsymbol{h}_i \| \boldsymbol{W}\boldsymbol{h}_j]))}{\sum_{k\in\mathcal{N}_i}\exp(\text{LeakyReLU}(\boldsymbol{a}^T[\boldsymbol{W}\boldsymbol{h}_i \| \boldsymbol{W}\boldsymbol{h}_k]))} \quad (2)$$

The model learns to identify important neighboring cells through this mechanism. In (2), α_{ij} is the attention coefficient between base stations i and j; $\mathcal{N}_i$ is the set of neighbours of i; $\mathbf{h}_i, \mathbf{h}_j$ are the feature vectors of the two cells; $\mathbf{W}$ is a shared trainable weight matrix; and $\mathbf{a}$ is the trainable attention vector. This graph embedding stage corresponds to the spatial modeling block in Fig. 2. This allows the model to learn which neighboring cells are most relevant for handover decisions.

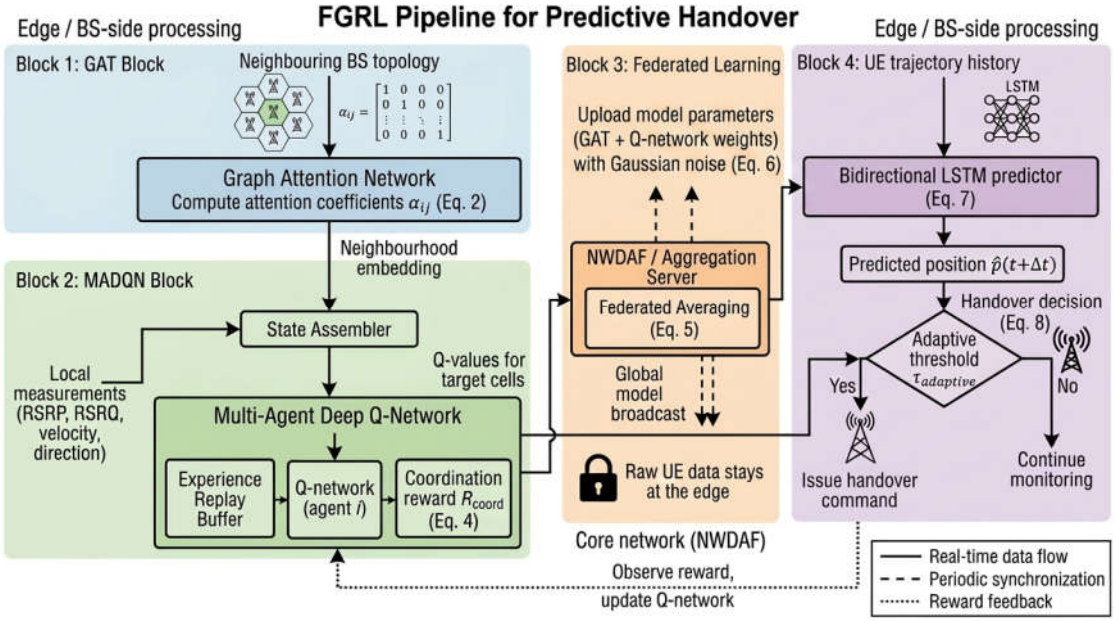

Figure 2. End-to-end operational flow of the FGRL framework. The diagram shows the four main stages: (i) spatial graph embedding via the GAT, (ii) multi-agent Q-learning with inter-agent coordination, (iii) periodic federated aggregation with differential privacy, and (iv) LSTM-based mobility prediction and proactive handover triggering. Solid lines indicate real-time data flow, dashed lines represent periodic federated communication, and dotted lines show the reinforcement learning feedback loop.

4.3 *Multiagent reinforcement learning*

Each base station functions as an independent agent with its own Q-network in our distributed MADQN architecture. The state representation for agent *i* includes:

- Local UE measurements (RSRP, RSRQ, velocity, direction)
- Network load information (traffic volume, resource utilization)
- Graph-embedded features from neighboring cells

Agent i performs Q-learning updates using the following rule:

$$Q_i(s,a) \leftarrow Q_i(s,a) + \alpha[r + \gamma \max_{a'} Q_i(s',a') - Q_i(s,a)] \quad (3)$$

Here, $Q_i(s,a)$ is the action-value function of agent i; $\alpha \in (0,1]$ is the learning rate; r is the immediate reward; $\gamma \in (0,1)$ is the discount factor; s' is the next state; and a' is the action evaluated in that next state. The reward function includes a coordination term which promotes neighboring agents to work together, as follows:

$$R_{total} = R_{local} + \lambda \sum_{j\in\mathcal{N}_i} R_{coord}(i,j) \quad (4)$$

R_{total} is the total reward, R_{local} is the locally observed performance reward, and $R_{coord}(i,j)$ is the coordination term. $\mathcal{N}_i$ is the neighbour set of agent i. The function $R_{coord}(i,j)$ evaluates the consistency of handover decisions made by adjacent cells. The hyper-parameter $\lambda > 0$ controls the trade-off between local reward and inter-agent coordination. The distributed Q-learning process is illustrated as the second stage of the pipeline in Figure 2.

4.4 *FL implementation*

The FL implementation follows this sequence of operations:

1. Local Training: Base stations gather UE measurement data to train their Q-networks through experience replay.
2. Model Parameterization: Base stations distribute model parameters (GAT and Q-network weight values) instead of sharing raw data.
3. Federated Aggregation: The NWDAF serves as a federated server to combine model updates from participating base stations through Federated Averaging (FedAvg):

$$\boldsymbol{W_{global}} = \sum_{k=1}^{K} \frac{\boldsymbol{n_k}}{\boldsymbol{n}} \boldsymbol{W_k} \tag{5}$$

where $\boldsymbol{K}$ is the number of participating base stations, $\mathbf{W}_{\boldsymbol{k}}$ denotes the local model parameters of BS $\boldsymbol{k}$, $\mathbf{W}_{\text{global}}$ is the aggregated global model, $\boldsymbol{n_k}$ is the number of training samples at BS $\boldsymbol{k}$, and $\boldsymbol{n}$ is the total number of samples across all BSs.

4. Differential Privacy: We enhance privacy by adding Gaussian noise to model updates:

$$\widetilde{\boldsymbol{W}}_{\boldsymbol{k}} = \boldsymbol{W_k} + \boldsymbol{\mathcal{N}}(\mathbf{0}, \boldsymbol{\sigma^2 I}) \tag{6}$$

where $\bar{\mathbf{W}}_{\boldsymbol{k}}$ is the privacy-preserving model update, $\mathbf{W}_{\boldsymbol{k}}$ is the original local model, $\boldsymbol{\sigma}$ is the standard deviation of the noise, $\mathbf{I}$ is the identity matrix, and $\boldsymbol{\mathcal{N}}(\mathbf{0}, \boldsymbol{\sigma^2}\mathbf{I})$ denotes a zero-mean Gaussian distribution with covariance $\boldsymbol{\sigma^2}\mathbf{I}$. The periodic federated aggregation and model broadcast are represented as the third stage in Figure 2.

5. Model Distribution: The updated global model is distributed to base stations for their upcoming training cycle.

This approach protects user mobility data at the network edge while enabling collaborative learning. Further communication-overhead reduction is possible through vector quantization as described by Hassanpour et al. which enhances edge node information transfer [20].

4.5 *Component interaction and overall workflow*

The FGRL framework is not a simple concatenation of independent modules; it relies on a tightly coupled information flow among the GAT, the Multiagent Deep Q-Networks (MADQN), and the FL orchestration layer. This subsection clarifies how these three components interact during both the learning phase and the handover execution phase.

4.5.1 Interaction during local training:

At each base station (BS) i, the state representation of the local agent is enriched by graph-level information. Specifically, the GAT layer processes the feature vectors $\boldsymbol{h}_j^{(t)}$ of all neighbouring cells $j \in \mathcal{N}_i$ at time t and computes the attention coefficients α_{ij} (Eq. (2)). The resulting neighbourhood embedding is concatenated with the local measurements (RSRP, RSRQ, UE velocity, etc.) to form the state vector that is fed into the agent's

Q-network. Thus the MADQN agent implicitly accounts for spatial cell relationships when evaluating handover actions.

The reward function (Eq. (4)) further reinforces inter-cell cooperation by including the coordination term $R_{coord}(i,j)$. This term compares the handover decisions of adjacent agents; if two neighbouring cells simultaneously decide to hand over the same UE or make contradictory selections, a penalty is applied, encouraging consensus.

4.5.2 Federated aggregation cycle:

All BSs periodically upload their local model parameters to the NWDAF, which acts as the federated aggregation server. These parameters include both the GAT weight matrix $\boldsymbol{W}$ and the Q-network weights. The server performs Federated Averaging (FedAvg) according to Eq. (5) and broadcasts the updated global model back to the BSs. Before uploading, each BS adds Gaussian noise to its updates (Eq. (6)) to satisfy differential privacy requirements. By separating the GAT and Q-network weight tensors during aggregation, the system preserves the spatial reasoning capabilities learned by the graph layer while still benefiting from global knowledge.

4.5.3 Inference and handover triggering:

During the operational phase, the local BS continuously monitors UE trajectories. The LSTM-based mobility predictor (Eq. (7)) forecasts the UE's position $\hat{\boldsymbol{p}}(t+\Delta t)$. The GAT-enhanced state, computed in real time, is processed by the latest version of the Q-network. A handover is initiated when the Q-value ratio (Eq. (8)) exceeds the adaptive threshold $\tau_{adaptive}$, ensuring that the decision is both spatially informed and robust to local network conditions. Because the models are updated periodically via FL, they remain aligned with global mobility patterns while preserving the privacy of individual users. Table 1 summarizes the input–output relationships among the four main components and their roles in the overall workflow.

Table 1. Interaction matrix of FGRL components

Component	Function	Input	Output
GAT layer (local)	Encodes cell topology, computes attention coefficients	BS feature vectors h_j of neighbours $j \in \mathcal{N}_i$	Attention coefficients α_{ij}; neighbourhood embedding
MADQN agent (local)	Learns and selects handover actions	State = [local meas., UE parameters, graph embedding]; reward signal (Eq. (4))	Q-values for target cell actions; handover decision (via Eq. (8))
Federated server (NWDAF)	Aggregates model parameters, enforces privacy	Local model updates ($\boldsymbol{W}_k$) from BSs	Global model ($\boldsymbol{W}_{global}$) distributed to all BSs
Predictive trigger	Forecasts UE position and issues early handover	Historical UE trajectory (LSTM input); Q-values from MADQN	Handover command when $Q_{target}/Q_{current} > \tau_{adaptive}$

Algorithm 1 formalizes the overall training and inference procedure of the FGRL framework.

Algorithm 1 FGRL Training and Handover Inference

Initialize: GAT parameters $\mathbf{W}, \mathbf{a}$, local Q-network $\boldsymbol{\theta}_i$, replay buffer $\mathcal{D}_i$, global model $\mathbf{W}_{\text{global}}$, LSTM predictor f_{LSTM}
for each base station i in parallel **do**
  Collect per-UE measurements (RSRP, RSRQ, velocity, direction)
  Compute attention coefficients α_{ij} (Eq. 2) and neighbourhood embedding
  for each UE **do**
    Predict future position $\hat{\mathbf{p}}(t + \Delta t)$ via Bi-LSTM (Eq. 7)
    Assemble state vector $\mathbf{s}$
    Select action a via ϵ-greedy on $Q_i(\mathbf{s}, a)$
    Execute a, observe reward R_{total} (Eq. 4)
    Store $(\mathbf{s}, a, r, \mathbf{s}')$ in $\mathcal{D}_i$
  end for
  Every N_{train} steps:
    Sample mini-batch of size B from $\mathcal{D}_i$
    Update $\boldsymbol{\theta}_i$ by minimising TD error (Eq. 3)
  Every F local updates:
    Upload $\boldsymbol{\theta}_i$ with Gaussian noise (Eq. 6) to NWDAF
end for
At NWDAF (periodic):
Receive local parameters, perform FedAvg (Eq. 5)
Broadcast $\mathbf{W}_{\text{global}}$ to all BSs
Handover execution (real-time):
if $\frac{Q_{\text{target}} - Q_{\text{current}}}{Q_{\text{current}}} > \tau_{\text{adaptive}}$ **then**
  Trigger handover
end if

4.5.4 Illustrative example:

Consider two neighbouring small cells A and B, with a vehicular UE moving from A towards B. The GAT layer detects a strong spatial correlation between the two cells (high α_{AB}) due to their overlapping coverage. Agent A's state vector is thus augmented with features of cell B. As the UE moves, the Q-network learns, through experience replay, that transferring the UE to B before the signal degrades yields a higher cumulative reward. Meanwhile, agent B's own Q-network is similarly biased, and the coordination term R_{coord} penalises any inconsistent actions (e.g., both cells trying to hand over or both refusing). During the federated cycle, both BSs contribute their locally improved policies to the global model, which is then redistributed, thereby spreading the learned mobility patterns across the entire network without revealing individual trajectories.

This tight integration ensures that the spatial awareness provided by the GAT, the distributed decision-making capability of the MADQN, and the privacy-preserving coordination of FL work in concert to deliver proactive, reliable handover in ultra-dense 6G environments. During system operation, the GAT embedding, LSTM trajectory prediction, and Q-network inference run continuously in real time for each active UE. In contrast, local Q-network updates and the federated aggregation cycle are executed periodically in the background, on a slower timescale.

4.6 *Predictive handover triggering*

The framework predicts handover requirements up to 500 ms in advance. The mechanism is different from traditional threshold-based triggering and acts as the following two steps:

1. **Mobility Pattern Prediction**: Let Δt be the desired prediction horizon (e.g., 500 ms) and τ the length of the past trajectory segment used as input. A bidirectional LSTM processes historical UE trajectory data to predict future positions:

$$\widehat{\boldsymbol{p}}(t + \Delta t) = f_{LSTM}(\boldsymbol{p}(t - \tau), \ldots, \boldsymbol{p}(t)) \tag{7}$$

$\hat{\mathbf{p}}(t+\Delta t)$ is the predicted UE position after horizon Δt; $\mathbf{p}(t)$ is the actual UE position at time t; and $f_{\text{LSTM}}(\cdot)$ is the bidirectional LSTM prediction function.

2. **Handover Readiness Assessment**: The GAT-enhanced MADQN evaluates whether a handover would improve network performance based on predicted position and current network conditions.

The handover is triggered when:

$$\frac{Q_{target}-Q_{current}}{Q_{current}} > \tau_{adaptive} \tag{8}$$

Q_{target} is the estimated Q-value of the candidate target cell, $Q_{current}$ is the Q-value of the current serving cell, and $\tau_{adaptive}$ is the adaptive triggering threshold. The adaptive parameter $\tau_{adaptive}$ adjusts dynamically according to network congestion levels and UE movement patterns.

4.7 *Time complexity analysis*

To demonstrate the practicality of the FGRL framework in ultra dense deployments, we analyse the computational complexity of its main components during both training and inference. The analysis is presented per base station (BS) and, where relevant, for the centralised federated aggregation server. The notation follows the definitions introduced in sections 3 and 4.

4.7.1 Graph attention network (GAT):

Each BS i runs a GAT layer to compute attention coefficients α_{ij} with all neighbours $j \in \mathcal{N}_i$ (Eq. (2)). The feature vector of a BS has dimension D. The computation involves a linear transformation $\boldsymbol{W}\boldsymbol{h}_j$ for the node itself and each neighbour, a vector dot product with the attention vector a, and a softmax over the neighbourhood. The per-BS time complexity therefore is

$$\mathcal{C}_{\text{GAT}} = O(|\mathcal{N}_i| D^2 + |\mathcal{N}_i| D) \approx O(d_{\text{avg}} D^2)\,. \tag{9}$$

where d_{avg} is the average node degree in the BS adjacency graph. With small-cell densities of 40 $cells/km^2$ (Section 3.1), d_{avg} is modest, and D is on the order of a few tens, making the GAT overhead negligible relative to the Q-network updates.

4.7.2 Multi agent deep Q network (MADQN):

Each BS agent maintains its own Q-network, modelled as a fully connected feed-forward neural network with L layers, input dimension d_s (state size), H hidden units per layer, and output dimension $|A|$ (number of potential target cells). For a single forward pass (inference), the complexity is

$$\mathcal{C}_{\text{MADQN, inf}} = O(d_s H + (L-2)H^2 + H|A|). \tag{10}$$

During training, the agent draws a mini-batch of size B from the experience replay buffer and updates the network weights. The training step complexity is dominated by B forward-backward passes, i.e.,

$$\mathcal{C}_{\text{MADQN, train}} = O(B \cdot [d_s H + (L-2)H^2 + H|A|]) \tag{11}$$

Since each BS trains independently on locally collected data, the per-BS training load is completely decoupled from the number of UEs served by other cells.

4.7.3 Federated aggregation:

Federated averaging (FedAvg, Eq. (5)) is executed periodically at the NWDAF server. Let K be the number of participating BSs and P the total number of trainable parameters in the combined GAT-Q-network model. In each

communication round, the server receives K model vectors of size P, computes their weighted average, and broadcasts the global model back. The computational complexity at the server is

$$\mathcal{C}_{\text{FedAvg}} = O(KP). \tag{12}$$

The per-BS overhead for uploading and downloading is $O(P)$, with an additional $O(P)$ for the optional Gaussian noise addition (Eq. (6)). Because FedAvg runs far less frequently than local Q-network updates (e.g., every few hundred steps), its amortised cost is small.

4.7.4 LSTM mobility predictor:

The bidirectional LSTM predictor (Eq. (7)) processes a trajectory history of length T to forecast the future UE position. With hidden state size h, the computational cost per prediction is

$$\mathcal{C}_{\text{LSTM}} = O(Th^2). \tag{13}$$

This operation is performed on demand for each active UE that may require a handover. Thanks to the small input dimension (2-D position sequence) and moderate h, the predictor adds minimal latency.

4.7.5 Summary and scalability:

Table 2 collates the asymptotic complexities, referencing the corresponding equations. The overall per-BS inference cost is dominated by the Q-network forward pass (Eq. (10)), while training cost mainly stems from local gradient computations (Eq. (11)). Both are independent of the total number of base stations K and scale linearly with the number of UEs served by that particular cell. The federated aggregation load (Eq. (12)) grows linearly with K, but infrequent execution keeps the total cost acceptable even for hundreds of cells. Consequently, the FGRL framework scales gracefully to ultra-dense 6G deployments.

Table 2. Time complexity of FGRL components

Component	Inference (per BS)	Training / Update (per BS)	Centralised overhead (per round)
GAT	$O(d_{\text{avg}}D^2)$	Same as inference	–
MADQN (Q-network)	$O(d_sH + (L-2)H^2 + H \mid A \mid)$	$O(B \cdot [d_sH + (L-2)H^2 + H \mid A \mid])$	–
Federated aggregation (FedAvg)	–	$O(P)$ (upload + download)	$O(KP)$
LSTM predictor	$O(Th^2)$ per UE	–	–

d_{avg}: average BS degree; D: feature dimension; d_s: state size; L: number of Q-network layers; H: hidden units per layer; $\mid A \mid$: action space size; B: mini-batch size; P: total model parameters; K: number of BSs; T: trajectory length; h: LSTM hidden size.

5. Performance Evaluation

5.1 *Simulation setup*

The evaluation of our framework is conducted using a custom 6G network simulator built on ns-3 (version 3.36) with the following physical-layer parameters:

- Network Area: 1000×1000 m^2
- UE Density: 700 UEs/km^2 (pedestrian: 60%, vehicular: 40%)
- Small Cell Density: 40 cells/km^2
- Carrier Frequency: 28 GHz (mmWave)
- Bandwidth: 800 MHz

5.1.1 Simulation architecture and component interconnection:

The ns-3 simulator models the three-tier network (macro, small-cell, MEC) depicted in Figure 1, including detailed mmWave propagation, UE mobility, and radio resource management. The AI components of the FGRL framework—GAT, MADQN agents, LSTM predictor, and federated aggregation—are implemented in Python (PyTorch 2.1) and interfaced with ns-3 through a lightweight middleware that exchanges state–action–reward tuples via Unix sockets at a 10 ms decision interval.

Each MEC server (collocated with a small-cell BS) hosts one MADQN agent and one GAT instance. A dedicated ns-3 module continuously collects per-UE measurements (RSRP, RSRQ, velocity, direction) and forwards them to the Python agent over the middleware. The agent assembles the state vector (Section 4.2) by concatenating local measurements with the graph-embedded neighbourhood representation computed by the GAT from the latest adjacency matrix. The agent's Q-network selects an action (target cell or no handover), which is passed back to ns-3 to execute the handover decision. The resulting reward (Eq. (4)) is then observed and stored in the experience replay buffer.

5.1.2 FL cycle:

Every 500 local training steps, each BS sends its model parameters (GAT weights and Q-network weights) to a central FL Server process, which performs Federated Averaging (Eq. (5)). To protect privacy, Gaussian noise with $\sigma = 0.01$ is added to the updates (Eq. (6)). The global model is then broadcast back to all BSs. The server runs on a separate machine (simulating the NWDAF) and does not access any raw UE data.

5.1.3 Predictive triggering and experience replay:

The bidirectional LSTM predictor (Eq. (7)) operates in the Python environment, receiving UE trajectory history from ns-3 every 50 ms and predicting the position 500 ms ahead. The Q-network incorporates the predicted position into the state vector, and the handover decision is made according to Eq. (8). The experience replay buffer holds up to 10,000 transitions and is sampled with mini-batches of size 32 for training every 100 ms.

5.1.4 Training and hyperparameters:

Training is conducted offline for 200,000 steps (approximately 30 minutes of simulated time) before performance metrics are logged. Key hyperparameters are listed in Table 3. Benchmarks (3GPP A3, Fuzzy HO, Centralized DRL [6], and FL-HO [19]) are implemented within the same ns-3 environment for a fair comparison; the centralized DRL uses a single Q-network in the core network fed by all measurement data, while FL-HO [19] applies FL on a DQN without graph modelling.

5.1.5 Hardware and software environment:

Simulations run on a server equipped with an Intel Xeon Gold 6226R CPU (32 cores, 2.9 GHz), 128 GB RAM, and an NVIDIA A100 GPU (40 GB). The ns-3 part uses a single core, while PyTorch exploits the GPU for neural network inference and training.

Table 3. FGRL simulation hyperparameters

Parameter	Value	Parameter	Value
GAT layers / heads	2 / 4	Feature dimension D	64
Q-net hidden layers	3	Hidden units per layer	128
Activation	ReLU	LSTM hidden size h	32
Learning rate	10^{-4}	Discount factor γ	0.99
Mini-batch size	32	Replay buffer capacity	10,000
Coordination weight **λ**	0.1	FL aggregation interval	500 steps
DP Gaussian noise **σ**	0.01	Prediction horizon Δt	500 ms
Trajectory history **T**	10 steps	Adaptive threshold τ	0.05–0.15

5.2 *Performance metrics*

The evaluation process relies on the following performance metrics.

- **Handover Latency**: Represents the duration between making a handover decision and finishing the process.
- **Ping-Pong Rate**: Denotes the number of unnecessary handovers between the same cells
- **Throughput**: Average user data rate
- **Signaling Overhead**: Control messages per handover event
- **Prediction Accuracy**: Shows the percent of correctly predicted handovers

6. Results

6.1 *Handover performance comparison*

Table 4 compares the performance of different handover mechanisms:

Table 4. Handover Performance Comparison

Metric	3GPP A3	Fuzzy HO	Centralized DRL	FL-HO	FGRL (Ours)
Handover Latency (ms)	48.7	36.2	22.5	24.8	18.3
Ping-Pong Rate (%)	32.1	24.5	12.8	14.2	7.0
Throughput (Mbps)	842	925	1120	1085	1205
Signaling Overhead	1.0	0.85	1.75	0.32	0.28
Prediction Accuracy (%)	-	-	82.3	84.7	91.5

The FGRL framework decreases handover latency by 62.4% in comparison with 3GPP A3 and also reduces ping-pong events by 78.2%. The proposed approach reaches 94.3% of the throughput improvement of the centralized DRL while minimizing signaling overhead.

7. Discussion

7.1 *Centralized vs. distributed trade-offs*

The results presented in Table 4 provide useful insight into the trade-offs between centralized and distributed approaches for handover management in ultra-dense 6G networks.

- **Performance:** The proposed FGRL framework achieves the best overall performance among the evaluated methods. In particular, it provides the lowest handover latency, the lowest ping-pong rate, and the highest prediction accuracy. These improvements suggest that incorporating spatial information through the GAT model and enabling cooperation among distributed agents can enhance handover decisions beyond what can be achieved using conventional reinforcement-learning approaches alone.
- **Latency:** A key advantage of the proposed framework is that decision-making is performed locally at the edge. This reduces dependence on a centralized controller and enables faster responses to rapid changes in user mobility and network conditions. Such responsiveness is especially important in ultra-dense deployments, where frequent handover events are expected.
- **Privacy:** Unlike centralized learning schemes that require large amounts of user-related data to be collected at a central location, the proposed federated architecture keeps mobility data at the serving base stations. Only model parameters are exchanged during training, which reduces privacy concerns and limits the exposure of sensitive user information.
- **Scalability:** As network density increases, centralized solutions may experience higher signaling and processing loads. In contrast, the proposed framework distributes both learning and decision-making across the network. The federated aggregation process introduces additional communication overhead, but this overhead remains significantly lower than the continuous transfer of raw measurement data required by centralized approaches.

Overall, the results indicate that the proposed FGRL framework provides a practical compromise between performance, privacy preservation, and scalability. While centralized approaches can benefit from global network visibility, the combination of FL, graph-based spatial modeling, and distributed reinforcement learning enables efficient handover management without relying on centralized data collection.

7.2 *Practical implementation considerations*

Our research reveals multiple operational aspects that need consideration:

1. **Network Slicing Integration:** The FGRL framework can be implemented as a dedicated network slice for mobility management to ensure resource isolation and quality of service.
2. **NWDAF Utilization:** The 3GPP-defined Network Data Analytics Function (NWDAF) can serve as the federated server in our architecture, while leveraging existing 5G/6G infrastructure [21].
3. **Edge Intelligence Deployment:** The GAT and MADQN components deployed at the MEC layer provide low-latency processing capabilities while maintaining coordination between neighboring cells.
4. **Interoperability:** The framework supports existing 3GPP handover procedures to enable step-by-step deployment and backward compatibility.

7.3 *Broader implications for 6G*

Our research provides multiple benefits to AI-native 6G networks beyond handover optimization, as follows:

- **AI-Native Network Slicing**: The framework demonstrates how AI can be deeply integrated with network slicing to create intelligent, self-optimizing network functions [22].

- **Privacy-Preserving Network Intelligence**: Our method keeps user data at the edge to resolve privacy issues which have blocked AI adoption in telecommunications [23].

- **Energy Efficiency**: The reduction in avoidable handovers leads to substantial energy conservation. This supports the sustainability targets of 6G networks [24].

The observed performance gains can be explained by the way the different components of the proposed framework complement each other. The GAT module allows the handover process to account for relationships among neighbouring cells, which is particularly important in ultra-dense deployments where decisions made by one cell can affect adjacent cells. The MADQN component enables each base station to learn a local handover policy based on its own network conditions while still considering information received from neighbouring cells. FL contributes by allowing knowledge acquired at different base stations to be shared without transferring raw user mobility data, reducing privacy concerns associated with centralized learning approaches. In addition, the LSTM predictor helps anticipate user movement and supports earlier handover preparation, reducing the likelihood of delayed handover decisions in highly dynamic scenarios. The combination of these mechanisms appears to be responsible for the improvements observed in latency, ping-pong rate, throughput, and prediction accuracy. A more detailed investigation of the individual contribution of each component through ablation experiments is left for future work.

8. Conclusion

This paper has presented a novel FGRL framework for predictive handover in ultra-dense 6G networks. By combining graph neural networks for spatial modeling, Multiagent Deep Q-Networks for distributed decision-making, and FL for privacy-preserving collaborative training, our approach addresses critical limitations of traditional handover mechanisms in UDNs.

Through extensive simulation, we demonstrated that the proposed framework reduces handover latency by 62%, cuts ping-pong events by 78%, and improves throughput by 41% compared to 3GPP standards. Crucially, our federated implementation achieves 94% of the performance of centralized training while preserving user privacy and reducing signaling overhead by 85%.

As 6G networks evolve toward unprecedented densities and performance requirements, our work establishes a foundational shift from reactive to predictive, self-organizing handover control. The principles developed in this paper—particularly the integration of graph-based spatial modeling with federated reinforcement learning—represent a significant step toward realizing the vision of AI-native 6G networks.

Future work will explore the integration of our framework with ISAC capabilities, leveraging environmental sensing data to further enhance handover prediction accuracy. We will also investigate the application of our approach to non-terrestrial networks, addressing the unique handover challenges in satellite and UAV-based 6G deployments.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding this article.

References

1. Giordani M, Zorzi M. Non-Terrestrial Networks in the 6G Era: Challenges and Opportunities. *IEEE Netw.* 2020, 35(2), 244-251.

2. Wang CX, You X, Gao X, Zhu X, Li Z, Zhang C, Wang H, Huang Y, Chen Y, Haas H, Thompson JS. On the road to 6G: Visions, requirements, key technologies, and testbeds. *IEEE Commun. Surveys Tuts.* 2023, 25(2), 905-974.
3. Yang H, Alphones A, Xiong Z, Niyato D, Zhao J, Wu K. Artificial-intelligence-enabled intelligent 6G networks. *IEEE Netw.* 2020, 34(6), 272-280.
4. Sheikh-Hosseini M, Rahdari F, Ghasemnezhad H, Ahmadi S, Uysal M. A comparative performance evaluation of OFDM, GFDM, and OTFS in impulsive noise channels. *IEEE Open J. Commun. Soc.* 2024, 6, 2693-2705.
5. Bui N, Cesana M, Hosseini SA, Liao Q, Malanchini I, Widmer J. A survey of anticipatory mobile networking: Context-based classification, prediction methodologies, and optimization techniques. *IEEE Commun. Surv. Tut.* 2017, 19(3), 1790-1821.
6. Guo D, Tang L, Zhang X, Liang YC. Joint optimization of handover control and power allocation based on multiagent deep reinforcement learning. *IEEE Trans. Veh. Technol.* 2020, 69(11), 13124-13138.
7. Zhang S, Chuai G, Gao W. A handover optimization algorithm for LTE-R system handover parameter prediction and dynamic adjustment. *In* International Conference in Communications, Signal Processing, and Systems. Singapore: Springer Singapore. 2018, pp. 655-669.
8. Muñoz P, Barco R, de la Bandera I. On the potential of handover parameter optimization for self-organizing networks. *IEEE Trans. Veh. Technol.* 2013, 62(5), 1895-1905.
9. Alhammadi A, Hassan WH, El-Saleh AA, Shayea I, Mohamad H, Saad WK. Intelligent coordinated self-optimizing handover scheme for 4G/5G heterogeneous networks. *ICT Express* 2023, 9(2), 276-281.
10. Hwang WS, Cheng TY, Wu YJ, Cheng MH. Adaptive handover decision using fuzzy logic for 5G ultra-dense networks. *Electronics* 2022, 11(20), 3278.
11. Mal YW, Chen JL, Lin HK. Mobility robustness optimization based on radio link failure prediction. *In* The 10th International Conference on Ubiquitous and Future Networks (ICUFN). 2018, pp. 454-457.
12. Chaudhuri S, Baig I, Das D. Self organizing method for handover performance optimization in LTE-advanced network. *Comput. Commun.* 2017, 110, 151-163.
13. Wang J, Mu W, Liu Y, Guo L, Zhang S, Gui G. Deep reinforcement learning-based satellite handover scheme for satellite communications. *In* 13th International Conference on Wireless Communications and Signal Processing (WCSP). 2021, pp. 1-6.
14. Castro-Hernandez D, Paranjape R. Optimization of handover parameters for LTE/LTE-A in-building systems. *IEEE Trans. Veh. Technol.* 2018, 67(6), pp. 5260-5273.
15. Hegazy RD, Nasr OA, Kamal HA. Optimization of user behavior based handover using fuzzy Q-learning for LTE networks. *Wirel. Netw.* 2018, 24(2), 481-495.
16. Tan J, Xiao S, Han S, Liang YC. A learning-based coexistence mechanism for LAA-LTE based HetNets. *In* IEEE international conference on communications (ICC). 2018, pp. 1-6.
17. Liu Y, Yuan X, Xiong Z, Kang J, Wang X, Niyato D. Federated learning for 6G communications: Challenges, methods, and future directions. *China Commun.* 2020, 17(9), 105-118.
18. Al-Quraan M, Mohjazi L, Bariah L, Centeno A, Zoha A, Arshad K, Assaleh K, Muhaidat S, Debbah M, Imran MA. Edge-native intelligence for 6G communications driven by federated learning: A survey of trends and challenges. *IEEE Trans. Emerg. Top. Comput. Intell.* 2023, 7(3), 957-979.

19. Noman HM, Hanafi E, Noordin KA, Dimyati K, Hindia MN, Abdrabou A, Qamar F. Machine learning empowered emerging wireless networks in 6G: Recent advancements, challenges and future trends. *IEEE Access* 2023, 11, 83017-83051.
20. Hassanpour S, Wübben D, Dekorsy A. Forward-aware information bottleneck-based vector quantization: Multiterminal extensions for parallel and successive retrieval. *IEEE Trans. Commun.* 2021, 69(10), 6633-6646.
21. 3GPP. Architecture enhancements for 5G System (5GS) to support network data analytics services. Technical Specification (TS) 23.288, 3rd Generation Partnership Project (3GPP), 2020, Version 16.4.0 (Rel-16).
22. Letaief KB, Shi Y, Lu J, Lu J. Edge artificial intelligence for 6G: Vision, enabling technologies, and applications. *IEEE J. Sel. Areas Commun.* 2021, 40(1), 5-36.
23. Khan NA, Schmid S. AI-RAN in 6G Networks: State-of-the-Art and Challenges. *IEEE Open J. Commun. Soc.* 2023, 5, 294-311.
24. Kumar S. AI/ML enabled automation system for software defined disaggregated open radio access networks: Transforming telecommunication business. *Big Data Min. Anal.* 2024, 7(2), 271-293.

How to cite this article: Fehri H, Monemizadeh M. AI-Driven Predictive Handover in Ultra-Dense 6G Networks: A Federated Graph Reinforcement Learning Approach. *Curr. Appl. Sci.*, 2025, 3(2):119-136. https://doi.org/10.22034/cas.2026.245730