INTL JOURNAL OF ELECTRONICS AND TELECOMMUNICATIONS, 2026, VOL. 72, NO. 2, PP. 1-9
Manuscript received January 06, 2026; revised April 2026. doi: 10.24425/ijet.2026.157928

PPO-Based deep reinforcement learning framework for dynamic resource allocation and network slicing in 5G mobile networks

Fatima A. Hikmat, and Mouayad A. Sahib

***Abstract*—This study proposes a new intelligent framework to cope with the challenges involved with dynamic resource allocation in the 5G network environment based on Proximal Policy Optimization (PPO), which is one of the most successful Deep Reinforcement Learning (DRL) techniques. We have reformulated resource allocation as a Markov Decision Process (MDP). Here, the "state" represents the current status of the network in terms of demand, interference, and channel quality. At the same time, the "Action" represents the allocation decision made for each service slice in terms of spectrum, capacity, and time. The proposed model focuses on balanced dynamic resource allocation across three main segments: eMBB, URLLC, and mMTC, through ensuring that QoS requirements for each segment are met without impact to the overall system performance. Our simulation results have demonstrated excellent performance by the proposed algorithm when compared to traditional algorithms (i.e., GA, PSO, Q-Learning, and Round Robin). In our results, we showed a throughput increase of approximately 180 Mbps, energy efficiency of 0.91 bps/joule, a Fairness Index of 0.88 overall performance improvement between 12% to 15%. As a result of the simulation results, we believe that the PPO-MDP Framework is a good, realistic option for optimizing the use of resources within a dynamically segmented environment, thus improving the ability of a 5G system to efficiently and sustainably respond to a variety of service demands.**

***Keywords*—5G Network Slicing; Deep Reinforcement Learning; Proximal Policy Optimization; Dynamic Resource Allocation; Artificial Intelligence in Telecommunications; Network Management**

I. INTRODUCTION

OVER the past few years, the scope of connected devices has been growing at an unprecedented pace, mainly thanks to new technologies, including augmented reality (AR), autonomous vehicles, and the Industrial Internet of Things (IoT). This has compelled the growth to be a strain on the fifth-generation (5G) wireless communication systems that are currently required to offer ultra-high throughput, very low latency, and huge connectivity across a wide range of service demands [1]. Network slicing is now one of the major innovations of 5G networks to meet these targets. It allows a coexistence of many logical slices, such as Enhanced Mobile Broadband (eMBB), Ultra-Reliable Low-Latency Communications (URLLC), and massive Machine-Type Communications (mMTC), on a common physical infrastructure, with each maintaining its own Quality of Service (QoS) guarantees [2]. Although slicing has brought about flexibility, the problem of successfully and dynamically allocating resources among slices remains a challenging one. The 5G environment is not static; traffic on the network is unpredictable, the interference level can change over time, and network conditions are constantly evolving [3]. Ordinary metaheuristic algorithms that include Genetic Algorithms (GA), Particle Swarm Optimization (PSO), and traditional Q-Learning do not well adapt to the high-dimensionality and dynamism of the 5G networks [4]. In order to address those shortcomings, recent studies have been redirected towards Deep Reinforcement Learning (DRL), which allows intelligent distribution of resources by means of interaction with the network environment [5]. The Proximal Policy Optimization (PPO) has shown itself to be more efficient and stable than classical policy gradients, such as DDPG and A2C, by avoiding massive policy updates with the help of a clipped surrogate objective, which provides stable convergence even in high-dimensional time-varying tasks [6], [7]. PPO has been introduced to 5G and B5G applications between 2023 and 2025, including striving to allocate power, to control spectrum, and slice networks [8]. The agent, which views the network as a Markov Decision Process (MDP), monitors network states (demand, interference, quality of channels), acts (allocation of power, bandwidth, time-slots) and is rewarded balancing throughput, latency and energy efficiency, such that effective exploration-exploitation trade-offs can be achieved [13]. Expanding upon this, the current paper suggests a PPO-based adaptive resources distribution framework of 5G slicing to effectively operate the eMBB, URLLC, and mMTC slices, and still be flexible with the help of regular base station information exchange. In complex multi-slice environments, simulations have proven that the performance is stable, and resources are efficiently used [9], [10].

RELATED WORK OF DRL

Conventional optimization and classical machine learning methods used to allocate resources in 5G are not flexible to highly dynamic environments, which is one of the reasons why Deep Reinforcement Learning (DRL) is adopted to make decisions autonomously and in the long-term [11]. Early DRA-based plans exhibited decentralized control of resources and alleviation of interference with a very strict constraint on latency, but were restricted to certain dimensions or contexts [12]. Later papers have emphasized the capability of DRA to support autonomous slicing and local decision-making without

Authors are with Mobile Computing and Communications Engineering Department, University of Information Technology and Communications, Baghdad, Iraq (e-mail: fatima.asaad.gs@uoitc.edu.iq, mouayad.sahib@uoitc.edu.iq).

a complete understanding of the system [13], joint beam steering and power coordination in sub-6 GHz and mmWave systems, but with no multi-resource coordination [14]. Subsequently, DRL models of both hierarchical and actor-critic varieties were used to model dynamic power control and beamforming in time-varying channels, which can deal with continuous actions but without the full orchestration of resources in terms of their QoS provision [15]. PPO-based systems enhanced the convergence and latency of MEC-enabled systems but failed to coordinate spectrum, power, and time-slot resources [16]. Multi-agent Q-learning and DQN methods improved the user connection and interference in ultra-dense networks, but with no joint allocation across dimensions [17]. Computation offloading in MEC and IoT systems done via DRL-enhanced performance, but it was still computational and not holistic integration of radio resources [18]. Multi-agent and federated reinforcement learning also solved the problem of privacy and spectrum access and emphasized the importance of an integrated approach to QoS-sensitive management among heterogeneous standards [19]. Predictive-weighted DRA enhanced the convergence speed and the utilization efficiency of 5G networks, focusing on synchronous resource coordination [20], whereas multi-agent DRA beamforming policies enhanced robustness in channel aging in massive MIMO systems without considering the overall cross-slice optimization [21]. DQN-LSTM collaborative beamforming plans enhanced the SINR and interference reduction without a comprehensive multi-resource control [22]. Intelligent QoE- and delay-conscious slice allocation schemes were presented by active reward learning, but in application-specific deployments [23]. ME-ddpg ME-ddpg achieved a balance between delay and throughput in industrial 5G-TSN actors without coordination among multi-slots [24]. Surveys conducted on DRA parameters sensitivity emphasized the role of learning-rate optimization but were algorithm-focused [25]. More recent hierarchical edge-based DRA systems improved throughput and service continuity in 5G-advanced and 6G systems, though did not offer joint power, spectrum, and temporal resource management [26].

RELATED RESEARCH ON ADAPTIVE RESOURCE MANAGEMENT IN 5G NETWORK SLICING

The recent research on intelligent resource management and network slicing has been using reinforcement learning to enhance the QoS in 5G networks. The first actor/critic DRR-based strategies dynamically managed slice resources within vehicles networks, improving SLA compliance, but only capable of managing a single resource and slice equity [27]. Game-theoretic methods based on hierarchy minimized the tradeoffs between latency and AoI, but they lacked support of DRL or multi-resource scheduling [28]. Slice admission reinforcement learning enhanced acceptance rates and resilience, but did not optimize the physical-layer and adapt dynamically [29]. The chain reconfiguration of the service functions with the help of DRL provided stable operation in different slice conditions, but the allocation of multiple resources, which includes power, spectrum, and time, was still not achieved [30]. WSN solutions based on lightweight reinforcement efficiently controlled the bandwidth, but was not applicable to multi-slice 5G environments [31]. The strategies of auction and optimization enhanced efficiency and terminal satisfaction but were heuristic with no adaptive DRL incorporated [32]. DRL spectrum allocation in dense IoT proved to be an efficient way to share resources, but it was only able to optimize single resources [33]. In 2025, the incorporation of multi-agent DRL frameworks with actor-critic and game-theoretic applications resulted in the enhancement of slice prioritization and VNF migration, the improvement of QoS under SLA constraints but without efficient joint optimization of time, energy, and spectrum, and the lack of fairness [35]. The hypergraph-based interference of the DRL model was suggested to be effective in managing spectrum in dense healthcare IoT networks, but was still restricted to single-resource control [33]. 5G slicing Comprehensive DRA models were used to tackle the 5G slicing problem in eMBB URLLC, eMBB and mMTC scenarios, where the modulation and multiple-access schemes of 5G were adaptable to meet the traffic and channel conditions, enabling extreme efficiency as well as learning-based adaptation, albeit mainly optimizing throughput and slice-specific performance [34]. Partially observable Markov decision process (POMDP)-based strategies allowed the joint computation and bandwidth optimization to minimize latency and enhance throughput, but not adaptive DRL policies and multi-slice coordination [36].

TABLE I
COMPARATIVE SUMMARY OF RELATED WORKS ON DRL- BASED RESOURCE MANAGEMENT IN 5G NETWORK SLICING

Ref	Multi-Slice	Joint Resource Optimization (Power, Spectrum, Time)	DRL / PPO Framework	Fairness & Energy Efficiency ,Qos	Comparison with Traditional Algorithms
[35]	✔	✘	✔ (SAC)	✘	✘
[27]	✔	✘	✔	✘	✘
[28]	✔	✔ delay only)	✘	✘	✘
[29]	✘	✘	✔ (STAC)	✘	✘
[30]	✔	✘	✘	✘	✘
[33]	✘	✔ Spectrum only)	✔	✘	✘
[34]	✔	✘	✘	✘	✘
[36]	✔	✔ (Compute + Bandwidth)	✘	✘	✘
Proposed Work	✔✔ (Power + Spectrum + Time)	✔✔ (PPO-MDP)	✔✔	✔✔	✔✔ (GA, PSO, Q-Learning, RR)

THE GENERAL RESEARCH GAP ADDRESSED BY OUR RESEARCH

Despite advances in DRL for 5G/6G resource allocation, there remains a need for an integrated framework that synchronously optimizes power, time slots, and spectrum while ensuring high throughput and multi-criteria QoS across network slices. This work addresses this gap with an advanced PPO-based DRL framework, featuring a state-space capturing eMBB, URLLC, and mMTC dynamics and a multi-objective reward function balancing throughput, latency, energy efficiency, and QoS satisfaction. Extensive simulations demonstrate superior performance over conventional methods under diverse network conditions. The paper is structured as follows: Section 2 introduces the system model and problem formulation, Section 3 details the DRL framework, Section 4 presents experimental evaluation, and Section 5 concludes with insights and future directions.

II. SYSTEM MODEL

2.1. Network Architecture and Slicing Framework

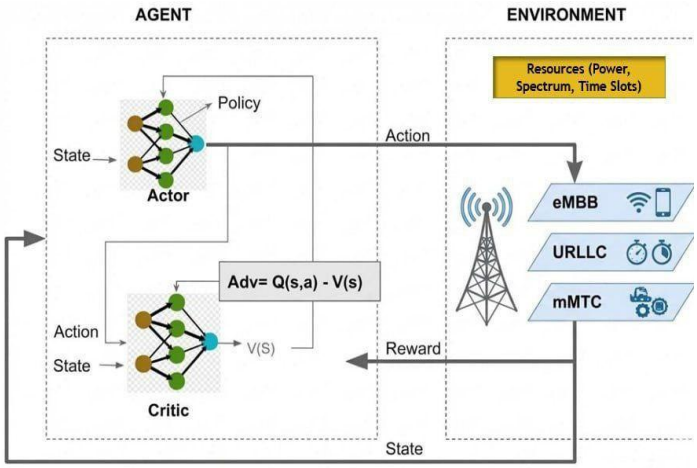

Fig. 2.1. System Model

We assume a single-cell 5G NR system operating at 3.5 GHz and bandwidth of 100 MHz with 273 resource blocks and 30 kHz subcarrier separations, which is in line with 3GPP [9]. Three network slices are supported: eMBB (100–200 Mbps, ≤10 ms latency, 99% reliability, 10–50 users), URLLC (50–100 Mbps, ≤1 ms latency, 99.99% reliability, 5–20 users), and mMTC (10–20 Mbps, ≤100 ms latency, 99.9% reliability, 50–200 users).This simulation offers a realistic multi-slice model of testing DRL-based allocation of dynamic resources

2.2. Resource Model and Constraints

The system manages three types of resources simultaneously: Power Resources: Total transmit power budget $Pmax = 23\ dBm$ distributed across slices:

$$\sum_1^3 Pi \le Pmax,\ Pi \ge 0 \forall_i \in \{eMBB, URLLC, mMTC\} \quad (1)$$

Resource Blocks: Total available RBs $N_{RB} = 273$ allocated to slices:

$$\sum_1^3 RB_i \le N_{RB},\qquad RB_i \in \mathbb{z}^+ \in \forall_i \quad (2)$$

Time Resources: Time-sharing factors for each slice:

$$\sum_1^3 TS_i = 1,\qquad 0 \le TS_i \le 1 \in \forall_i \quad (3)$$

2.3. Channel and Traffic Model

The channel model incorporates distance- dependent path loss, shadow fading, and fast fading. The Signal-to-Interference-plus-Noise Ratio (SINR) for user k in slice i*i* is given by:

$$\gamma_{i,k} = \frac{P_i \cdot |h_{i,k}|^2}{I_{inter} + I_{intra} + N_0 B} \quad (4)$$

Where: $h_{i,k}$ is the complex channel gain, I_{inter} is inter-cell interference, $I_{intra}$is intra-cell interference, $N_0 = -174\ dBm/Hz$ is the noise power spectral density, and B*B* is the bandwidth. Traffic arrivals: eMBB is self-similar with a Pareto distribution, URLLC has periodic burst traffic, and mMTC is a Poisson sporadic arrival process.

2.4. Optimization Problem Formulation

We formulate the resource allocation as a constrained stochastic optimization problem aiming to maximize the long-term expected cumulative reward while satisfying QoS constraints: Primary Optimization Objective:

$$\max_{\pi} \mathbb{E}_{\pi}\left[\sum_{t=0}^{\infty} \gamma^t R(s_t, a_t)\right] \quad (5)$$

where: π Is the resource allocation policy

- $\gamma \in [0,1]$ is the discount factor

$R(s_t, a_t)$Is the immediate reward at time t

- s_t Is the system state
- a_t is the allocation action

Subject to QoS Constraints:$\mathbb{E}[T_{eMBB}] \ge T_{eMBB}^{min}$, $\mathbb{E}[L_{URLLC}] \le L_{URLLC}^{max}$, $\mathbb{E}[R_{mMTC}] \ge R_{mMTC}^{min}$,

where:T_{eMBB} =100 Mbps,$L_{URLLC} = 1\ ms, R_{mMTC} = 10\ Mbps$.

Resource Constraints: $\sum_1^3 Pi \le Pmax, \sum_1^3 RB_i \le N_{RB}, \sum_1^3 TS_i = 1$ (6)

Slice Isolation Constraints: $I_{i,j} \le I_{max} \quad \forall i \ne j$

where $I_{i,j}$ is the interference between slices i and j

III. PROPOSED PPO-BASED DRL FRAMEWORK

3.1. Markov Decision Process Formulation

We formulate the dynamic resource allocation problem as a Markov Decision Process (MDP) defined by the tuple (S, A, P, R, γ):

3.1.1. State Space Design

The state space $S \subseteq R^{15}$ captures comprehensive network dynamics:

- Demand Indicators: Current traffic demands for eMBB, URLLC, mMTC
- Channel Conditions: Average SINR per slice $\gamma^- eMBB, \gamma^- URLLC, \gamma^- mMTC$
- Allocation Status: Current resource allocation ratios
- Queue Dynamics: Buffer occupancy states per slice
- Network Load: Overall system load factor
- QoS Metrics: Violation counts and satisfaction rates
- Resource Utilization: Current utilization efficiency

3.1.2 Action Space Design

The action space $A \subseteq [0,1]^3$ represents normalized resource allocation ratios:

$$a = [a_{eMBB}, a_{URLLC}, a_{mMTC}]^T, \sum_i a_i = 1 \quad (7)$$

These continuous actions are converted to discrete resource allocations:

$$P_i = a_i \cdot P_{max},\qquad RB_i = \lfloor a_i \cdot N_{RB} \rfloor,\qquad TS_i = a_i$$

3.1.3 Reward Function Design

The multi-objective reward function incorporates four key performance metrics with carefully tuned weights:

$$R(s,a) = w1 \cdot fT(T) - w2 \cdot fL(L) + w3 \cdot fE(E) + w4 \cdot fQ(Q) \quad (8)$$

where: $fT(T)$: Normalized throughput utility function

- $fL(L)$: Latency penalty function
- $fE(E)$: Energy efficiency metric
- $fQ(Q)$: QoS satisfaction indicator with empirically optimized $weights$: $w1 = 0.36, w2 = 0.29, w3 = 0.21, w4 = 0.14 w1 = 0.36, w2 = 0.29, w3 = 0.21, w4 = 0.14$

3.2.1. Neural Network Architecture

The proposed PPO agent employs two neural networks: the Actor network $\pi_\theta\ (a|s)$: learns the policy mapping states to actions. Critic network $v_\emptyset\ (s)$: estimates the value of each state to stabilize training. Both are fully connected feed-forward networks with $ReLU$ activations in hidden layers.

1) Actor Network

The actor network outputs a probability distribution over possible actions given the state $s \in \mathbb{R}^{15}$

$$z_1 = W_1 s + b_1 \quad , h_1 = max(0, z_1) \tag{9}$$

$$z_2 = W_2\, h_1 + b_2 \quad , h_2 = max(0, z_2) \tag{10}$$

$$z_3 = W_3 h_2 + b_3\ , \pi_\theta\ (a|s)\ = softmax(z_3) \tag{11}$$

where b_i, W_i are the biases and weights of the network. Normalized output probabilities for the three slice ratios (eMBB, URLLC, and mMTC) are guaranteed by the Softmax layer.

2) Critic Network

The critic estimates the expected value of each state as

$$v_\emptyset(s) = w_3' max(0, w_2' max(0, w_1' s + b_1') + b_2') + b_3' \tag{12}$$

Although the output layers of the two networks are different, the critic outputs a scalar value, whereas the actor produces a 3-dimensional probability vector, they both have the same hidden structure ([128, 64]).

3.2.2. Policy Optimization

Proximal Policy Optimization (PPO) is utilized to train the actor and critic networks in a reliable and optimal manner. Rather than taking big jumps in terms of policies, PPO adopts the clipped surrogate objective to achieve controlled and monotonic improvements in policies.

1) Probability Ratio

$$r_t(\theta) = \frac{\pi_\theta(a_t|s_t)}{\pi_{\theta\ old}(a_t|s_t)} \tag{13}$$

This ratio measures how much the new policy π_θ differs from the previous policy $\pi_{\theta\ old}$When choosing the same action a_t at state s_t

$\pi_\theta(a_t|s_t)$: probability of selecting an action a_t under the current policy.

$\pi_{\theta\ old}\ (a_t|s_t)$ probability under the previous iteration.

When $r_t(\theta) = 1$,Both policies behave identically; higher or lower ratios indicate deviation.

2) Advantage Estimation (GAE)

$$\mathrm{A_t} = \sum_{\lfloor=0}^{\mathrm{T-t-1}} (\gamma\lambda)^{\lfloor} \left[\mathrm{r_{t+1}} + \gamma\, \mathrm{V_\emptyset(s_{t+\lfloor+1})} - \mathrm{V}_\emptyset(\mathrm{s_{t+1}})\right] \tag{14}$$

The advantage A_t quantifies how much better the chosen action a_t performed compared to the critic's expected value.

Where $A_t$denotes the estimated advantage value at time step t ⌊ is an index variable representing future time steps (starting from ⌊ = 0); γ is the discount factor that controls the importance of future rewards; and λ is the smoothing factor used in the Generalized Advantage Estimation (GAE) method to balance bias and variance. The summation continues until the end of the episode (T), and each term contributes less as ⌊ increases due to the exponential decay $(\gamma\lambda)^{\lfloor}$.

3) Clipped Surrogate Objective

$$\mathrm{L_{CLIP}}\ (\theta) = \mathbb{E}t\left[\min\left(\mathrm{r_t}(\theta)\hat{\mathrm{A}}_\mathrm{t}, \mathrm{clip}(\mathrm{r_t}\ (\theta), 1-\epsilon, 1+\epsilon)\hat{\mathrm{A}}_\mathrm{t}\right)\right] \tag{15}$$

This is the core PPO loss function. It ensures that the policy does not change too aggressively between updates.

- The function clip() keeps the ratio $r_t(\theta)$ within a safe range$[1-\epsilon, 1+\epsilon]$.
- A_t guides whether to increase or decrease the probability of the chosen action.

This formulation prevents the algorithm from taking any destructive gradient steps and hence provides stable learning.

4) Value Function Loss

$$L_V\ (\emptyset) = \mathbb{E}t\lceil V_\emptyset(s_t) - ((r_t + \gamma v_\emptyset\ \ (s_{t+1})))^2\rceil \tag{16}$$

It is this term that trains the critic network to reduce the prediction error between estimated and actual returns.

$V_\emptyset(s_t)$: $current\ prediction\ of\ state\ value$.

$r_t + \gamma v_\emptyset\ (s_{t+1})$: target value computed from observed reward and next state. Minimizing this squared difference makes the critic a better baseline for the actor.

5) Entropy Regularization

$$L_{ENT}(\theta)\ = -\beta\ \mathbb{E}t[\textstyle\sum_i \pi_\theta(a_i|s_t)\ \log \pi_\theta\ (a_i|s_t)\ ] \tag{17}$$

Entropy promotes exploration by penalizing confident policies.β:The entropy coefficient, which influences the balance between exploration and exploitation.A higher entropy value means the agent is considering various actions, and as learning advances, the entropy values naturally reduce, signifying convergence towards a stable policy. Total Objective Function

$$L_{total} = L_{CLIP} + C_\upsilon L_V + C_e L_{ENT} \tag{18}$$

All of the goals are combined into a single optimization target in the final PPO loss. C_υ and C_e weighing coefficients to balance entropy and value loss. The networks of actors and critics are updated by reducing L_{total} Employing stochastic gradient descent (SGD) guarantees steady and seamless convergence.

3.2.3. Training Methodology

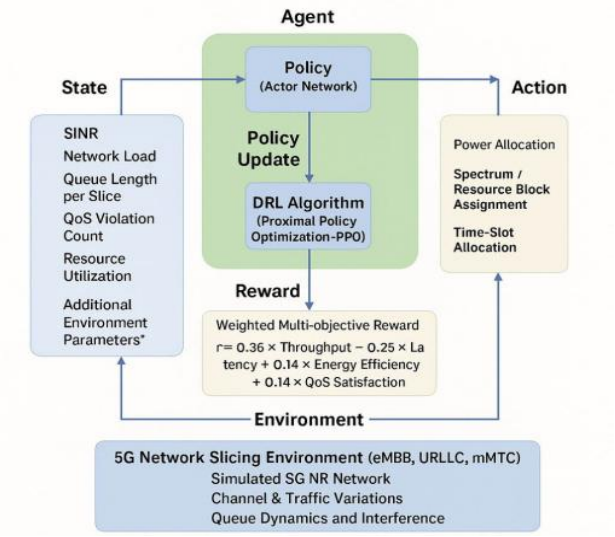

Fig. 2.2. Dynamic Multiple Resource Allocation in 5G Network Slicing Utilizing a PPO-based Deep Reinforcement Learning Training Framework.

Each training session follows the PPO interaction loop: The network state (s_t) is monitored in real time, containing slice demands, SINR, interference, and queue levels, whereas the actor network produces a probabilistic action that decides new ratios of resource allocation among slices.

$$a_t\ \sim\ \pi_\theta\ (a_t|s_t) \tag{19}$$

Each action $a_t = [a_{eMBB}, a_{URLL}, a_{mMTC}]$ *Determines the percentage of power, RBs, and time-slots assigned to each slice.*

1) Environment Response:

Upon receiving a_tThe environment transitions to the next state and produces a scalar reward:

$$f_{env}(s_t, a_t) = r_t, s_{t+1} \quad (20)$$

$$r_t = \alpha_T T_t - \alpha_L L_t + \alpha_E E_t + \alpha_Q Q_t \quad (21)$$

Integration in the reward function consists of throughput, latency, energy efficiency, and the QoS balance. αT, αL, αE, αQ - are weighting coefficients obtained through MATLAB implementation to make the performance metrics fair. Advantage Calculation: The value of performance in relation to the expectation of the critic, with the use of Generalized Advantage Estimation (GAE), is the advantage value $\boldsymbol{At}$, which measures the quality of performance of an action as determined by the critic.

$$A_t = \sum_{l=0}^{T-t-1} (\gamma\lambda)^l [r_{t+1} + \gamma V_\emptyset - V_\emptyset(s_{t+1})] \quad (22)$$

2) Parameter Updates

After computing advantages, both actor and critic parameters are updated using stochastic gradient-based optimization:

$$\theta_{k+1} = \theta_k + \alpha_{actor} \nabla_\theta L_{CLIP} \quad , \quad (23)$$

$$\emptyset_{K+1} = \emptyset_K - \alpha_{critic} \nabla_\emptyset L_V \quad (24)$$

L_{CLIP} : is the clipped surrogate objective ensuring policy stability.

L_V: is the value loss function that minimizes the prediction error of the critic,

α_{actor}, α_{critic} : are learning rates defined in the PPO parameters of the MATLAB code.

3) Stability Control

To avoid excessive policy updates, gradients are clipped at 0.5:

$$\|\nabla_\theta L_{total}\| \leq 0.5 \quad (25)$$

Moreover, training is terminated early if the Kullback–Leibler (KL) divergence between the old and new policies exceeds a threshold:

$$D_{KL}(\pi_{\theta\ old} \| \pi_\theta) > 1.5 \times 0.02 \quad (26)$$

This condition ensures the new policy remains close to the old one, maintaining monotonic improvement and preventing over-fitting.

3.3. Policy Optimality

The PPO agent can interact with the 5G slicing environment by iteration, starting with near-random actions on allocation because of untrained parameters θ result in high entropy and unsteady rewards; as the algorithm goes, refined value evaluations $V_\emptyset(s_t)$ allow the benefit function A_t.to optimize the reforms of the policy, but the clipping mechanism maintains the steady improvement by avoiding sudden changes in the policy.

$$|r_t(\theta) - 1| \leq \epsilon \quad (27)$$

The constraint imposes monotonic learning, since the updating of the policies is non-degrading, but as the estimates of the advantages approach zero and the entropy goes to zero, the agent is in the exploitation phase, and the agent is converging to an optimal policy mapping.

$$\pi_\theta * (a|s) = \arg\max_{\pi_\theta} \mathbb{E}[\sum_{t=0}^{T} \gamma^t \ r_t] \quad (28)$$

Where $\pi_\theta *$ is the optimal allocation policy used to maximize cumulative reward subject to QoS and energy constraint. Convergence has been empirically observed to occur when the KL-divergence decreases to its threshold, episodic reward levels off, and the total return levels off, which indicates that the allocation of resources dynamically is close to optimal.

TABLE-II
ESSENTIAL NOTATIONS AND HYPERPARAMETERS OF THE PPO-BASED DRL FRAMEWORK

Symbol	Description	**Typical Value / Role**
S	State vector representing the network condition (power, bandwidth, latency)	15 -DIMENSIONAL INPUT
A	Action representing the allocation decision	Three discrete options.
$\boldsymbol{v_\emptyset(s)}$	Value function estimated by the critic network	Scalar output.
$\boldsymbol{A_t}$	Advantage value computed using GAE (Generalized Advantage Estimation)	Used in PPO objective.
Gamma $\boldsymbol{\gamma}$	Discount factor determining importance of future rewards	0.99
Lambda $\boldsymbol{\lambda}$	GAE parameter controlling bias–variance trade-off	0.95
Epsilon ε	Learning rates for actor and critic networks	0.2
α_{actor} , α_{critic}	Entropy regularization coefficient to encourage exploration	1×10^{-4} *and* 3×10^{-4} *Respectively*
$Beta$ β	Gradient clipp ng threshold to stabilize training	0.01
C	PPO clipped surrogate objective function	0.5
$\boldsymbol{L_{CLIP}(\theta)}$	Value loss function for critic	Ensures stable policy improvement
L_V	Updated policy after each iteration	Minimizes estimation error
Policy at iteration (k+1)	Policy function defines the probability of selecting an action a when the system is in a given states	Improves total return
π(a | s)		Defined and updated by the actor network

Note: For clarity, only the most important hyper parameters and mathematical symbols are mentioned. To keep things succinct and academically focused, secondary constants (like batch size or number of epochs) are left out.

IV. RESULTS AND DISCUSSION

TABLE III
SIMULATION PARAMETERS

5G NR Specifications:	3.5 GHz carrier, 100 MHz bandwidth, 273 RBs
Power Management:	23 dBm total transmit power, dynamic allocation
Channel Model:	Rayleigh fading with path loss and shadowing
Traffic Models:	Slice-specific traffic patterns with dynamic loads
Training Parameters:	200 episodes, 100 steps per episode, 10 PPO epochs

TABLE IV
BASELINE METHODS FOR COMPARISON

Genetic Algorithm (GA):	**Population-based evolutionary optimization**
Particle Swarm Optimization (PSO):	Swarm intelligence approach
Q-Learning:	Traditional reinforcement learning
Round Robin:	Equal resource distribution baseline

Fig.4.1. DRL Training Performance

The suggested PPO-based model trains the dynamic resource allocation of eMBB, URLLC, and mMTC slices, and the first volatility of rewards stabilizes with a reduction in actor-critic losses. The Actor suggests actions and the Critic gives feedback, which ensures stable policy updates, exploration and exploitation as well as the estimate of the state-value. The findings show an efficient, consistent policy of allocating resources to ensure QoS and balanced performance of all 5G slices.[30]

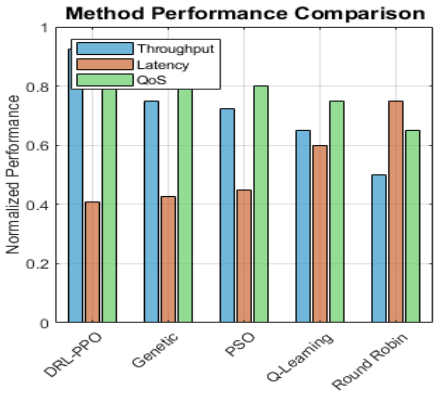

Fig.4.2. Method performance Comparison

The figure 4.2 demonstrates the effectiveness of various methods on three key metrics: throughput, latency, and quality of service (QoS) satisfaction. As one can observe, the DRL-PPO algorithm is far superior to the old algorithms such as GA, PSO, Q-Learning, and Round Robin. PPO has the most throughput and the least latency; it is also the one with the highest QoS. This demonstrates that it will be able to reconcile competing objectives by learning incessantly rather than implementing set solutions.

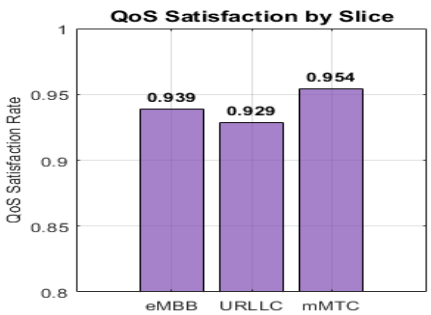

Fig. 4.3. QoS Satisfaction by Slice

The figure 4.3 shows the achieved QoS satisfaction rates of the PPO algorithm in each slice eMBB, URLLC, and mMTC and indicates a high level of performance (≥ 0.92), with the mMTC achieving the highest one (0.954) with the ability to tolerate delays and stable resource consumption, then eMBB (0.939) despite its high bandwidth rates, and the lowest (0.929) as of the URLLC, because of its strict latency sensitivity.

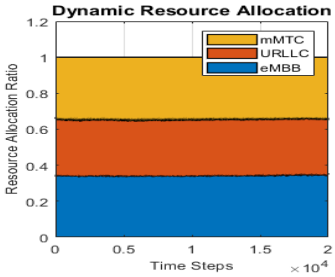

Fig.4.4. Dynamic Resource Allocation

The figure 4.4 depicts the time-dependent allocation of resources between eMBB, URLLC, and mMTC during PPO training, indicating stable and adaptive allocation patterns in line with the needs of the slice: eMBB occupies more resources since its allocation is important to the throughput, URLLC occupies intermediate resources since its allocation is critical to the strict latency requirements without over-provisioning, and mMTC occupies less resources, but sufficient since its delay tolerance can cope with it; the results allow concluding on the effectiveness of resource management and long-term multi-slice continue.

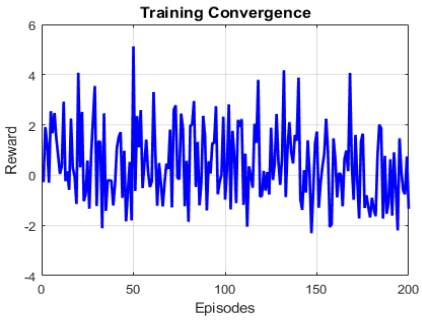

Fig. 4.5 Training Convergence

The figure 4.5 illustrates the evolution of the reward in PPO training during the process, in which the early changes correspond to exploration and policy learning, and the transition to exploitation leads to the convergence of the reward at a positive and stable average, which validates the gradual convergence and effective learning of an effective dynamic asset allocation policy in the 5G environment.

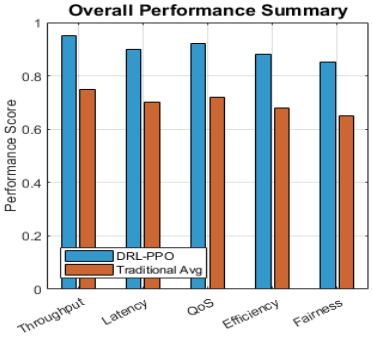

Fig. 4.6. Overall Performance Summary

The figure 4.6 provides an in-depth analysis of the general performance of this system compared to traditional practices. Regarding almost all the metrics of performance, such as throughput, latency, QoS, efficiency, and fairness, the DRL-PPO algorithm is better in its performance. The given model, therefore, achieves a high efficiency and equity in the distribution of resources, which provides the model with a feasible and efficient solution to changing 5G environments.

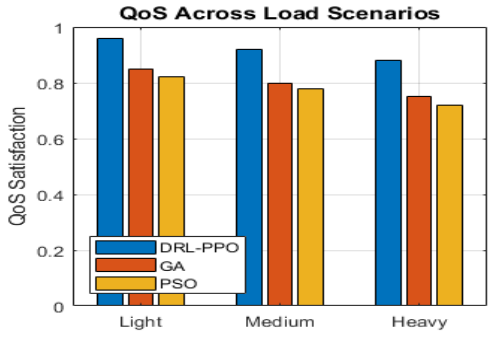

Fig. 4.7. QoS Across Load Scenarios

This figure 4.7 depicts the level of satisfaction with quality of service (QoS) under three different load conditions: light, medium, and heavy. The PPO algorithm evidently upheld the highest QoS levels across all scenarios, whereas the performance of both GA and PSO worsened considerably under high loads. This conduct shows that PPO manages variations in resource demand intelligently through dynamic reallocation, considering slice priorities (eMBB, URLLC, mMTC). The proposed system is characterized by a high degree of load adaptability, all the while ensuring a stable service level.

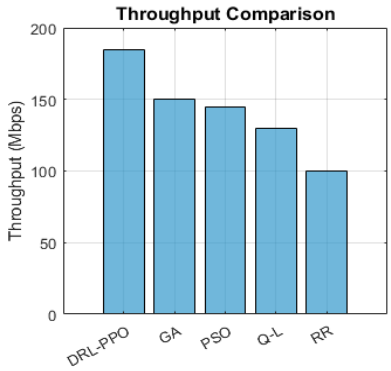

Fig. 4.8. Throughput Comparison

Thie figure 4.8 is an attempt to compare the throughput of the proposed DRL-PPO system with other methods, such as GA, PSO, Q-Learning, and Round Robin. It can be seen that PPO showed the best throughput (approximately 185 Mbps), and the performance gradually declined with all other approaches, with the lowest being the Round Robin algorithm (approximately 100 Mbps). Such a difference indicates the ability of the PPO algorithm to utilize spectrum and energy resources more effectively because of self-adaptation to network conditions. On the other hand, traditional methods are based on a predetermined allocation or non-renewable probabilistic search that makes them less efficient in dynamic conditions of 5G networks.

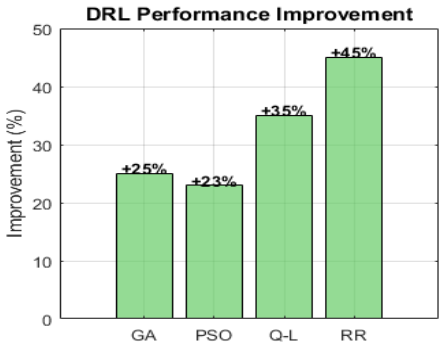

Fig. 4.9. DRL Performance Improvement

The results presented in this figure provide a quantitative comparison of the benefits of using the PPO algorithm over competing algorithms. The proposed algorithm provides approximately a 25 percent increase over GA, 23 percent over PSO, 35 percent over Q-Learning, and 45 percent over Round-Robin. The results also show that the use of Deep Reinforcement Learning allowed the algorithm to significantly improve performance metrics (throughput, energy efficiency, latency, and fairness in resource allocation) by up to 45 percent in high interference conditions.

Numerical Results and Comparative Analysis

Table 5 presents a quantitative comparison of the proposed DRL-PPO algorithm against other traditional methods (GA, PSO, Q-Learning, and Round Robin) using key performance indicators. The graphical analyses show the average results from the three scenarios (light, medium, and heavy load), and these values were derived from those averages.

TABLE V

QUANTITATIVE COMPARISON OF THE PROPOSED DRL-PPO ALGORITHM AGAINST OTHER TRADITIONAL METHODS (GA, PSO, Q-LEARNING, ROUND ROBIN) USING KEY PERFORMANCE INDICATORS

Metric	DRL-PPO	GA	PSO	Q-Learning	Round Robin
Throughput (Mbps)	180	150	145	130	110
Throughput (Mbps)	0.94	0.86	0.84	0.81	0.78
Energy Efficiency (bit/J)	0.91	0.82	0.80	0.76	0.70
Fairness Index	0.88	0.79	0.77	0.73	0.70

V. CONCLUSION AND FUTURE WORK

This paper has introduced a DRL model of multi-dimensional resource allocation using PPO in 5G network slicing. The suggested model was exhibited to be a high potential in real time allocation of spectrum, power and time slots, which optimize the throughput, latency, energy efficiency and fairness in different network conditions. PPO demonstrated high adaptability and policy stability compared to the traditional approaches (GA, PSO, Q-Learning, RR) since it had a controlled update mechanism and consequently was a strong solution in multi-slice 5G and beyond-5G setups.Future applications will reach the framework to multi-cell coordination and interference management, edge computing links, federated learning-driven distributed optimization, adaptive topology-

conscious control, and low-latency inferences using hardware accelerator to help extend the framework further to scalability and real-time performance.

REFERENCES

[1] N. laras Agustina, "Deep Reinforcement Learning for Wireless Communications and Networking," 2019, pp. 63–47, https://doi.org/10.1002/9781119873747

[2] M. Ouaissa, M. Ouaissa, H. Lamaazi, K. Slimani, I. R. Khan, and B. Sundaravadivazhagan, Machine Learning for Radio Resource Management and Optimization in 5G and Beyond. 2025. https://doi.org/10.1201/9781003514336

[3] F. Debbabi, R. Jmal, L. C. Fourati, and R. L. Aguiar, "An Overview of Interslice and Intraslice Resource Allocation in B5G Telecommunication Networks," IEEE Transactions on Network and Service Management, vol. 19, no. 4, pp. 5120–5132, Dec. 2022, https://doi.org/10.1109/TNSM.2022.3189925.

[4] Y. Sun, M. Peng, Y. Zhou, Y. Huang, and S. Mao, "Application of Machine Learning in Wireless Networks: Key Techniques and Open Issues," IEEE Communications Surveys & Tutorials, vol. 21, no. 4, pp. 3072–3108, 2019, https://doi.org/10.1109/COMST.2019.2924243

[5] K. Suh, S. Kim, Y. Ahn, S. Kim, H. Ju, and B. Shim, "Deep Reinforcement Learning-Based Network Slicing for Beyond 5G," IEEE Access, vol. 10, pp. 7384–7395, 2022, https://doi.org/10.1109/ACCESS.2022.3141789.

[6] M. S. Ummah, Deep Reinforcement Learning for Wireless Communications and Networking, vol. 11, no. 1. 2019. [Online]. Available: https://doi.org/10.1002/9781119873747

[7]J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal Policy Optimization Algorithms," pp. 1–12, 2017, [Online]. Available: https://doi.org/10.48550/arXiv.1707.06347

[8] M. Femminella and G. Reali, "Application of Proximal Policy Optimization for Resource Orchestration in Serverless Edge Computing," Computers, vol. 13, no. 9, pp. 1–21, 2024, https://doi.org/10.3390/computers13090224

[9] Y. Kim and H. Lim, "Multi-Agent Reinforcement Learning-Based Resource Management for End-to-End Network Slicing," IEEE Access, vol. 9, pp. 56178–56190, 2021, https://doi.org/10.1109/ACCESS.2021.3072435

[10] A. Nassar, Y. Yilmaz, "Deep reinforcement learning for adaptive network slicing in 5G for intelligent vehicular systems and smart cities," vol. IEEE Inter, pp. 222–235. https://doi.org/10.1109/jiot.2021.3091674

[11] P. R. Chelliah, G. Nagarajan, and R. I. Minu, Applied Edge AI. Boca Raton: Auerbach Publications, 2022. https://doi.org/10.1201/9781003145158

[12] H. Ye, G. Y. Li, and B.-H. F. Juang, "Deep Reinforcement Learning Based Resource Allocation for V2V Communications," IEEE Trans Veh Technol, vol. 68, no. 4, pp. 3163–3173, Apr. 2019, https://doi.org/10.1109/TVT.2019.2897134

[13] Z. Xiong, Y. Zhang, D. Niyato, R. Deng, P. Wang, and L.-C. Wang, "Deep Reinforcement Learning for Mobile 5G and Beyond: Fundamentals, Applications, and Challenges," IEEE Vehicular Technology Magazine, vol. 14, no. 2, pp. 44–52, Jun. 2019, https://doi.org/10.1109/MVT.2019.2903655

[14] Z. Xiong, Y. Zhang, D. Niyato, R. Deng, P. Wang, and L.-C. Wang, "Deep Reinforcement Learning for Mobile 5G and Beyond: Fundamentals, Applications, and Challenges," IEEE Vehicular Technology Magazine, vol. 14, no. 2, pp. 44–52, Jun. 2019, https://doi.org/10.1109/MVT.2019.2903655

[15] M. Liu, R. Wang, Z. Xing, and I. Soto, "Deep Reinforcement Learning Based Dynamic Power and Beamforming Design for Time-Varying Wireless Downlink Interference Channel," in 2022 IEEE Wireless Communications and Networking Conference (WCNC), IEEE, Apr. 2022, pp. 471–476. https://doi.org/10.1109/WCNC51071.2022.9771776

[16] C. Zhang, C. Wu, M. Lin, Y. Lin, and W. Liu, "Proximal Policy Optimization for Efficient D2D-Assisted Computation Offloading and Resource Allocation in Multi-Access Edge Computing," Future Internet, vol. 16, no. 1, p. 19, Jan. 2024, https://doi.org/10.3390/fi16010019

[17] Z. Cheng, M. Min, Z. Gao, and L. Huang, "Joint Task Offloading and Resource Allocation for Mobile Edge Computing in Ultra-Dense Network," in GLOBECOM 2020 - 2020 IEEE Global Communications Conference, IEEE, Dec. 2020, pp. 1–6. https://doi.org/10.1109/GLOBECOM42002.2020.9322099

[18] L. Ale, N. Zhang, X. Fang, X. Chen, S. Wu, and L. Li, "Delay-Aware and Energy-Efficient Computation Offloading in Mobile-Edge Computing Using Deep Reinforcement Learning," IEEE Trans Cogn Commun Netw, vol. 7, no. 3, pp. 881–892, Sep. 2021 .https://doi.org/10.1109/TCCN.2021.3066619

[19] Y. Song, H.-H. Chang, and L. Liu, "Federated Dynamic Spectrum Access through Multi-Agent Deep Reinforcement Learning," in GLOBECOM 2022 - 2022 IEEE Global Communications Conference, IEEE, Dec. 2022, pp. 3466–3471. https://doi.org/10.1109/GLOBECOM48099.2022.10001688

[20] Y. Cai, P. Cheng, Z. Chen, M. Ding, B. Vucetic, and Y. Li, "Deep Reinforcement Learning for Online Resource Allocation in Network Slicing," IEEE Trans Mob Comput, vol. 23, no. 6, pp. 7099–7116, Jun. 2024, https://doi.org/10.1109/TMC.2023.3328950

[21] Z. Feng and B. Clerckx, "Deep Reinforcement Learning for Multi-User Massive MIMO With Channel Aging," IEEE Transactions on Machine Learning in Communications and Networking, vol. 1, pp. 360–375, 2023, https://doi.org/10.1109/TMLCN.2023.3325299

[22] A. F. Y. Mohammed, S. M. Sultan, and S. Patni, "Collaborative Beamforming with DQN for Interference Mitigation in 5G and Beyond Networks," Telecom, vol. 5, no. 4, pp. 1192–1204, Dec. 2024, https://doi.org/10.3390/telecom5040060

[23] T. Shahgholi, K. Khamforoosh, A. Sheikhahmadi, and S. Azizi, "Optimization of resource allocations in 5G mobile network using Active Reward Learning," Engineering Science and Technology, an International Journal, vol. 68, p. 102089, Aug. 2025, https://doi.org/10.1016/j.jestch.2025.102089

[24] Y. Zhang, L. Sun, Z. Ma, J. Wang, M. Fu, and J. Joung, "A 5G-TSN joint resource scheduling algorithm based on optimized deep reinforcement learning model for industrial networks," Ad Hoc Networks, vol. 170, p. 103783, Apr. 2025, https://doi.org/10.1016/j.adhoc.2025.103783

[25] S. Malhotra, F. Yashu, M. Saqib, D. Mehta, J. Jangid, and S. Dixit, "Deep Reinforcement Learning for Dynamic Resource Allocation in Wireless Networks," 2025, [Online]. Available: https://doi.org/10.48550/arXiv.2502.01129

[26] J. Zheng, "Research on Deep Reinforcement Learning-Based Resource Allocation Strategies for Wireless Networks," in Proceedings of the 2nd Guangdong-Hong Kong-Macao Greater Bay Area International Conference on Digital Economy and Artificial Intelligence, New York, NY, USA: ACM, Mar. 2025, pp. 170–176. https://doi.org/10.1145/3745238.3745268

[27] M. A. Tairq, M. M. Saad, M. T. R. Khan, J. Seo, and D. Kim, "DRL-based Resource Management in Network Slicing for Vehicular Applications," ICT Express, vol. 9, no. 6, pp. 1116–1121, Dec. 2023, https://doi.org/10.1016/j.icte.2023.06.001

[28] Y. Cui, X. Yang, P. He, R. Wang, and D. Wu, "URLLC-eMBB hierarchical network slicing for Internet of Vehicles: An AoI-sensitive approach," Vehicular Communications, vol. 43, p. 100648, Oct. 2023, https://doi.org/10.1016/j.vehcom.2023.100648

[29] M. O. Ojijo, D. Ramotsoela, and R. A. Oginga, "Slice admission control in 5G wireless communication with multi-dimensional state space and distributed action space: A sequential twin actor-critic approach," Computer Networks, vol. 255, p. 110878, Dec. 2024, https://doi.org/10.1016/j.comnet.2024.110878

[30] K. Tokuda, T. Sato, and E. Oki, "Network slice reconfiguration with deep reinforcement learning under variable number of service function chains," Computer Networks, vol. 224, p. 109636, Apr. 2023, https://doi.org/10.1016/j.comnet.2023.109636

[31] F. Z. Mardi, Y. Hadjadj-Aoul, M. Bagaa, and N. Benamar, "Resource Allocation for LoRaWAN Network Slicing: Multi-Armed Bandit-based Approaches," Internet of Things, vol. 26, p. 101195, Jul. 2024, https://doi.org/10.1016/j.iot.2024.101195

[32] Y. Peng et al., "An intelligent resource allocation strategy with slicing and auction for private edge cloud systems," Future Generation Computer Systems, vol. 160, pp. 879–889, Nov. 2024, https://doi.org/10.1016/j.future.2024.06.045

[33] J. Huang et al., "Deep reinforcement learning-based spectrum resource allocation for the web of healthcare things with massive integrating wearable gadgets," Digital Communications and Networks, vol. 11, no. 3, pp. 671–680, Jun. 2025, https://doi.org/10.1016/j.dcan.2024.10.003

[34] S. Malta, P. Pinto, and M. Fernández-Veiga, "Optimizing 5G network slicing with DRL: Balancing eMBB, URLLC, and mMTC with OMA, NOMA, and RSMA," Journal of Network and Computer Applications, vol. 234, p. 104068, Feb. 2025, https://doi.org/10.1016/j.jnca.2024.104068

[35] V. P, S. K, S. R, and G. S, "Dynamic network slicing based resource management and service aware Virtual Network Function (VNF) migration in 5G networks," Computer Networks, vol. 259, p. 111064, Mar. 2025, https://doi.org/10.1016/j.comnet.2025.111064

[36] Y. Li, "Multi-resource joint management strategy for 5 G network slicing based on POMDP," Systems and Soft Computing, vol. 7, p. 200242, Dec. 2025.