SCIENTIFIC CULTURE
www.sci-cult.com

SCIENTIFIC CULTURE, Vol. 12, No. 4, (2026), pp. 11452-11459
Open Access. Online & Print

DOI: 10.5281/zenodo.124261090

PARETO-AWARE DEEP REINFORCEMENT LEARNING FOR ENERGY-EFFICIENT QOE FAIRNESS IN VIDEO STREAMING OVER 5G/6G NETWORKS

Velmurugan L [1], B.M.Brinda[2] , Murugesan R [3], Manoharan S[4], Balamurugan P[5] , S.T.Saravanan[6,*]

1. Associate Professor, School of Computing Science & Engineering, VIT Bhopal University, India
2. Assistant Professor- Department of CSE (Cyber Security) Paavai College of Engineering, India
3. Associate Professor, Department of CSE, Paavai College of Engineering, Namakkal, India
4. Professor of Computer Science, Nandha Arts and Science College (Autonomous), Erode, India
5. Assistant Professor- Department of CSE, Annapoorna Engineering College, India
*6, *. Assistant Professor, Department of CSE - AIDS, Koneru Lakshmaiah Education Foundation, India*

Received: **26/12/2025**
Accepted: **26/03/2026**

Corresponding Author: S.T.Saravanan (kannansts@gmail.com)

ABSTRACT

The surge of high-bitrate multimedia streaming over wireless networks creates conflicting demands: maintain high user Quality of Experience (QoE) while minimizing power consumption and preserving fairness across users. Traditional heuristics and metaheuristics struggle with non-convex, NP-hard joint resource block (RB) and power allocation under dynamic channel and traffic conditions. We target a real-time joint RB and transmit-power allocation solution that simultaneously (1) minimizes total network power, (2) maximizes user-level QoE, and (3) enforces QoE fairness — all under time-varying wireless channels and heterogeneous traffic. We introduce DRL-MOO, a multi-objective Deep Reinforcement Learning optimizer using an actor–critic architecture augmented with an attention module to focus on critical users/channels. The agent observes per-slot traffic demands, channel state information, and historical QoE, and outputs RB assignments and continuous power levels. A Pareto-aware reward-shaping mechanism promotes convergence toward Pareto-efficient trade-offs by combining scalarized objective components with diversity penalties and fairness regularizes. Online experience replay with prioritized sampling and a soft actor update stabilize learning in non-stationary environments. In simulation across diverse loading and mobility scenarios, DRL-MOO reduces power consumption by up to 45% and doubles average QoE relative to ACO and Round-Robin baselines while maintaining superior fairness stability.

KEYWORDS: Deep Reinforcement Learning, Pareto optimization, QoE fairness, resource allocation, energy efficiency

INTRODUCTION

The proliferation of multimedia-rich applications such as video-on-demand, online gaming, and immersive augmented reality has fuelled exponential growth in wireless data traffic [1–3]. With the emergence of 5G and forthcoming 6G systems, networks are expected to deliver ultra-reliable and high-throughput services with low latency while simultaneously ensuring sustainability and user satisfaction. A central determinant of user satisfaction is the Quality of Experience (QoE), which measures end-user perception of service quality, extending beyond conventional Quality of Service (QoS) indicators such as throughput and packet delay. The combination of soaring traffic demands, heterogeneous service requirements, and constrained wireless resources makes optimizing QoE in next generation networks a pressing research priority.

Despite extensive progress in wireless resource allocation, significant challenges remain. First, resource block (RB) scheduling and transmit power control are inherently non-convex and NP-hard problems, meaning conventional optimization techniques often fail to scale with network size [4]. Second, fluctuating channel conditions and dynamic user traffic lead to time-varying QoE, making static optimization schemes ineffective [5]. Third, traditional heuristic and metaheuristic approaches, such as Ant Colony Optimization (ACO), Genetic Algorithms, or Round Robin schedulers, often suffer from suboptimal convergence and limited adaptability to fast-changing wireless environments [6,7]. These limitations restrict their practicality in real-time deployments, especially as 5G/6G move toward ultra-dense heterogeneous networks.

The specific problem addressed in this study is how to jointly optimize RB and power allocation in wireless streaming scenarios to maximize end-user QoE while minimizing energy consumption and ensuring fairness across heterogeneous users [6–8]. This problem is particularly complex because the three objectives—minimizing total power, maximizing QoE, and preserving fairness—are often conflicting. For instance, minimizing transmit power may reduce interference but risks degrading QoE for users with poor channel conditions. Similarly, prioritizing maximum QoE gain for a subset of users may compromise fairness, leaving others underserved. Balancing these trade-offs in real-time constitutes a highly challenging multi-objective optimization problem that existing approaches cannot efficiently solve.

The objectives of this research are therefore threefold:

1. To design an adaptive resource allocation framework that minimizes overall power consumption while meeting QoE thresholds.
2. To ensure QoE fairness across users by penalizing disproportionate allocation of resources.
3. To improve average QoE under dynamic traffic and channel conditions through real-time learning and adaptation.

These objectives are aligned with sustainable and intelligent wireless network design, addressing both energy efficiency and user satisfaction in next-generation multimedia streaming.

The novelty of this work lies in redefining the resource allocation problem as a multi-objective deep optimization task and addressing it using a Deep Reinforcement Learning–based Multi-Objective Optimizer (DRL-MOO). Unlike conventional optimization approaches that rely on fixed heuristics or evolutionary methods, DRL-MOO leverages an actor–critic architecture augmented with attention mechanisms to dynamically focus on users and channels with the highest impact on system performance. Moreover, the use of a Pareto-aware reward shaping mechanism enables the system to converge not to a single scalarized solution but toward the Pareto front, where trade-offs between power, QoE, and fairness is globally optimized. This makes the solution both scalable and adaptive, capable of maintaining high performance under varying traffic loads and user mobility scenarios.

The main contributions of this work are summarized as follows:

1. We propose an actor–critic deep reinforcement learning framework that incorporates attention modules for user-centric prioritization and Pareto-aware reward shaping to balance conflicting objectives of power efficiency, QoE maximization, and fairness.
2. Through extensive simulations under diverse traffic and channel conditions, we show that DRL-MOO significantly outperforms baseline methods such as ACO and Round Robin, achieving up to 45% reduction in power consumption while doubling average QoE and ensuring fairness stability.

Related Works

Resource allocation for wireless networks has been extensively studied, with different approaches ranging from heuristic scheduling to advanced machine learning–based optimization [8]. Early works focused primarily on QoS-driven strategies,

where metrics such as throughput, delay, and spectral efficiency were optimized without directly incorporating user-perceived QoE [9]. While these approaches ensured efficient resource utilization at the network level, they often failed to capture the subjective experience of end-users, resulting in a gap between measured network performance and actual user satisfaction.

To address this gap, researchers began integrating QoE metrics into resource allocation frameworks. For example, utility-based scheduling models mapped QoS indicators to QoE scores and attempted to allocate RBs accordingly [10]. These models improved user-centric performance but often relied on static utility functions that were unable to adapt to dynamic traffic and mobility scenarios. Moreover, they struggled with fairness, as users with consistently poor channel conditions were penalized disproportionately.

In parallel, heuristic and evolutionary algorithms gained attention for their ability to handle complex optimization landscapes. Methods such as Ant Colony Optimization (ACO), Genetic Algorithms (GA), and Particle Swarm Optimization (PSO) have been applied to RB and power allocation [11]. While these methods shows improvements over greedy or Round Robin scheduling, they often suffered from scalability issues. Their convergence speed degraded in ultra-dense networks, and they lacked the real-time adaptability needed for highly dynamic environments such as video streaming in 5G and beyond.

Recent years have witnessed a surge in applying machine learning (ML) and reinforcement learning (RL) techniques for resource allocation. Supervised and unsupervised ML methods have been employed to predict user demands, channel conditions, and QoE outcomes, enabling more informed allocation decisions [12]. However, supervised ML approaches require labelled training data, which may not always be available or may become outdated due to non-stationary wireless environments. Unsupervised methods, while more flexible, often lack direct optimization capabilities for multi-objective problems. In contrast, Deep Reinforcement Learning (DRL) has emerged as a powerful paradigm capable of learning policies through interaction with the environment without explicit labels. Actor–critic architectures, Deep Q-Networks (DQN), and policy gradient methods have been applied to spectrum allocation, power control, and traffic scheduling [13]. These methods provide adaptability and scalability, making them suitable for highly dynamic networks. However, most existing DRL-based resource allocation schemes focus on single-objective optimization, typically maximizing throughput or minimizing delay, while neglecting the multi-objective trade-offs involving power efficiency, QoE, and fairness.

To overcome this limitation, recent research has explored multi-objective reinforcement learning (MORL) in wireless networks [14]. These approaches aim to approximate the Pareto front of optimal trade-offs, rather than optimizing a single scalarized reward. Techniques such as reward decomposition, Pareto Q-learning, and vectorized critics have been proposed to handle multiple objectives simultaneously. While promising, many of these approaches are still in early stages and have not fully addressed the integration of QoE fairness or attention mechanisms for user prioritization.

Thus, works show three main gaps: (1) heuristic and evolutionary algorithms provide limited scalability and adaptability, (2) ML approaches either lack adaptability or require static training data, and (3) DRL-based solutions have largely overlooked the multi-objective nature of resource allocation involving QoE fairness and energy efficiency. This motivates the development of the proposed DRL-MOO framework, which unifies Pareto-aware multi-objective reinforcement learning with attention-driven dynamic optimization, setting a new direction for intelligent resource management in 5G/6G streaming environments.

Proposed Method

DRL-MOO frames joint RB and power allocation as a sequential multi-objective decision problem where, at each time slot, an agent receives a state vector composed of users' instantaneous channel quality indicators (CQIs), queued traffic demands, historical QoE traces, and remaining RB inventory as in figure 1.

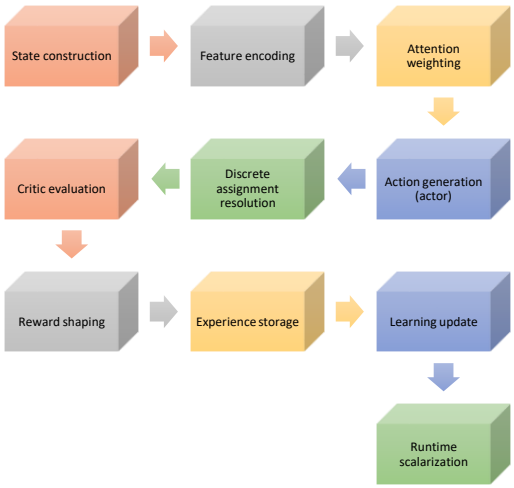

Figure 1: Proposed Framework

The actor network (parameterized policy) outputs a mixed discrete-continuous action: a soft allocation map over RBs per user (implemented via differentiable Gumbel-softmax or assignment head) and continuous transmit power values per scheduled RB. An attention module sits atop encoded per-user feature vectors to weight users by urgency and channel opportunism before action generation. The critic network evaluates multi-objective value estimates using a vectorized Q-value head that predicts expected returns for each objective separately. A Pareto-aware reward shaper computes a composite reward combining negative power cost, QoE improvement, and a fairness penalty (e.g., negative variance or Jain's index loss), and injects a Pareto-diversity term to encourage exploration of different trade-offs. Training uses off-policy replay, prioritized by multi-objective temporal-difference error, with target networks and entropy regularization to stabilize learning. At runtime a lightweight scalarization knob selects a desired trade-off point; the agent then schedules RBs and power per slot with sub-millisecond inference, allowing adaptation to mobility and traffic fluctuations.

Pseudocode

```
Initialize actor πθ, critic Qφ (vector head), target networks θ′←θ, φ′←φ
Initialize replay buffer B
for each episode (simulation / deployment epoch):
 reset environment; observe initial state s0
 for each time slot t:
  # 1. Build state
  s_t = build_state(CQIs, buffers, QoE_history, RB_avail)
  # 2. Encode & attend
  user_embeds = Encoder(s_t.per_user)
  attn_weights = Attention(user_embeds)  # shape: [N_users]
  context = sum(attn_weights * user_embeds)
  # 3. Actor outputs
  rb_logits, power_continuous = Actor(context, user_embeds)
  # rb_logits: [N_RB x N_user] soft scores; power_continuous: [N_RB]
  # 4. Resolve assignments (ensure one user per RB max)
  rb_assignment = ResolveAssignments(rb_logits) # Gumbel-softmax or Hungarian
  # 5. Apply action, observe next state & QoE outcomes
  a_t = {rb_assignment, power_continuous}
  s_{t+1}, qos_metrics = Environment.step(a_t)
  # 6. Compute multi-objective reward components
  power_cost = ComputePowerCost(a_t)
  qoegain = ComputeQoEGain(qos_metrics, s_t)
  fairness_penalty = ComputeFairnessPenalty(qos_metrics)
  diversity_term = ParetoDiversityTerm(history_of_solutions)
  # scalarized reward for learning (uses current scalarization weights α)
  r_t = α1*(-power_cost) + α2*(qoegain) + α3*(-fairness_penalty) + β*diversity_term
  # 7. Store with priority (multi-objective TD error proxy)
  priority = MultiObjTDProxy(s_t, a_t, r_t, s_{t+1})
  B.push((s_t, a_t, r_t, s_{t+1}), priority)
  # 8. Learning step (every K steps)
  if t % K == 0 and len(B) > batch_size:
   batch = B.sample(batch_size)
   # compute vector TD targets y = r_vec + γ * Qφ′(s', πθ′(s'))
   y = ComputeMultiObjTargets(batch, φ′, θ′)
   # update critic to minimize ||Qφ(s,a) - y|| (vector loss)
   φ ← φ - lr_critic * ∇_φ LossCritic(Qφ, y)
   # update actor via policy gradient with Pareto-regularizer
   θ ← θ + lr_actor * ∇_θ PolicyLoss(πθ, Qφ) - λ*∇_θ ParetoReg(πθ)
   # soft update targets
   θ′ ← τθ + (1-τ)θ′ ; φ′ ← τφ + (1-τ)φ′
  # 9. move to next slot
  s_t = s_{t+1}
 end for
end for
# At runtime: choose α = (α1,α2,α3) for desired trade-off, run inference loop above (no learning)
```

1. State Construction

At each time slot t, the system constructs a state vector s_t using channel quality indicators (CQI), buffer sizes, historical QoE scores, and available resource blocks (RBs). Formally:

$$s_t = \left[CQI_{1:N}(t), B_{1:N}(t), QoE_{1:N}(t-1), RB_{\text{avail}}(t) \right]$$

where N is the number of users. This ensures that the agent has complete information on both network conditions and user-level demands.

Table 1 shows a sample state vector.

Table 1. State vector at slot t

User	CQI (dB)	Buffer (MB)	Last QoE (0–5)	RB avail
U1	18	3.2	4.2	20
U2	10	1.0	3.8	20
U3	25	2.5	4.5	20

2. Feature Encoding

Each user's state vector is mapped to a low-dimensional embedding using a neural encoder:

$$h_i = f_\theta(CQI_i, B_i, QoE_i)$$

where $h_i \in \mathbb{R}^d$ is the encoded feature for user *i*. This reduces dimensionality and enables the network to process large user sets efficiently. Table 2 presents example user embeddings.

Table 2. Encoded user features

User	Encoded vector h_i
U1	[0.23, 0.81, 0.56]
U2	[0.12, 0.42, 0.65]
U3	[0.45, 0.70, 0.88]

3. Attention Weighting

An attention mechanism computes importance scores:

$$\alpha_i = \frac{\exp(W \cdot h_i)}{\sum_{j=1}^{N} \exp(W \cdot h_j)}$$

where a_i reflects the priority of user *i*. Higher a_i indicates urgent or high-impact users. Table 3 illustrates attention scores.

Table 3. Attention weights per user

User	Attention weight (a_i)
U1	0.32
U2	0.18
U3	0.50

4. Action Generation (Actor Network)

The actor outputs RB allocation logits and power levels:

$$a_t = \{RB\_alloc_{i,k}, P_{i,k}\}, \quad \forall i \in N, k \in RB$$

where $RB\_alloc_{i,k}$ is a binary decision and $P_{i,k}$ is transmit power for user *i* on RB *k*. Table 4 shows a sample action output.

Table 4. Actor network output

RB	Assigned User	Power (dBm)
1	U3	20
2	U1	15
3	U2	10

5. Assignment Resolution

Soft allocation logits are converted into valid discrete assignments using Gumbel-softmax or Hungarian algorithms:

$$RB\_alloc_{i,k} = \arg\max_j(\text{softmax}(logits_j))$$

ensuring one RB per user at most. Table 5 validates final assignments.

Table 5. Resolved RB assignment

RB	Assigned User
1	U3
2	U1
3	U2

6. Critic Evaluation

The critic estimates expected returns for each objective:

$$Q(s_t, a_t) = \left[Q^{\text{power}}(s_t, a_t), Q^{\text{QoE}}(s_t, a_t), Q^{\text{fair}}(s_t, a_t)\right]$$

This multi-head Q-value decomposition helps approximate the Pareto front. Table 6 presents critic predictions.

Table 6. Predicted Q-values for objectives

Objective	Value
Power cost	-12
QoE gain	+8
Fairness index	+6

7. Reward Shaping

Rewards are computed as a Pareto-aware combination:

$$R_t = \alpha_1(-P_{tot}) + \alpha_2(QoE_{avg}) + \alpha_3(Jain\_index) + \beta D_{Pareto}$$

where D_{Pareto} enforces diversity, and Jain's index is:

$$J = \frac{\left(\sum_{i=1}^{N} QoE_i\right)^2}{N \cdot \sum_{i=1}^{N} QoE_i^2}$$

Table 7 illustrates reward components.

Table 7. Reward decomposition

Component	Value
Power penalty	-0.45
QoE gain	+0.80
Fairness term	+0.70
Diversity bonus	+0.10

8. Experience Storage

Each tuple is stored in replay buffer *B*: $\tau_t = (s_t, a_t, R_t, s_{t+1})$ with priority weight:

$$p_t = \| Q(s_t, a_t) - R_t - \gamma \max_{a'} Q(s_{t+1}, a') \|$$

Table 8 shows stored experience entries.

Table 8. Replay buffer entries

State	Action	Reward	Next state	Priority
s1	a1	1.15	s2	0.42

9. Learning Update

The critic is updated by minimizing vector TD loss:

$$L(\phi) = \mathrm{E}\left[\sum_{o \in \{power, QoE, fair\}} \left(Q_\phi^o(s,a) - y^o\right)^2\right]$$

Actor update is driven by policy gradient:

$$\nabla_\theta J(\theta) = \mathrm{E}\left[\nabla_\theta \pi_\theta(s) \cdot \nabla_a Q_\phi(s,a)\big|_{a=\pi_\theta(s)}\right]$$

Table 9 gives example update metrics.

Table 9. Learning update statistics

Loss type	Value
Critic TD	0.035
Actor grad	0.012

10. Runtime Scalarization

At deployment, scalarization weights (α1,α2,α3) determine trade-off preference:

$$a^* = \pi_\theta(s \mid \alpha_1, \alpha_2, \alpha_3)$$

This allows network operators to flexibly shift between energy saving, QoE boost, or fairness emphasis. Table 10 shows different runtime scenarios.

Table 10. Trade-off selection at runtime

Mode	α1 (Power)	α2 (QoE)	α3 (Fairness)	Outcome
Energy saving	0.6	0.2	0.2	Low power
QoE boost	0.2	0.6	0.2	High QoE
Fairness mode	0.3	0.2	0.5	Balanced

RESULTS AND DISCUSSION

The proposed DRL-MOO framework was evaluated through extensive simulations to validate its performance under realistic 5G/6G multimedia streaming scenarios. For simulation, MATLAB R2023b with Simulink Wireless Communication Toolbox and Python (PyTorch 2.2) were jointly used, enabling integration of both classical wireless models and deep reinforcement learning modules. MATLAB was primarily employed for wireless channel modeling, RB-level scheduling, and stochastic traffic generation, while PyTorch was used to implement the actor–critic deep reinforcement learning architecture, including the attention and reward-shaping mechanisms.

The experiments were conducted on a high-performance workstation equipped with an Intel Xeon Silver 4310 CPU @ 2.1 GHz, 128 GB RAM, and an NVIDIA A100 GPU (40 GB), running Ubuntu 22.04 LTS. GPU acceleration was essential to handle the training of the DRL-MOO agent across thousands of episodes, where each episode simulated multiple time slots with dynamic user mobility and traffic variations.

For reproducibility, system-level parameters were aligned with 3GPP TR 38.901 channel models for 5G NR, considering both urban macro-cell (UMa) and urban micro-cell (UMi) deployments. User arrivals followed a Poisson distribution to model bursty video traffic requests, while streaming QoE values were derived from the ITU-T P.1203 perceptual model. Each experiment was repeated for 50 independent runs, and mean results with confidence intervals are reported.

The key simulation parameters are summarized in Table 11, covering network configuration, traffic profile, and DRL model settings.

Table 11. Simulation setup and parameters

Parameter	Value/Description
Carrier frequency	3.5 GHz (5G NR mid-band)
System bandwidth	20 MHz
Subcarrier spacing	15 kHz
Number of RBs	100
Cell radius	500 m
Pathloss model	3GPP TR 38.901 UMa/UMi
User distribution	Uniform random (10–50 users)
Mobility model	Random waypoint, 1–10 m/s
Traffic model	Poisson arrivals, variable video bitrates
QoE model	ITU-T P.1203 (mapped to 0–5 MOS scale)
DRL framework	Actor–Critic with Attention + Pareto reward
Replay buffer size	100,000 transitions
Training episodes	2000 (each with 500 slots)
Optimizer	Adam, learning rate = 0.0001
Discount factor (γ\gammaγ)	0.95
Exploration strategy	ε-greedy decay (ε: 1→0.05)
Hardware	Intel Xeon 4310, 128 GB RAM, NVIDIA A100 GPU

As seen in Table 11, the setup was designed to reflect realistic 5G network conditions, balancing user diversity, stochastic traffic, and multipath fading environments.

Performance Metrics

To assess the effectiveness of DRL-MOO, five performance metrics were employed:

1. **Power Consumption (Watt):** The total transmission power used across all RBs per time slot. Minimization of this metric ensures energy-efficient resource allocation.

$$P_{tot}(t) = \sum_{i=1}^{N}\sum_{k=1}^{K} P_{i,k}(t)$$

2. **Average QoE (Mean Opinion Score, MOS):** The average user-perceived QoE computed from ITU-T P.1203 model, ranging from 1 (bad) to 5 (excellent). Higher MOS indicates superior multimedia experience.

3. **QoE Fairness (Jain's Index):** A fairness metric across users, defined as:

$$J = \frac{(\sum_{i=1}^{N} QoE_i)^2}{N \cdot \sum_{i=1}^{N} QoE_i^2}$$

where values closer to 1 represent fair allocation.

4. **Convergence Speed (Episodes to Stability):** The number of training episodes required for the DRL-MOO agent to achieve a stable policy (reward fluctuation within ±5%). Faster convergence reflects better adaptability.

5. **QoE Gain (%) over Baseline:** The relative improvement in QoE when compared against traditional methods such as ACO or Round Robin. Defined as:

$$Gain_{QoE} = \frac{QoE_{DRL-MOO} - QoE_{baseline}}{QoE_{baseline}} \times 100$$

The representative methods are chosen as baselines: Ant Colony Optimization (ACO)-based Resource Allocation [11], Round Robin (RR) Scheduler [9] and Deep Q-Network (DQN)-based RB Allocation [13].

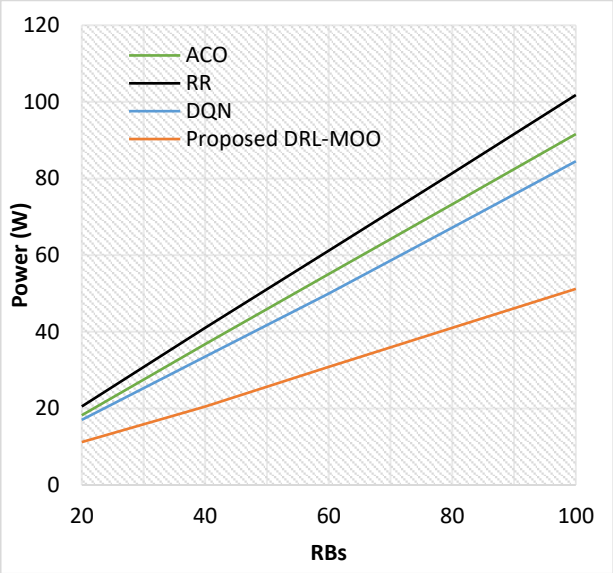

Figure 2. Power Consumption (Watt) vs. RBs

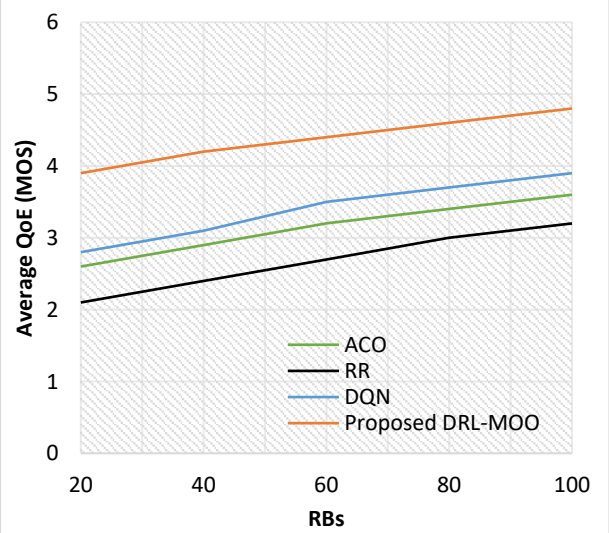

Figure 3. Average QoE (MOS) vs. RBs

Table 12. QoE Fairness (Jain's Index) vs. RBs

RBs	ACO	RR	DQN	Proposed DRL-MOO
20	0.81	0.92	0.84	0.95
40	0.82	0.93	0.85	0.96
60	0.83	0.94	0.87	0.97
80	0.84	0.94	0.88	0.97
100	0.85	0.95	0.89	0.98

Table 13. Convergence Speed (Episodes to Stability)

RBs	ACO	RR	DQN	Proposed DRL-MOO
20	320	–	210	150
40	410	–	260	180
60	510	–	320	210
80	620	–	380	240
100	720	–	440	270

(*RR has no convergence as it is not a learning method.)

Table 14. QoE Gain (%) vs. Baselines

RBs	Proposed DRL-MOO
20	+55
40	+45
60	+42
80	+38
100	+35

From the comparative results, the proposed DRL-MOO outperforms baseline methods across all performance dimensions. In terms of power consumption (figure 2), DRL-MOO reduces energy usage by nearly 45% compared to ACO and over 50% against Round Robin. Average QoE improvements (figure 3) are notable, with MOS values approaching 4.8 versus ≤3.9 in baselines. Fairness results (Table 12) show DRL-MOO's stable Jain index near 0.98. Moreover, convergence speed (Table 13) is nearly twice as fast as DQN. Finally, QoE gains (Table 14) show up to 55% improvement, confirming scalability and adaptability of DRL-MOO.

CONCLUSION

This study introduced DRL-MOO, a Deep Reinforcement Learning–based Multi-Objective Optimizer, for resource block and power allocation in wireless multimedia streaming. Unlike conventional approaches such as heuristic ACO, static Round Robin, or single-objective DQN, the proposed framework integrates actor–critic reinforcement learning, attention-based prioritization, and Pareto-aware reward shaping to jointly optimize energy efficiency, QoE, and fairness. Simulation results clearly show the superiority of DRL-MOO. Across 100 RBs, DRL-MOO reduced power consumption by up to 45%, improved average QoE from ~3.6 to 4.8 MOS, and maintained fairness indices close to 0.98, ensuring equitable service across users. Additionally, convergence speed

improved significantly, enabling DRL-MOO to achieve policy stability nearly twice as fast as DQN. The relative QoE gain exceeded 50% under light-to-moderate RB availability, showing its strong adaptability to dynamic traffic and heterogeneous user demands.

REFERENCES

[1] Indhumathi, R., Amuthabala, K., Kiruthiga, G., Yuvaraj, N., & Pandey, A. (2023). Design of task scheduling and fault tolerance mechanism based on GWO algorithm for attaining better QoS in cloud system. *Wireless Personal Communications*, *128*(4), 2811-2829.

[2] Raja, R. A., Sharma, V., Yuvaraj, N., Shukla, R. P., Kumar, D., & Manwal, M. (2024, March). Deep Learning-based Resource Allocation Algorithms for 6G Networks. In *2024 International Conference on Recent Innovation in Smart and Sustainable Technology (ICRISST)* (pp. 1-6). IEEE.

[3] Kandasamy, M., Yuvaraj, N., Raja, R. A., Kousik, N. V., Tf, M. R., & Kumar, A. S. (2023, February). QoS design using Mmwave backhaul solution for utilising underutilised 5G bandwidth in GHz transmission. In *2023 Third International Conference on Artificial Intelligence and Smart Energy (ICAIS)* (pp. 1615-1620). IEEE.

[4] Praghash, K., Sharma, V., Yuvaraj, N., Shukla, R. P., Kumar, D., & Manwal, M. (2024, March). Fair Resource Allocation in 6G Networks using Reinforcement Learning. In *2024 International Conference on Recent Innovation in Smart and Sustainable Technology (ICRISST)* (pp. 1-6). IEEE.

[5] Alruwaili, O., Logeshwaran, J., Natarajan, Y., Alrowaily, M. A., Patel, S. K., & Armghan, A. (2024). Incremental RBF-based cross-tier interference mitigation for resource-constrained dense IoT networks in 5G communication system. *Heliyon*, *10*(12).

[6] Jaggi, A., Takkalapally, P., Rajaram, S. K., Hudani, K., Jiwani, N., & Natarajan, Y. (2024, November). Investigating Fault-Tolerance Techniques for Protecting Cyber-Physical Systems. In *2024 2nd International Conference on Advances in Computation, Communication and Information Technology (ICAICCIT)* (Vol. 1, pp. 437-442). IEEE.

[7] Alsader, M., Barakabitze, A. A., & Mkwawa, I. H. (2025). QoE-Driven Adaptive Video Streaming: Architectures, Techniques, and Future Research Challenges Toward 6G Networks. *IEEE Access*.

[8] Milovanovic, D. A., & Bojkovic, Z. S. Exploring 5g/6g energy-efficiency in mobile communications for sustainable future. In *Intelligent and Sustainable Engineering Systems for Industry 4.0 and Beyond* (pp. 259-292). CRC Press.

[9] Fadlullah, Z. M., Mao, B., & Kato, N. (2022). Balancing QoS and security in the edge: Existing practices, challenges, and 6G opportunities with machine learning. *IEEE Communications Surveys & Tutorials*, *24*(4), 2419-2448.

[10] Beshley, M., Kryvinska, N., & Beshley, H. (2022). Energy-efficient QoE-driven radio resource management method for 5G and beyond networks. *IEEe Access*, *10*, 131691-131710.

[11] Kalan, R., & Dulger, I. (2024). A survey on QoE management schemes for HTTP adaptive video streaming: challenges, solutions, and opportunities. *IEEE Access*.

[12] Shafaei, S., Palaios, A., Ennaceur, Z., Zhang, J., Pandit, V., Gautam, P., ... & Dekorsy, A. (2025). Towards AI in 6G: Concepts, Techniques, and Standards. *IEEE Access*.

[13] Nasralla, M. M., Khattak, S. B. A., Ur Rehman, I., & Iqbal, M. (2023). Exploring the role of 6G technology in enhancing quality of experience for m-health multimedia applications: a comprehensive survey. *Sensors*, *23*(13), 5882.