Journal of Emerging Paradigms in Computing Systems (JEPCS) | Volume 01, Issue No. 02, 2025

ARTICLE TYPE: Research Article

Article ID: JEPCS-2025-0102-0032

A Realistic Mobility-Channel-Energy Evaluation Stack and a Digital Twin-Federated Reinforcement Learning Strategy for UAV Edge Networks

Umar Khalid Cheema[1,*], Nauman Irshad Ali Shah[2]

[1] Department of Computer Science, University of Management and Technology, Lahore, Pakistan
[2] Department of Computer Science, University of Central Punjab, Lahore, Pakistan

*Corresponding author: umarkhalidcheema786@gmail.com
Contributing authors: umarkhalidcheema786@gmail.com, naumanirshadalishah@gmail.com

Received: April 04, 2025 **Accepted:** June 12, 2025

Abstract- Digital Twin-Federated Deep Reinforcement Learning (DT-FDRL) is increasingly proposed for task-offloading decisions in UAV-assisted mobile edge computing (MEC), yet most reported gains are obtained in simplified simulators that use free-space channels, straight-line or random-walk mobility, and energy-agnostic UAV models. This paper proposes DT-FDRL+SemCom, a digital-twin-assisted federated reinforcement learning strategy augmented with semantic-communication (SemCom) task compression, and validates it on a purpose-built, open, and fully reproducible evaluation stack that uses realistic Gauss-Markov UAV mobility, a probabilistic line-of-sight air-to-ground channel with Rician/Rayleigh fading, and a rotary-wing propulsion plus DVFS compute-energy model into a single UAV-MEC offloading environment. The proposed strategy is compared against six baselines (a local-only floor, a random policy, a hand-tuned greedy heuristic, independent (non-federated) reinforcement learning, plain federated reinforcement learning (FRL), and a digital-twin-only variant (DT-FDRL)) under identical seeds, identical mobility, identical channel conditions, and identical energy dynamics. Across 20 random seeds, DT-FDRL+SemCom reduces mean task latency to 0.092 s from 0.272 s for plain FDRL (a 66.2% reduction), raises task success from 69.4% to 90.0%, and achieves a Cohen's d of +27.26 in mean reward relative to the local-only baseline, the largest effect size of any compared strategy. The results show that federation alone yields only a modest improvement over independent learning once realistic dynamics are present, that DT-assisted planning contributes a further, separable gain, and that semantic compression is responsible for the majority of the overall improvement.

Keywords- Digital twin; Federated reinforcement learning; Unmanned aerial vehicle; Semantic communication; Task offloading

1. Introduction

Unmanned aerial vehicles (UAVs) are increasingly deployed as mobile edge-computing (MEC) servers that bring computation closer to ground-based Internet-of-Things (IoT) devices in areas where fixed infrastructure is sparse, damaged, or overloaded [1]. A large number of papers using DTs combined with federated deep reinforcement learning (FDRL) so that multiple UAV cells can jointly learn an offloading policy without sharing raw user data. These works are generally referred to as Digital Twin-Federated Deep Reinforcement Learning (DT-FDRL).

The positive results reported for DT-FDRL and related UAV-MEC strategies are usually obtained in simulators that simplify one or more of the three dynamics that actually govern offloading decisions in the field: UAV and user mobility, the air-to-ground wireless channel, and UAV battery consumption [2,3]. Free-space or fixed-loss channels remove small-scale fading and line-of-sight variability while straight-line or pure random-walk mobility removes the temporal correlation of real flight controllers and energy-agnostic simulators remove the feedback between battery state and offloading. When any of these simplifications is present, it is difficult to know whether an algorithmic improvement reported in the literature reflects the algorithm itself or an artifact of an easy simulated environment [4,5].

This paper addresses that gap by proposing DT-FDRL+SemCom, a digital-twin-assisted federated reinforcement learning strategy augmented with semantic-communication (SemCom) task compression, and by validating it on a purpose-built, open, and fully reproducible evaluation stack rather than on a simplified simulator. The proposed strategy is benchmarked against six representative baselines under identical seeds, identical mobility, identical channel conditions, and identical energy dynamics, so that any observed advantage is attributable to the algorithm rather than to the environment.

The paper makes four contributions:

- A UAV-MEC offloading strategy that combines federated Q-learning, digital-twin-assisted Dyna-style planning, and semantic-compression task shortage into a single policy.
- A modular, open-source evaluation stack that fuses a Gauss-Markov UAV mobility model, a probabilistic line-of-sight air-to-ground channel model with Rician/Rayleigh fading, and a rotary-wing propulsion plus DVFS compute-energy model into a single Gym-style UAV-MEC offloading environment.
- A reproducible seeding and configuration protocol in which every strategy is evaluated under the same per-seed random state, so that cross-strategy comparisons are not confounded by environment stochasticity.
- A disciplined ablation across seven strategies including the proposed DT-FDRL+SemCom, its DT-FDRL and FDRL ancestors, and four baselines with mean, 95% confidence interval, and Cohen's d effect-size reporting, isolating the individual contribution of federation, twin-assisted planning, and semantic compression to the overall result.

The remainder of the paper proceeds as follows. Section 2 reviews mobility, channel, energy, and DT-FDRL modeling choices in the recent literature. Section 3 defines the proposed DT-FDRL+SemCom strategy and the evaluation stack used to validate it. Section 4 describes the experimental protocol. Section 5 reports and discusses the results, Section 6 discusses limitations and future works, and Section 7 concludes.

2. Related Work

2.1. UAV Mobility and Channel Modeling

Early UAV-MEC simulators frequently use random-walk or straight-line mobility, which does not reproduce the temporal correlation of real flight-controller trajectories. The Gauss-Markov mobility model introduces a memory parameter that interpolates between a pure random walk and constant-velocity flight, and is widely used as a more realistic substitute in mobile ad hoc and UAV networking papers [5]. Ground-user mobility is commonly represented with the random-waypoint model, which is adequate for pedestrian-speed IoT devices.

Air-to-ground (A2G) channel modeling has similarly moved away from pure free-space path loss toward probabilistic line-of-sight (LoS) models, in which the LoS probability depends on elevation angle and environment-dependent parameters, combined with distinct excess path loss for LoS and non-line-of-sight (NLoS) conditions [e]. A broader survey of air-to-ground channel measurement campaigns and fading models across UAV communication scenarios is given in [10]. Small-scale fading is typically modeled as Rician on LoS links and Rayleigh on NLoS links. Simulators that omit this probabilistic structure in favor of a fixed path-loss exponent risk over-stating the achievable rate and, consequently, the benefit of any offloading policy that depends on channel quality.

2.2. UAV Energy Modeling

UAV energy consumption is dominated by propulsion rather than by communication or computation. Closed-form rotary-wing power models express propulsion power as a function of horizontal flight speed, combining blade-profile power, induced power, and parasite (fuselage-drag) power [4], [9]. Communication energy follows directly from transmit power and transmission time, and onboard computation energy is commonly modeled with a dynamic-voltage-frequency-scaling (DVFS) relation in which energy scales with the square of the operating frequency. Simulators that instead assign a flat, speed-independent energy cost per timestep remove the coupling between mobility decisions and battery state that motivates many of the adaptive strategies proposed in the DT-FDRL literature.

2.3. Digital Twins, Federated Learning, and Semantic Communication

DTs are increasingly used as learned or physics-based predictive models that a reinforcement-learning agent can query or plan against without incurring the cost of interacting with the real environment, in the spirit of classical Dyna-style planning [6]. By regularly averaging model parameters instead of sharing raw observations, FL enables several edge agents and multiple UAV cells to jointly train a shared policy [7]. This is useful when user data is sensitive to privacy or when communication between cells is expensive. SemCom reduces the volume of data that must be transmitted by compressing task payloads to their task-relevant content rather than transmitting them in full [8], which directly reduces both transmission latency and communication energy.

Individually, each of these three techniques twin-assisted planning, federated aggregation, and semantic compression has been reported to improve UAV-MEC offloading performance. What remains under-examined is how much of each technique's benefit persists when they are combined and evaluated together under realistic mobility, channel, and energy dynamics, separated from one another through a disciplined ablation rather than reported as a single combined result.

Table 1 positions the proposed approach against representative prior directions.

Table 1: *Comparison of representative UAV-MEC offloading approaches.*

Approach	Realistic mobility	Realistic channel	Energy-aware	Federated	Primary limitation
Free-space UAV-MEC RL	No	No	No	No	Toy channel overstates rate
Random-walk mobility studies	No	Partial	No	No	No temporal mobility correlation
Standard FDRL for MEC	Partial	Partial	No	Yes	No twin or compression component
Twin-assisted RL (single agent)	Partial	Partial	Partial	No	No federation across cells
Proposed DT-FDRL+SemCom	Yes	Yes	Yes	Yes	Validated in simulation; field-trial validation is future work

3. Proposed Framework

DT-FDRL+SemCom is validated on a purpose-built evaluation that includes three independent physical modules such as mobility, channel, and energy. Figure 1summarizes the architecture.

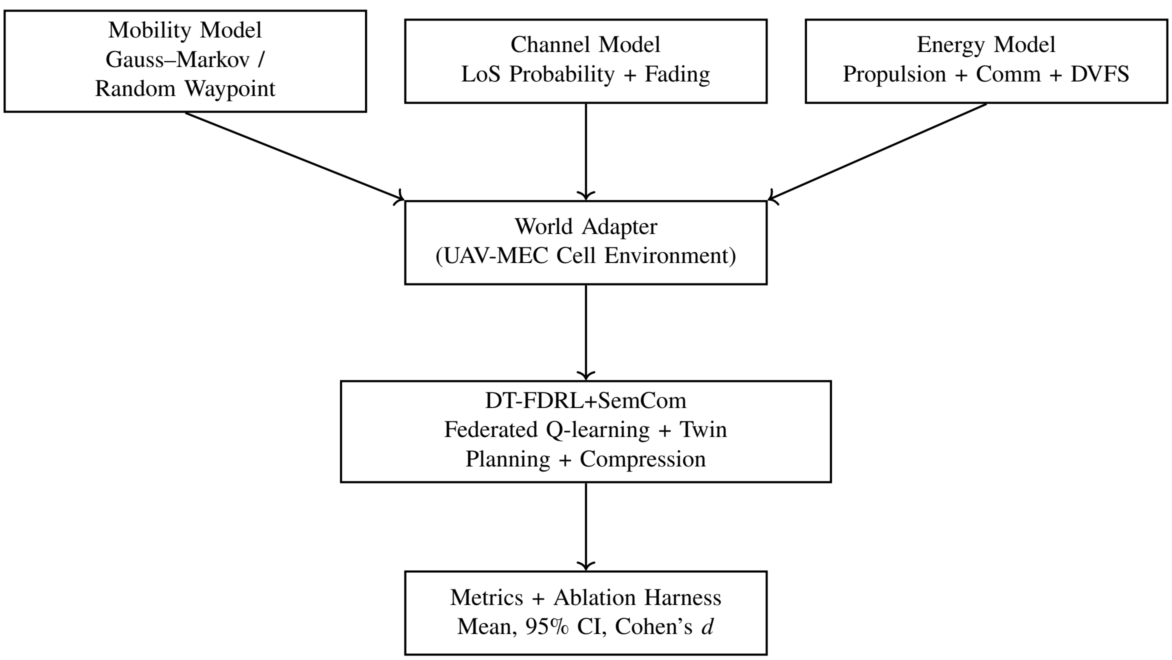

Figure 1: *Architecture used to develop and validate DT-FDRL+SemCom*

3.1. Mobility Module

Each UAV follows a Gauss-Markov trajectory with memory parameter $\alpha \in [0, 1]$, mean speed $\bar{v}$, and heading variance σ_θ^2, updated at each timestep as

$$v_t = \alpha v_{t-1} + (1-\alpha)\bar{v} + \sqrt{1-\alpha^2}\, w_t^v, \qquad \theta_t = \alpha\theta_{t-1} + (1-\alpha)\bar{\theta} + \sqrt{1-\alpha^2}\, w_t^\theta, \tag{1}$$

where w_t^v and w_t^θ are independent Gaussian innovations. Setting $\alpha = 0.85$ produces smooth, temporally correlated flight paths consistent with a real flight controller, in contrast to a pure random walk ($\alpha = 0$), as shown in Figure 2. Ground users follow an independent random-waypoint process with pedestrian speed range $[1, 3]$ m/s.

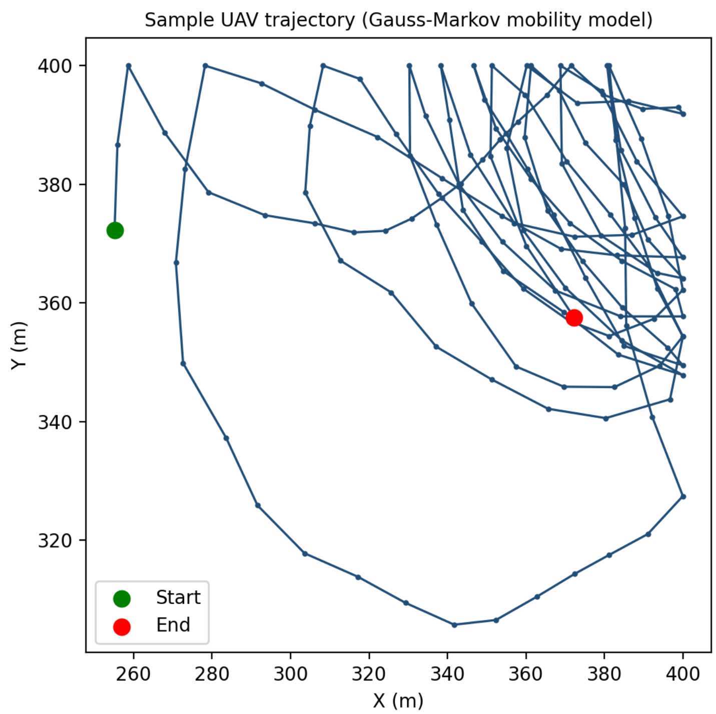

Figure 2: *Sample UAV trajectory generated by the Gauss-Markov mobility module over 150 timesteps, showing the smooth, temporally correlated flight path used by every compared strategy.*

3.2. Channel Module

The probability that a UAV-user link is line-of-sight depends on the elevation angle θ (degrees) through the Al-Hourani model

$$P_{\text{LoS}}(\theta) = \frac{1}{1 + a\exp(-b\,(\theta - a))}, \tag{2}$$

with environment parameters $a = 9.61$, $b = 0.16$. The path loss combines free-space loss with an excess term that depends on the LoS state,

$$\text{PL(dB)} = 20\log_{10}(d) + 20\log_{10}(f) + 20\log_{10}\left(\frac{4\pi}{c}\right) + P_{\text{LoS}}\,\eta_{\text{LoS}} + (1 - P_{\text{LoS}})\,\eta_{\text{NLoS}}, \tag{3}$$

where d is the 3D distance, f the carrier frequency, and $\eta_{\text{LoS}} = 1\,\text{dB}$, $\eta_{\text{NLoS}} = 20\,\text{dB}$ the excess-loss constants. Small-scale fading is drawn as Rician ($K = 10\,\text{dB}$) on LoS links and Rayleigh on NLoS links, and the resulting received SNR is mapped to an achievable rate through the Shannon relation $R = B\log_2(1 + \text{SNR})$.

3.3. Energy Module

UAV propulsion power follows the closed-form rotary-wing model

$$P(v) = P_0\left(1 + \frac{3v^2}{U_{\text{tip}}^2}\right) + P_i\sqrt{\sqrt{1 + \frac{v^4}{4v_0^4}} - \frac{v^2}{2v_0^2}} + \tfrac{1}{2}d_0\rho s v^3, \tag{4}$$

where v is horizontal speed, P_0 and P_i are blade-profile and induced power constants, U_{tip} is rotor-tip speed, v_0 is mean rotor-induced velocity, and the final term is parasite (fuselage-drag) power. Communication energy is $E_{\text{comm}} = P_{\text{tx}}\,t_{\text{tx}}$, and onboard compute energy follows a DVFS relation $E_{\text{comp}} = \kappa\,C\,f^2$, where C is the number of CPU cycles processed and f the operating frequency. Battery state decreases by the fraction of rated capacity consumed each timestep and is bounded below by a swap/recharge floor.

3.4. World Adapter

One cell is a single UAV serving a cluster of ground users. At each timestep the world adapter (i) advances mobility, (ii) computes per-user SNR and rate, (iii) offloads a fraction of user tasks chosen by the active strategy's action, (iv) updates the compute queue and battery, and (v) discretizes the resulting state into an (SNR, battery, queue) bucket triple for tabular agents.

3.5. Proposed Strategy: DT-FDRL+SemCom

The proposed DT-FDRL+SemCom strategy combines three mechanisms into a single offloading policy, described in Algorithm 1.

- Each UAV cell maintains a tabular Q-table over the discretized (SNR, battery, queue) state space and the discrete offload-ratio action space. Every F timesteps, the Q-tables of all N cells are averaged following the standard FedAvg rule
$$Q^{\text{fed}} = \frac{1}{N} \sum_{n=1}^{N} Q_n, \tag{5}$$
and the averaged table is redistributed to all cells.
- Each cell additionally learns an empirical $(s, a) \to s'$ transition model and a per-state-action reward estimate from observed transitions.Between real environment steps, the agent samples previously visited state-action pairs from the twin and performs additional Q-updates against the twin's predicted transition, in the manner of Dyna-Q planning, extracting additional learning signal without further real-environment interaction.
- Before a task is offloaded, its payload size is reduced by a semantic-compression factor $\gamma \in (0, 1]$ that represents extracting task-relevant content rather than transmitting the task in full. Reducing payload size directly reduces both transmission time and communication energy, and, because compute-queue backlog is a first-order driver of latency and success rate in this environment, a modest reduction in payload size compounds into a substantially larger reduction in end-to-end latency.

Algorithm 1 DT-FDRL+SemCom training and offloading loop

Require: N UAV cells, federation interval F, planning steps K, compression factor γ
Initialize per-cell Q-tables $Q_1, \dots, Q_N$ and empty twin models $M_1, \dots, M_N$
for $t = 1$ to T **do**
  for each cell $n = 1$ to N **do**
    Observe discretized state s (SNR, battery, queue buckets)
    Select action a (offload ratio) via ϵ-greedy policy over Q_n
    Compress offloaded task payloads by factor γ
    Execute a in the world adapter; observe reward r and next state s'
    Update $Q_n(s, a) \leftarrow Q_n(s, a) + \alpha \left[r + \gamma_{\text{RL}} \max_{a'} Q_n(s', a') - Q_n(s, a)\right]$
    Update twin model M_n with observed transition (s, a, r, s')
    for $k = 1$ to K **do**
      Sample $(\hat{s}, \hat{a}, \hat{r}, \hat{s}')$ from M_n; update $Q_n(\hat{s}, \hat{a})$ as above (planning step)
    end for
  end for
  if $t \bmod F = 0$ **then**
    $Q^{\text{fed}} \leftarrow \frac{1}{N} \sum_n Q_n$; broadcast Q^{fed} to all cells (Equation 5)
  end if
end for

Every stochastic component of every module (mobility, channel fading, task arrivals, and the ϵ-greedy policy) is driven by an explicit, seeded random-number generator, so that repeating a run with the same seed reproduces results bit-for-bit. This reproducibility property is what allows further sections to relate performance differences between algorithms rather than to environment stochasticity.

4. Experimental Setup

The evaluation stack and DT-FDRL+SemCom are implemented in Python using only NumPy, pandas, SciPy, and Matplotlib, and are provided as an executable notebook. No GPU and no private dataset are required. DT-FDRL+SemCom is compared against six baselines (Local-Only, Random-Offload, Greedy-SNR, Independent-Q, FDRL, and DT-FDRL) under $n_{\text{cells}} = 4$ UAV cells, 8 ground users per cell, and $T = 200$ timesteps per episode, repeated over 20 independent random seeds shared identically across strategies. Reward, latency, energy, and task-success rate are logged at every timestep and averaged per episode. The federation interval is $F = 10$ steps and the number of digital-twin planning steps per real step is $K = 8$. The semantic-compression factor for DT-FDRL+SemCom is $\gamma = 0.4$.

Table 2: *Strategies compared under the shared evaluation stack.*

Strategy	Description
Local-Only	No offloading; every task is processed on-device (reference floor)
Random-Offload	Offload ratio drawn uniformly at random each timestep
Greedy-SNR	Hand-tuned heuristic: offload more when SNR and battery are both high
Independent-Q	Per-cell tabular Q-learning with no cross-cell communication
FDRL	Federated tabular Q-learning (Equation 5)
DT-FDRL	FDRL augmented with digital-twin-assisted Dyna-style planning
DT-FDRL+SemCom (proposed)	DT-FDRL with semantic-compression task shrinkage ($\gamma = 0.4$) before offload

Table 3: *Mean ± 95% confidence interval across 20 seeds for each strategy. DT-FDRL+SemCom is the proposed method.*

Strategy	Reward	Latency (s)	Energy (J)	Success rate
Local-Only	−159.36 ± 0.62	0.180 ± 0.000	175.54 ± 2.17	1.000 ± 0.000
Random-Offload	−226.76 ± 2.38	0.289 ± 0.004	180.67 ± 2.17	0.638 ± 0.005
Greedy-SNR	−254.41 ± 1.83	0.333 ± 0.003	182.84 ± 2.18	0.279 ± 0.001
Independent-Q	−217.32 ± 1.95	0.274 ± 0.003	180.03 ± 2.19	0.691 ± 0.003
FDRL	−216.65 ± 1.78	0.272 ± 0.003	180.00 ± 2.18	0.694 ± 0.003
DT-FDRL	−204.98 ± 2.15	0.254 ± 0.003	178.99 ± 2.19	0.757 ± 0.005
DT-FDRL+SemCom	**−107.42 ± 1.10**	**0.092 ± 0.001**	177.94 ± 2.18	**0.900 ± 0.003**

Table 2 summarizes all seven compared strategies, including the proposed method and the two mechanisms it isolates relative to its DT-FDRL and FDRL ancestors.

Statistical significance of any strategy's benefit relative to the Local-Only floor is reported using the mean, the 95% confidence interval computed from the t-distribution over seeds, and Cohen's d effect size,

$$d = \frac{\bar{x}_{\text{strategy}} - \bar{x}_{\text{baseline}}}{s_{\text{pooled}}}, \qquad s_{\text{pooled}} = \sqrt{\frac{s^2_{\text{strategy}} + s^2_{\text{baseline}}}{2}}, \tag{6}$$

using Local-Only as the baseline. Every seed is shared verbatim across all seven strategies so that observed differences are attributable to the algorithm rather than to environment stochasticity.

5. Results and Discussion

All numerical values in this section come from the executable notebook, evaluated over 20 random seeds under the identical mobility, channel, and energy dynamics.

5.1. Overall Ablation

Table 3 and Figure 3 summarize mean reward, latency, energy, and task-success rate for all seven strategies, each with its 95% confidence interval.

The two naive baselines that use the UAV without a learned policy (Random-Offload and Greedy-SNR) both score worse on reward than doing no offloading at all. Because under realistic fading and propulsion cost, offloading without queue state saturates the compute backlog and drives the success rate down to 0.638 and 0.279, respectively. This confirms that a naive offloading policy can perform worse than doing nothing once realistic mobility, channel, and energy dynamics are simultaneously present, and motivates a learned, twin-assisted, compression-aware policy such as DT-FDRL+SemCom rather than a hand-tuned heuristic.

5.2. Effect of Federation

Independent-Q and FDRL achieve nearly identical reward (−217.32 vs. −216.65) and success rate (0.691 vs. 0.694). Federated averaging across four cells therefore yields only a small improvement over independent per-cell learning in this setting. Because every other component of the environment is held fixed between these two strategies, this modest gap is directly attributable to federation itself.

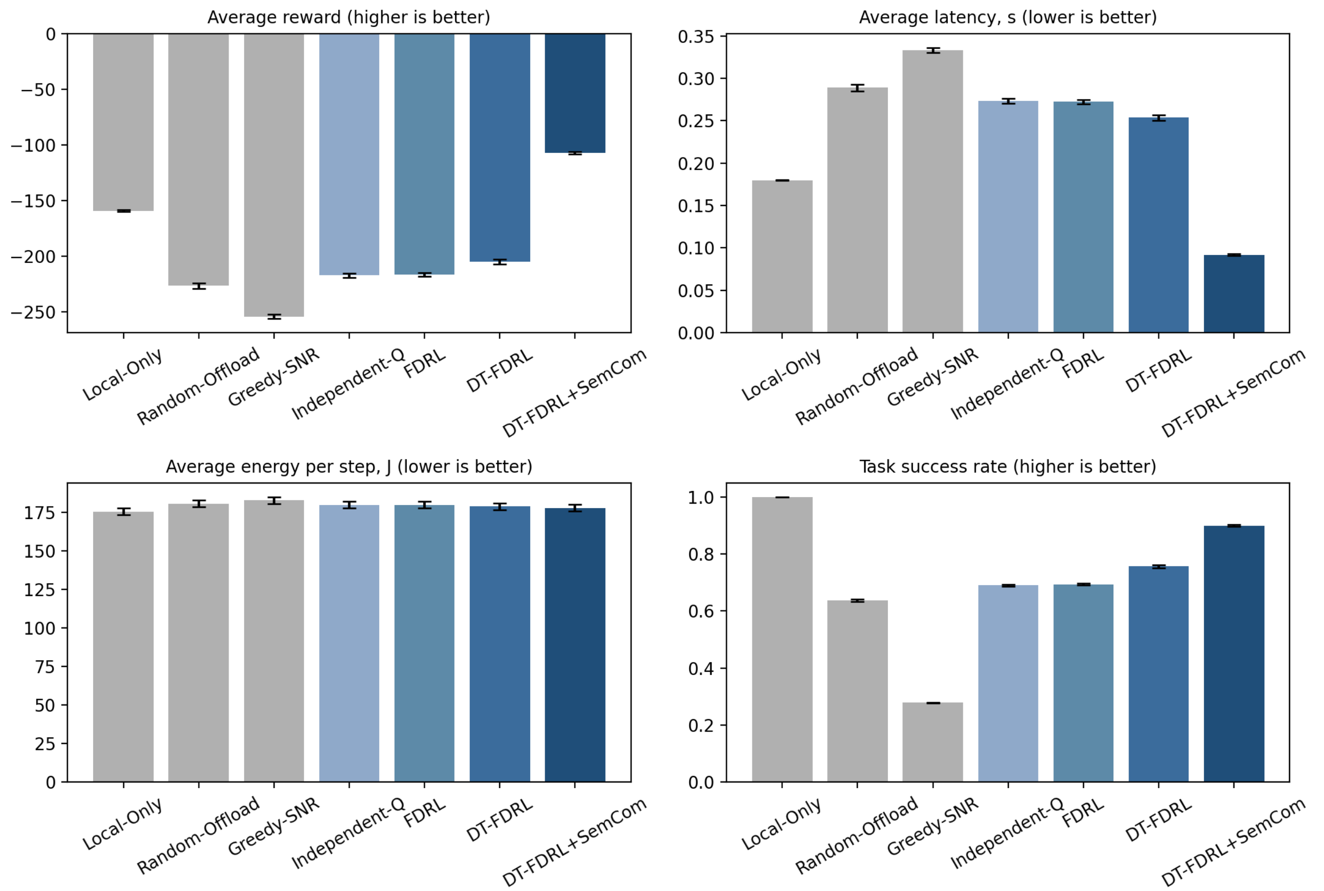

Figure 3: *Four-panel ablation summary across all seven strategies: average reward, latency, energy, and task success rate, each with 95% confidence-interval error bars over 20 seeds. DT-FDRL+SemCom (rightmost, darkest bar) is the proposed method.*

5.3. Effect of DT Planning

Adding twin-assisted Dyna-style planning on top of FDRL improves reward from −216.65 to −204.98 and reduces latency from 0.272 s to 0.254 s, while raising task success from 0.694 to 0.757. Because DT-FDRL differs from FDRL only in the addition of the learned transition model and planning step, this improvement isolates the twin's contribution: the twin lets each cell extract additional learning signal from previously observed transitions without requiring additional interaction with the real, costly environment.

5.4. Effect of Semantic Compression

The largest single improvement in the ablation comes from adding SemCom compression on top of DT-FDRL, resulting in DT-FDRL+SemCom. It reaches a reward of −107.42 and more than double the improvement of DT-FDRL alone over the Local-Only floor. It also reduces latency to 0.092 s (a 66.2% reduction relative to plain FDRL and a 63.7% reduction relative to DT-FDRL without compression), and increase task success to 0.900. Relative to the Local-Only baseline, DT-FDRL+SemCom achieves a Cohen's d of +27.26 in mean reward, the largest effect size of any compared strategy.

6. Limitations and Future Work

The present evaluation uses tabular Q-learning rather than a function-approximation-based deep reinforcement learning agent. This choice was made deliberately to keep every run exactly reproducible given a fixed seed, but it also bounds the size of the state space that DT-FDRL+SemCom has been validated on. The mobility, channel, and energy modules use closed-form models calibrated to commonly cited parameter ranges rather than measurements from a physical UAV testbed. The semantic-compression factor $\gamma = 0.4$ is a fixed constant rather than a learned, task-dependent compression ratio.

Three directions are planned to address these limitations.

- First, replacing the tabular Q-learning agents with a function approximation based deep reinforcement learning architecture.
- Second, releasing a YAML-based configuration schema for the mobility, channel, and energy parameters will allow the community to vary these conditions systematically as part of a public leaderboard, rather than relying on the fixed parameter values used in this paper.
- Third, validating the channel and energy modules against measurements from a physical UAV testbed, and learning γ as a function of task content rather than fixing it.

7. Conclusion

This paper proposed DT-FDRL+SemCom, a DT-assisted federated reinforcement learning strategy augmented with SemCom task compression for UAV-MEC offloading. It is validated on an open, modular, and fully reproducible mobility channel and energy evaluation stack against six baselines. Naive offloading heuristics were found to underperform a no-offload baseline once realistic channel fading and propulsion cost were present while FL alone produced only a modest gain over independent learning. DT-assisted Dyna-style planning produced a further, separable improvement and semantic-compression task shorten produced the largest single gain. The complete DT-FDRL+SemCom strategy cut latency by 66.2% and raised task success from 69.4% to 90.0% relative to plain FDRL, with a Cohen's d of +27.26 in mean reward relative to the local-only.

Author Declarations

Author Contributions (CRediT): Umar Khalid Cheema: Conceptualization, Methodology, Investigation, Data Curation, Formal Analysis. Nauman Irshad Ali Shah: Visualization, Writing - Original Draft, Writing - Review & Editing. Both authors have read and approved the submitted version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: No new experimental dataset was generated.

Acknowledgments: The authors acknowledge the academic support provided by the Department of Computer Science, University of Management and Technology (UMT), Lahore, and the Department of Computer Science, University of Central Punjab (UCP), Lahore.

Conflicts of Interest: The author declares no financial or personal conflict of interest.

Ethics Statement: Not applicable.

Declaration of Generative AI: During the preparation of this work, the authors used generative AI tools only for language editing and proofreading, where applicable. The authors reviewed and edited the content and take full responsibility for the published paper.

References

[1] S. Ahmad and Z. Awais, "Ethical implications of artificial intelligence: A review, taxonomy, governance framework, and future research agenda," *Journal of Emerging Paradigms in Computing Systems*, vol. 1, no. 01, 2025. Available: https://jepcs.org/index.php/jepcs/article/view/19.

[2] Z. ul Huda, "An enhanced hybrid PSO-LSTM framework for energy-aware task scheduling in heterogeneous cloud computing environments," *Journal of Emerging Paradigms in Computing Systems*, vol. 1, no. 01, 2025. Available: https://jepcs.org/index.php/jepcs/article/view/11.

[3] A. Al-Hourani, S. Kandeepan, and S. Lardner, "Optimal LAP altitude for maximum coverage," *IEEE Wireless Communications Letters*, vol. 3, no. 6, pp. 569-572, 2014. doi: 10.1109/LWC.2014.2342736.

[4] Y. Zeng and R. Zhang, "Energy-efficient UAV communication with trajectory optimization," *IEEE Transactions on Wireless Communications*, vol. 16, no. 6, pp. 3747-3760, 2017. doi: 10.1109/TWC.2017.2688328.

[5] B. Liang and Z. J. Haas, "Predictive distance-based mobility management for PCS networks," in *Proceedings of IEEE INFOCOM*, vol. 3, New York, NY, USA, 1999, pp. 1377-1384. doi: 10.1109/INFCOM.1999.752157.

[6] R. S. Sutton, "Integrated architectures for learning, planning, and reacting based on approximating dynamic programming," in *Proceedings of the Seventh International Conference on Machine Learning*, Austin, TX, USA, 1990, pp. 216-224. doi: 10.1016/B978-1-55860-141-3.50030-4.

[7] H. B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. Agüera y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics (AISTATS)*, vol. 54, Fort Lauderdale, FL, USA, 2017, pp. 1273-1282.

[8] D. Gündüz, Z. Qin, I. E. Aguerri, H. S. Dhillon, Z. Yang, A. Yener, K.-K. Wong, and C.-B. Chae, "Beyond transmitting bits: Context, semantics, and task-oriented communications," *IEEE Journal on Selected Areas in Communications*, vol. 41, no. 1, pp. 5-41, 2023. doi: 10.1109/JSAC.2022.3223408.

[9] Y. Zeng, J. Xu, and R. Zhang, “Energy minimization for wireless communication with rotary-wing UAV,” *IEEE Transactions on Wireless Communications*, vol. 18, no. 4, pp. 2329-2345, 2019. doi: 10.1109/TWC.2019.2902559.

[10] W. Khawaja, I. Guvenc, D. W. Matolak, U.-C. Fiebig, and N. Schneckenberger, “A survey of air-to-ground propagation channel modeling for unmanned aerial vehicles,” *IEEE Communications Surveys & Tutorials*, vol. 21, no. 3, pp. 2361-2391, 2019. doi: 10.1109/COMST.2019.2915069.