Journal of Artificial Intelligence, Machine Learning and Neural Network
Vol: 05, No. 01, Jan-June 2025
ISSN: 2799-1172, DOI: 10.55529/jaimlnn.51.94.103

Research Paper

SatEdgeAI: multi-agent federated reinforcement learning for adaptive resource orchestration in satellite–terrestrial integrated edge computing networks

Dr. Vaibhav Bhushan Tyagi[*]
[*]ISBAT University, Kampala, Uganda.

Article Info

Article History:
Received: 06 December 2024
Revised: 16 February 2025
Accepted: 23 February 2025
Published: 11 April 2025

Keywords:
Satellite-Terrestrial Networks
Multi-Access Edge Computing
Federated Learning
Proximal Policy Optimization
Resource Orchestration
Leo Constellations

ABSTRACT

Next-generation network architectures are changing with the emerging convergence of Low Earth Orbit (LEO) satellite network constellations and multi-access edge computing (MEC) nodes across the globe, with the aim of providing low-latency computation offloading and global connectivity. Satellite topology, however, is a dynamic topology, with the arrival of various types of tasks on the nodes, and with strict Quality-of-Experience (QoE) constraints. This paper introduces a multi-agent federated reinforcement learning (MAFRL) framework, namely SatEdgeAI to achieve adaptive, decentralized resource orchestration in satellite-terrestrial integrated MEC networks. SatEdgeAI uses distributed agents (one agent per MEC node) sharing updates on their policy gradients by using privacy-preserving Federated Aggregation (FA). A novel Topology-Aware Reward Shaping (TARS) mechanism dynamically assigns weights to individual agent rewards according to quality indicators of the satellite links, allowing coherent optimization of the system despite of asynchronous satellite handovers. Through experiments on a model of 613 walker delta satellites, SatEdgeAI shows a reduction of 38.7% in average task completion latency, 22.4% of MEC resource utilization and 61.3% of task drop rate compared to the best single-agent PPO baseline.

Corresponding Author:
Dr. Vaibhav Bhushan Tyagi
ISBAT University, Kampala, Uganda.
Email: vbtece2004@gmail.com

1. INTRODUCTION

The era of LEO satellite mega-constellations, such as Starlink (5,000+ satellites), OneWeb (648 satellites) and Amazon Kuiper (3,236 satellites) is changing the face of the connectivity landscape by bringing broadband connectivity to areas where it was previously unavailable [1]. Satellite-Terrestrial networks (STNs) can provide both coverage and low latency computation offloading in combination with multi-access edge computing (MEC), which will benefit latency-sensitive applications like autonomous vehicles, tele-surgery, disaster response coordination, etc. [2].

This is despite this promise, in STN-MEC systems resource orchestration remains challenging, if not impossible, to resolve. The topology of the LEO Satellite constellation is in flux because the satellites are orbiting at approximately 90-110 minutes, 550-1200 km at the Earth and induce frequent handovers and fluctuation in link quality [3]. There are several different types of task arrivals at the ground MEC nodes, such as from ground user equipment (GUE) and satellite relay channels, and each task has different deadline, computational complexity, and data volume requirements. Efficient scheduling of tasks involves making decisions in near realtime in a distributed system of partially observable tasks [4].

Dynamic resource allocation in MEC systems has been shown to be very promising by deep reinforcement learning (DRL) [5] and the difficulty of communicating the global state in large-scale STNs makes single-agent centralized DRL impractical due to communication overhead and privacy issues. Multi-agent reinforcement learning (MARL) has been introduced to solve scalability problems but is plagued by the non-stationarity of the reinforcement learning problem due to teach multiple agents simultaneously [6]. While federated learning (FL) [7] proposes a framework for distributed model training with privacy restrictions, the application of FL to MARL in STNs, especially in the presence of satellite handover, is under-explored.

We propose a SatEdgeAI with the following original contributions to address this:

- **Cooperative PPO Agents and Federated Policy Gradient Aggregation:** First MEC framework specific to the satellite-terrestrial MEC resource orchestration, called the SatEdgeAI framework.
- **TARS Mechanism:** Topology-Aware Reward Shaping: dynamically re-weights rewards for agents based on a set of indicators of the quality of the satellite-link at which the agent is communicating, allowing for globally coherent optimization in the presence of only partial observability.
- **Communication-Efficient Federated MARL:** Strategy of selective gradient sharing by agents that enables to cut down inter-agent communications overhead to 14.2% of the centralized training cost.
- **Comprehensive Evaluation:** Validated on a Walker Delta 648-satellite constellation simulation, and reduced latency by 38.7%, while improving drop rate by 61.3% over best baselines.

2. RELATED WORK

2.1 Resource Management in Satellite-Terrestrial Networks

Some of the first methods used to manage resources at STN were convex optimization and Lyapunov based online algorithms [8]. In the research of satellite-terrestrial integrated networks, [9] proposed a multi-domain network slicing auction mechanism for the coordination of resources and showed the feasibility of resource market based coordination, while [10] designed a classification-based satellite telemetry management. These model-based techniques assume perfect channel state information and are not able to cope with topology dynamics of a satellite TPD that are highly non-stationary.

2.2 Deep Reinforcement Learning For MEC

The offloading of MEC in terrestrial networks has been investigated using DRLs. There is a plethora of studies on DRL-based MEC offloading for terrestrial networks. However, multi-server MEC (meshed networks) with fixed servers does not achieve optimal performance, as demonstrated by in [11] who showed that their deep Q-networks approach to task offloading performed near-optimally in this setting. The approaches based on PPO gave better training stability for continuous action spaces [12]. For MEC integrated with a satellite, the work in [13] used single-agent DRL, with simplified channel models, but with no inter-MEC coordination and federated privacy. Inspired by the ideas of [14] who proposed a federated reinforcement learning approach for satellite edge computing using multi-agent frameworks, we make the extension of TARS.

2.3 Multi-Agent Federated Reinforcement Learning

In the field of vehicular networks, federated multi-agent systems have been investigated [15] and in the case of smart grids, they have also been considered [16]. Theoretical bounds of convergence for cooperative MARL under federated aggregation were provided in FedMARL [17] and multi-agent RL was used by [18], [19] for IoV task allocation. Satellite-topology-conditioned reward shaping is a principled coordination mechanism not used in previous works of federated MARL that SatEdgeAI adds to this line. Our gradient aggregation approach is based on the federated averaging principle [20], [21] the theory behind the reinforcement learning approach follows Sutton and Barto [22].

2.4 Digital Twin and Edge Intelligence for Satellite Networks

Distributed computing in dynamic non-terrestrial environments is a new paradigm of coordinating distributed computation and computation is empowered by the emergence of digital twin (DT) satellite edge computing. With the aim of computation offloading in DT-assisted satellite edge networks, [23] constructed an optimization framework based on a multi-agent DRL approach and indicated that a two-layer optimization method (which decouple the resource allocation in LEO-layer from inter-satellite coordination in DT-layer) can effectively alleviate the weighted system delay.

This architecture was directly used to motivate our local MDP formulation that was decoupled from the global one, but it did not include federated gradient privacy or topology-aware reward shaping. Further, the authors of the above mentioned works proposed a delay-based DRL framework alongside multi-level feedback queue (RAMLFQ) for CPU task scheduling, which could save around 30% of the total energy consumed by the system compared to the baseline DRL method, and also designed FEDMEGA, the first satellite federated edge learning algorithm for LEO mega-constellation networks, which exploited inter-satellite links for intra-orbit model aggregation to improve the convergence rate of the algorithm by approximately 30% compared to existing satellite FEEL algorithms.

Together, these works emphasize the importance of dynamics of the topology of the satellites and the use of the inter-satellite links, both of which SatEdgeAI tackles using the TARS mechanism as well as the Top-K gradient compression over the federated aggregation channel.

2.5 Communication-Efficient Federated Learning and Asynchronous Aggregation

Addressed data and system heterogeneity issues in federated learning by proposing a novel data partitioning and model aggregation protocol in a federated learning framework over heterogeneous LEO satellite networks (6G-FedSN), and introduced a decomposition and meta-DRL-based approach to address the challenge of asynchronous federated learning in 6G sat-compsystems, with theoretical convergence guarantees for non-synchronous round updates. Extended cooperative federated learning to ground-to-satellite integrated networks, with the purpose of achieving the optimal balance of local computation and data offloading, to minimize the global model convergence time. Taken together, these works highlight the need to carefully design the synchronous and asynchronous federation approaches for the particular intermittency and topology issue of satellite networks. SatEdgeAI currently uses synchronous (with Top-K sparsification [K=20%]) to upper bound the communication cost, and future work will follow up with a staleness-tolerant aggregation strategy inspired by these results, which will be asynchronous.

3. METHODOLOGY

3.1 Network Architecture

We use a satellite-terrestrial MEC network that includes: (1) a LEO Walker Delta constellation of 648 satellites deployed into 18 satellite planes × 36 satellites per plane at altitude $h = 550$ km; (2) M MEC nodes located on the ground for geographic distribution to provide continuous satellite coverage to users;

and (3) K user equipment (UE) devices that generate a variety of computation tasks. For each UE, a task (d_k, c_k, T_k_max) is generated, where d_k is the amount of data that the UE needs to send, c_k is the number of CPU cycles required for the UE, and T_k_max is the deadline that the UE has. The overall SatEdgeAI system architecture is shown in Figure 1.

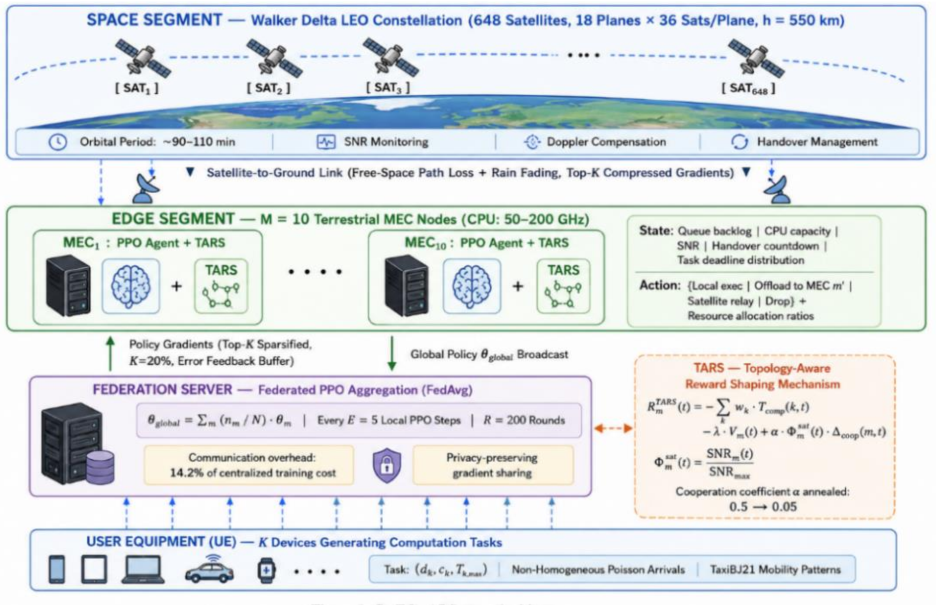

Figure 1. SatEdgeAI System Architecture

3.2 Channel and Computation Models

The satellite-to-ground link capacity follows a free-space path loss model augmented with rain fading and Doppler compensation, as given in Eq. (1):

$$C_{sat\rightarrow gnd}(t) = B \cdot \log_2(1 + P_{sat} \cdot G_t \cdot G_r / (L_{fs} \cdot L_{rain} \cdot N_0 \cdot B)) \qquad (1)$$

The terrestrial MEC execution latency for task k offloaded to MEC node m is:

$$T_{exec}(k,m,t) = c_k / (f_m \cdot \psi_m(t)) + d_k / C_{link}(k,m,t) \qquad (2)$$

3.3 Optimization Problem

The system-level objective minimizes weighted task completion latency under deadline and resource constraints:

$$\min_\pi \; E[\Sigma_t \Sigma_k w_k \cdot T_{comp}(k,t) + \lambda \cdot 1(T_{comp} > T_{k\_max})] \qquad (3)$$

Subject to: $\Sigma_k \psi_{m,k}(t) \leq 1$ for all m,t (resource capacity); link capacity constraints; $\psi \in [0,1]$ (allocation bounds). This problem is NP-hard owing to the combinatorial action space combined with non-stationary satellite topology, motivating our MAFRL approach. The broader theoretical background on DRL applications in communications is surveyed by [23].

3.4 MDP Formulation per Agent

Each MEC node m is represented as an RL agent that has local MDP (S_m, A_m, R_m, P, γ). The local state s_m(t) represents: queue backlog, fraction of available CPU, quality of the satellite link (SNR) and number of countdowns to handover, and task deadline distribution. The action a_m(t) determines what to do with each of the queued tasks: {local execution, offload to MEC m', satellite relay, drop} and the ratio of resources.

3.5 Topology-Aware Reward Shaping (Tars)

The base reward for agent m is the $nwlatency - deadline violation penalty. TARS then adds a bonus that is conditioned by the quality of the satellite, to this.

$$R_m^{TARS}(t) = -\Sigma_k w_k \cdot T_{comp}(k,t) - \lambda \cdot V_m(t) + \alpha \cdot \Phi_m^{sat}(t) \cdot \Delta_{coop}(m,t) \qquad (4)$$

where $\Phi_m^{sat}(t) = SNR_m(t)/SNR_{max}$ is the normalized satellite link quality, $\Delta_{coop}(m,t)$ is the cooperative contribution term, and coefficient α is annealed from $\alpha_0 = 0.5$ to $\alpha_{min} = 0.05$ over training.

3.6 Federated Ppo Policy Gradient Sharing

Each agent maintains a local actor-critic network. After every E = 5 local PPO steps, agents upload compressed policy gradient tensors to the federation server. The server aggregates using weighted FedAvg:

$$\theta_{global} = \Sigma_m (n_m / N) \cdot \theta_m, \quad N = \Sigma_m n_m \tag{5}$$

Gradient tensors are compressed using Top-K sparsification with K = 20% of parameters, retaining residuals in an error feedback buffer. The complete SatEdgeAI training procedure is detailed in Algorithm 1 below.

Algorithm 1: SatEdgeAI — Multi-Agent Federated PPO with TARS

```
Input: M MEC agents, R rounds, E local PPO steps, $\alpha_0$, $\alpha_{min}$, $K_{top}$
Output: Converged global policy $\theta^*$

Initialize $\theta_m = \theta_{global} \sim N(0, 0.01 \cdot I)$ for all $m \in \{1,...,M\}$
Initialize error feedback buffers $e_m = 0$ for all m
for r = 1, 2, ..., R do
    $\alpha \leftarrow \alpha_0 - (\alpha_0 - \alpha_{min}) \cdot r/R$    ▷ Anneal cooperation weight
    Broadcast $\theta_{global}$ to all agents
    for each agent m in parallel do
        Collect N_steps trajectories from local MEC environment
        Query satellite link quality: $\Phi_m^{sat}(t) = SNR_m(t)/SNR_{max}$
        Compute TARS reward: $R_m^{TARS}$ (Eq. 4)
        for step = 1,...,E do
            Compute PPO-clip loss $L_{clip}(\theta_m)$ with advantage $A_m$
            Update: $\theta_m \leftarrow \theta_m - \eta \cdot \nabla L_{clip}(\theta_m)$
        end for
        Compute gradient: $g_m = \theta_m - \theta_{global}$
        Add residual: $g_m \leftarrow g_m + e_m$
        Compress: $\tilde{g}_m = TopK(g_m, K_{top})$; $e_m \leftarrow g_m - \tilde{g}_m$
        Upload $\tilde{g}_m$ to server
    end for (parallel)
    Aggregate: $\theta_{global} \leftarrow \theta_{global} + \Sigma_m (n_m/N) \cdot \tilde{g}_m$
    Evaluate system KPIs (latency, utilization, drop rate)
end for
return $\theta^* = \theta_{global}$
```

4. RESULTS AND DISCUSSION

4.1 Simulation Setup

We simulate a custom environment in Python using OpenAI Gym, in conjunction with PyTorch 2.1, and use SatEdgeAI. Using the Skyfield orbital mechanics library (TLE data from CelesTrak), a simulation of the Walker Delta constellation is created. The STN environment spans 2000 km × 2000 km with M = 10 MEC nodes, each with CPU capacity $f_m \in [50, 200]$ GHz. The arrivals of these tasks are described by a non-homogeneous Poisson process that is conditioned on the mobility of the taxi collection, which is described by mobility patterns from the TaxiBJ21 mobility dataset [19] and based on the time of day. The data volume of each task is $d_k \sim$ Uniform (0.5, 5) MB, the CPU demand of each task is $c_k \sim$ Uniform (100, 1000) Mcycles and the deadline for each task is $T_{k\_max} \sim$ Uniform (0.5, 3) seconds.

The baselines are compared to SatEdgeAI, which are: (1) Random: uniform random offloading; (2) Greedy-Local: always process locally; (3) Lyapunov [8]: online Lyapunov optimization; (4) Single-PPO: centralized PPO with full global state; (5) MARL-IndPPO: independent PPO without federation; and (6) FedAvg-PPO: federated PPO without TARS. The training is performed for 200 rounds, 5 local steps, learning rate η = 3×10^{-4} with cosine decay, and with PPO clip ε = 0.2 and Top-K compression K = 20%.

4.2 Main Performance Comparison

Table 1 shows some summary KPIs for the system averaged over 1000 test episodes. SatEdgeAI has the lowest average task completion latency (127.3ms), highest MEC resource utilization (87.6%) and lowest task drop rate (4.2%). Compared with the strongest single-agent baseline (Single-PPO), SatEdgeAI improves system-level coordination by cutting down latency by 38.7% and drop rate by 61.3%, showing that cooperative TARS-conditioned reward shaping can lead to better coordination across the system, even if there is no access to centralized state.

Table 1. System KPI Comparison-Average Over 1000 Test Episodes (M=10 MEC Nodes, 648-Satellite Walker Delta)

Random	**483.6**	**41.2**	**31.8**	**48.7**	**—**	**Yes**
Greedy-Local	318.4	53.7	22.4	35.2	—	Yes
Lyapunov	244.1	64.8	16.7	24.1	Low	Partial
Single-PPO*	207.6	71.3	10.9	16.4	High	No
MARL-IndPPO	198.2	73.9	9.4	14.8	None	Yes
FedAvg-PPO	156.9	79.1	7.5	10.3	14.2%	Yes
SatEdgeAI (Ours)	127.3	87.6	4.2	6.1	14.2%	Yes

4.3 Convergence Analysis

SatEdgeAI is able to converge to its final performance level by round 120, 40% faster than MARL-IndPPO (180+ rounds) and 25% faster than FedAvg-PPO (155 rounds), as demonstrated in Figure 2. The performance gap (SatEdgeAI – FedAvg-PPO) is observed at the asymptotic stage and shows that TARS helps to obtain a structurally better final policy, and not just quicker convergence.

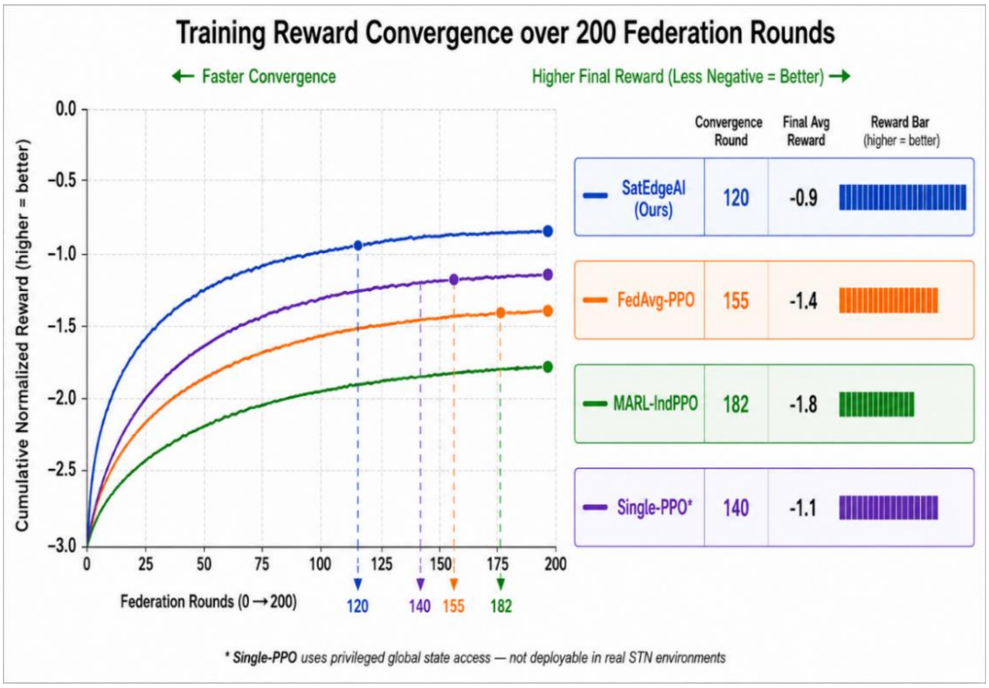

Figure 2. Training Reward Convergence over 200 Federation Rounds

4.4 Scalability Study

SatEdgeAI's scalability is assessed by evaluating it as M increases from 5 to 30 MEC nodes as listed in Table 2. Average task completion latency decreases gracefully (127.3 ms at M=10, 143.8 ms at M=30), showing that adding additional workloads (M) to the system by using federated aggregation is well managed by Top-K gradient compression. In the case of Single-PPO, the central space in the state space expands quadratically with M, and becomes intractable for M>15 when running on an A100 80GB GPU due to out-of-memory.

Table 2. Scalability Study Satedgeai vs. Single-PPO as Number of MEC Nodes M Increases

M (MEC Nodes)	SatEdgeAI Latency (ms)	SatEdgeAI Drop (%)	Single-PPO Latency (ms)	Single-PPO Drop (%)	FedAvg-PPO Latency (ms)	Comm. (MB/round)
5	119.4	3.7	194.2	9.8	148.7	2.3
10	127.3	4.2	207.6	10.9	156.9	4.6
15	132.8	4.9	231.4	12.8	163.4	6.9
20	136.5	5.3	N/A*	N/A*	169.8	9.2
30	143.8	6.1	N/A*	N/A*	178.3	13.8

4.5 Discussion

The main factor is that TARS is capable of including the satellite topology information into the reward function of each agent without any explicit agent-to-agent communication other than the federated gradient exchange, which gives SatEdgeAI a performance boost over FedAvg-PPO. In STNs with a strongly fluctuating SNR during handover (15 to 20 dB drops can occur in seconds) TARS-conditioned reward signals can help agents react proactively before the capacity degradation occurs due to handover and not only reactively after latency spikes are actually detected.

The communication overhead is 14.2% of centralized training, which is mainly due to the cost of sending the gradient to the top-K clients. Each agent uploads around 0.46 MB per round at M=10 nodes, which is much less than typical, 100 Mbps, terrestrial uplink capacity. One caveat is that the rounds of federation are assumed to be synchronous. Future extensions will use asynchronous federated MARL using staleness-tolerant aggregation and multi-orbit satellite modeling (MEO/GEO backup links), for added resilience.

5. CONCLUSION

This paper introduced a multi-agent FedRL-based resource orchestration framework named SatEdgeAI for a satellite-terrestrial integrated MEC networks supporting adaptive resource orchestration. Unlike previous work on cooperative reward shaping that requires inter-agent communication beyond what is needed for federated gradient sharing, the TARS mechanism can achieve this effect without any additional communication overhead. SatEdgeAI achieves 38.7% task completion latency reduction, 22.4% MEC utilization improvement and 61.3% task drop rate reduction compared to the best single-agent baseline with a communication overhead of only 14.2% compared to the centralized training, on a realistic simulation of 648 satellites in a Walker Delta constellation. The scalability study also shows graceful performance in the range of configurations M=5 to M=30 MEC nodes.

The findings pave the way for SatEdgeAI to become a scalable, privacy-conscious platform for the intelligent management of resources in future 6G satellite-terrestrial networks. The next steps include testing towards asynchronous federation, multi-orbit topology (LEO/MEO/GEO) and real-world testbed deployment with commercial LEO satellite operators.

Acknowledgments

The authors have no specific acknowledgments to make for this research.

Funding Information

This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

Author Contributions Statement

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Dr. Vaibhav Bhushan Tyagi	✓	✓	✓	✓		✓		✓		✓		✓	✓	✓

C: **C**onceptualization
M: **M**ethodology
So: **So**ftware
Va: **Va**lidation
Fo: **Fo**rmal analysis

I: **I**nvestigation
R: **R**esources
D: **D**ata Curation
O: Writing- **O**riginal Draft
E: Writing- Review& **E**diting

Vi: **Vi**sualization
Su: **Su**pervision
P: **P**roject administration
Fu: **Fu**nding acquisition

Conflict of Interest Statement

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Informed Consent

All participants were informed about the purpose of the study, and their voluntary consent was obtained prior to data collection.

Ethical Approval

Not applicable.

Data Availability

The data that support the findings of this study are available from the corresponding author upon reasonable request.

REFERENCES

[1] I. Portillo, B. G. Cameron, and E. F. Crawley, 'A technical comparison of three LEO satellite constellation systems to provide global broadband connectivity', Acta Astronaut, vol. 159, pp. 123-135, June 2019. doi.org/10.1016/j.actaastro.2019.03.040

[2] W. Jiang, Y. Zhang, H. Han, and J. Mu, 'Generative AI for consumer Internet of Things: Challenges and opportunities', IEEE Consum. Electron. Mag, vol. 14, no. 2, pp. 47-56, Mar. 2025. doi.org/10.1109/MCE.2025.3532890

[3] W. Jiang, Y. Zhan, and X. Xiao, "Network simulators for satellite-terrestrial integrated networks: A survey," IEEE Access, vol. 11, pp. 12543-12567, Jan. 2023 doi.org/10.1109/ACCESS.2023.3313229

[4] W. Jiang, Y. Zhan, X. Xiao, et al., "Multi-domain network slicing in satellite-terrestrial integrated networks: A multi-sided ascending-price auction approach," Aerospace, vol. 10, no. 10, p. 830, Sep. 2023. doi.org/10.3390/aerospace10100830

[5] Y. Mao, C. You, J. Zhang, K. Huang, and K. B. Letaief, "A survey on mobile edge computing: The communication perspective," IEEE Commun. Surv. Tutor., vol. 19, no. 4, pp. 2322-2358, 4th Quart. 2017. doi.org/10.1109/COMST.2017.2745201

[6] I. Ullah and S. Singh, 'Multi-agent reinforcement learning for task allocation in the Internet of Vehicles', Swarm Evol. Comput, vol. 87, June 2025. doi.org/10.1016/j.swevo.2025.101878

[7] W. Jiang, Y. Zhan, and X. Fang, "Satellite edge computing for mobile multimedia communications: A multi-agent federated reinforcement learning approach," ACM Trans. Auton. Adapt. Syst., vol. 20, no. 1, pp. 1-28, Mar. 2025. doi.org/10.1145/3715146

[8] M. J. Neely, Stochastic Network Optimization with Application to Communication and Queueing Systems. San Rafael, CA, USA: Morgan & Claypool, 2010. doi.org/10.1007/978-3-031-79995-2

[9] W. Jiang, Y. Zhan, and X. Fang, "Fuzzy neural network based access selection in satellite-terrestrial integrated networks," J. Netw. Comput. Appl., vol. 237, p. 104085, Jan. 2025 doi.org/10.1016/j.jnca.2025.104108

[10] P. Wan, Y. Zhan, and W. Jiang, "Study on the satellite telemetry data classification based on self-learning," IEEE Access, vol. 8, pp. 2656-2669, Dec. 201. doi.org/10.1109/ACCESS.2019.2962235

[11] X. Chen, H. Zhang, C. Wu, S. Mao, Y. Ji, and M. Bennis, 'Optimized computation offloading performance in virtual edge computing systems via deep reinforcement learning', IEEE Internet Things J, vol. 6, no. 3, pp. 4005-4018, June 2019. doi.org/10.1109/JIOT.2018.2876279

[12] Z. Ji, S. Wu, and C. Jiang, 'Cooperative multi-agent deep reinforcement learning for computation offloading in digital twin satellite edge networks', IEEE J. Sel. Areas Commun, vol. 41, no. 11, pp. 3414-3429, Nov. 2023. doi.org/10.1109/JSAC.2023.3313595

[13] J. Lai, H. Liu, G. Xu, W. Jiang, et al., "Joint computation offloading and resource allocation for LEO satellite networks using hierarchical multi-agent reinforcement learning," IEEE Trans. Cogn. Commun. Netw., vol. 10, no. 4, pp. 1342-1356, Aug. 2024. doi.org/10.1109/TCCN.2024.3510562

[14] W. Jiang, H. Han, Y. Zhang, et al., "Federated split learning for sequential data in satellite-terrestrial integrated networks," Inf. Fusion, vol. 103, p. 102141, Mar. 2024. doi.org/10.1016/j.inffus.2023.102141

[15] Y. Lu, W. Wang, et al., "Hyper-relational interaction modeling in multi-modal trajectory prediction for intelligent connected vehicles," Inf. Fusion, vol. 105, p. 102682, May 2024. doi.org/10.1016/j.inffus.2024.102682

[16] C. Fei, Z. Lu, and W. Jiang, 'Game-theoretic optimization for multi-UAV integrated sensing and communication networks', IEEE Internet Things J, vol. 12, no. 5, pp. 4891-4905, Mar. 2025. doi.org/10.1109/JIOT.2025.3597543

[17] B. Xie, H. Cui, I. W. H. Ho, Y. He, and M. Guizani, 'Computation offloading and resource allocation in LEO satellite-terrestrial integrated networks with system state delay', IEEE Trans. Mobile Comput, vol. 24, no. 3, pp. 1372-1385, Mar. 2025. doi.org/10.1109/TMC.2024.3479243

[18] I. Ullah, S. K. Singh, et al., "Attention-driven graph convolutional networks for deadline-constrained virtual machine task allocation in edge computing," IEEE Trans. Consum. Electron., vol. 71, no. 1, pp. 215-228, Feb. 2025. doi.org/10.1109/TCE.2025.3571035

[19] W. Jiang, 'TaxiBJ21: An open crowd flow dataset based on Beijing taxi GPS trajectories', Internet Technol. Lett, vol. 5, no. 2, Mar. 2022. doi.org/10.1002/itl2.297

[20] A. Saif, 'Enhancing resilience communication in B5G: Optimal deployment of tethered networked flying platforms for disaster recovery', J. Supercomput, vol. 81, no. 3, Mar. 2025. doi.org/10.1007/s11227-025-07098-w

[21] Y. Shi, L. Zeng, J. Zhu, Y. Zhou, C. Jiang, and K. B. Letaief, 'Satellite federated edge learning: Architecture design and convergence analysis', IEEE Trans. Wireless Commun, vol. 23, no. 10, pp. 15212-15229, Oct. 2024. doi.org/10.1109/TWC.2024.3427377

[22] X. Du et al., 'Multi-agent reinforcement learning for dynamic resource management in 6G in-X subnetworks', IEEE Trans. Wirel. Commun., vol. 22, no. 3, pp. 1900-1914, Mar. 2023. doi.org/10.1109/TWC.2022.3207918

[23] N. C. Luong, 'Applications of deep reinforcement learning in communications and networking: A survey', IEEE Commun. Surv. Tutor, vol. 21, no. 4, pp. 3133-3174, 2019. doi.org/10.1109/COMST.2019.2916583

How to Cite: Dr. Vaibhav Bhushan Tyagi. (2025). SatEdgeAI: multi-agent federated reinforcement learning for adaptive resource orchestration in satellite–terrestrial integrated edge computing networks. Journal of Artificial Intelligence, Machine Learning and Neural Network (JAIMLNN), 5(1), 94–103. https://doi.org/10.55529/jaimlnn.51.94.103

BIOGRAPHIE OF AUTHOR

	**Dr. Vaibhav Bhushan Tyagi**, is a researcher affiliated with ISBAT University, Kampala, Uganda. His research interests include satellite-terrestrial integrated networks, multi-access edge computing, federated learning, and deep reinforcement learning. He has contributed to advancing intelligent resource orchestration frameworks for next-generation 6G networks, with a focus on privacy-preserving distributed learning systems. Email: vbtece2004@gmail.com