MATOFF-A3C: Multi-Agent Task Offloading with Shared Global Model Training for UAV Edge Computing

Taewon Song and Taeyoon Kim

***Abstract*—Multi-UAV task offloading optimization in edge computing environments represents a critical challenge for enabling efficient distributed computation, as UAV swarms must make coordinated decisions under dynamic network conditions, limited computational resources, and stringent latency constraints. This paper presents MATOFF-A3C, a multi-agent framework that employs shared global model training through Asynchronous Advantage Actor-Critic algorithms to optimize task offloading decisions across distributed UAV networks. The methodology formulates the problem as a multi-agent Markov Decision Process where several UAV workers operating in heterogeneous environments contribute learning experiences to a centralized global model, enabling coordinated decision-making for local processing, MEC server offloading, or task discard actions. Experimental evaluation demonstrates that the shared global model achieves superior reward distribution and convergence characteristics compared to individual worker training, with statistical analysis revealing significant performance improvements across diverse operational scenarios. Cross-environment generalization experiments show enhanced knowledge transfer capabilities, where the global model maintains robust performance when deployed in previously unseen environmental configurations. These findings provide essential insights for designing scalable multi-UAV systems and demonstrate the effectiveness of centralized learning approaches for distributed edge computing applications.**

***Index Terms*—UAV, Task Offloading, A3C, Deep Reinforcement Learning, Multi-Agent Systems, Edge Computing**

I. Introduction

The proliferation of Unmanned Aerial Vehicles (UAVs) in various applications has created new opportunities for distributed computing and task offloading in edge computing environments. As UAV swarms become increasingly sophisticated, the challenge of optimizing computational task distribution among multiple agents becomes critical for system performance and energy efficiency.

Modern UAV systems are required to process computationally intensive tasks such as real-time image processing, path planning, and sensor data analysis while operating under strict energy and latency constraints. The limited computational resources onboard individual UAVs necessitate intelligent task offloading strategies that can dynamically distribute workloads across the network, including mobile edge computing (MEC) servers and cloud infrastructure.

Reinforcement Learning (RL), particularly deep reinforcement learning approaches, has emerged as a promising solution for addressing the complex decision-making challenges in multi-UAV systems. Among various RL algorithms, Asynchronous Advantage Actor-Critic (A3C) has shown significant potential due to its ability to handle continuous action spaces and its inherent parallelization capabilities. The asynchronous nature of A3C allows multiple agents to learn simultaneously, making it particularly suitable for distributed UAV scenarios where coordination and communication constraints exist.

However, a fundamental question remains regarding the optimal training strategy for A3C in multi-UAV environments: should the system employ a centralized approach with a shared global model, or would decentralized training with individual worker specialization yield superior performance? This question becomes even more critical when considering the heterogeneous nature of operational environments that UAV systems encounter.

This paper investigates two fundamental training strategies for A3C-based multi-UAV task offloading: (1) centralized training using a global model shared across all workers, and (2) decentralized training where individual workers develop specialized policies. Through comprehensive experimental evaluation across multiple environmental configurations, we analyze the performance characteristics, convergence properties, and generalization capabilities of each approach.

To sum up, the contributions of this paper are fourfold:

- We present a comprehensive performance comparison between A3C global and individual training strategies for multi-UAV task offloading optimization, providing the first systematic evaluation of these training paradigms in UAV-assisted edge computing environments.
- We conduct rigorous statistical analysis of reward distributions and convergence characteristics across multiple heterogeneous environments, demonstrating significant performance differences between training approaches.
- We develop an evaluation framework for cross-environment generalization in multi-UAV systems, enabling assessment of training strategy effectiveness across diverse operational scenarios.
- We provide practical insights into optimal training paradigms for different operational scenarios, offering guidelines for UAV system deployment in varying environmental conditions.

T. Song is with the Department of Internet of Things, College of SW Convergence, Soonchunhyang University, 22 Soonchunhyang-ro, Shinchang-myeon, Asan-si, Chungcheongnam-do, 31538, Korea (e-mail: twsong@sch.ac.kr) and T. Kim is with ... (e-mail: 2000kty@dankook.ac.kr), Corresponding author: T. Kim.

This work was supported in part by ..., and in part by ..., and in part by ...

The remainder of this paper is organized as follows. Section II offers a comprehensive review of relevant studies in UAV task offloading and multi-agent reinforcement learning. We describe the system model and problem formulation in Section III. The experimental setup and implementation details are presented in Section IV. Performance evaluation results are given in Section V, followed by discussion in Section VI and conclusion in Section VII.

II. Related Works

The intersection of UAV-assisted mobile edge computing and deep reinforcement learning has emerged as a critical research area, with various approaches addressing the challenges of distributed task offloading optimization. This section reviews existing work across three key themes: deep reinforcement learning approaches for UAV task offloading, multi-agent coordination strategies, and federated learning frameworks.

A. Deep Reinforcement Learning for UAV Task Offloading

Recent advances in deep reinforcement learning have shown significant promise for addressing the complex decision-making challenges in UAV-assisted mobile edge computing environments. Traditional optimization approaches often struggle with the dynamic and uncertain nature of wireless channels, computational demands, and energy constraints.

Song [1] addresses the challenge of hidden channel conditions in UAV-assisted MEC systems by proposing PORTO-MEC, a Deep Recurrent Q-Network (DRQN) based algorithm. The work formulates the problem as a Partially Observable Markov Decision Process (POMDP), where UAVs must make task offloading decisions without complete information about channel states between UAVs and cloud servers. The DRQN approach leverages temporal dependencies in the environment to make informed decisions under uncertainty, achieving 58.82% higher rewards compared to traditional DQN approaches, 92.39% improvement over local-only processing, and 213.51% improvement over offload-only strategies. However, this work focuses on single-agent scenarios and does not address multi-UAV coordination challenges.

Wang et al. [2] and Liu et al. [3] have explored various deep reinforcement learning approaches for UAV-enabled wireless communications and collaborative task offloading. These works demonstrate the effectiveness of DRL methods in handling dynamic environments but primarily focus on individual UAV optimization or simplified multi-UAV scenarios without systematic comparison of training strategies.

B. Multi-Agent Deep Reinforcement Learning Approaches

The complexity of multi-UAV systems necessitates sophisticated coordination mechanisms that can handle the distributed nature of the problem while ensuring efficient resource utilization and performance optimization.

Zhao et al. [4] propose a comprehensive multi-agent deep reinforcement learning framework for task offloading in UAV-assisted mobile edge computing systems. Their approach, named MATD3 (Multi-Agent Twin Delayed Deep Deterministic Policy Gradient), addresses the joint optimization of UAV trajectories, computation task allocation, and communication resource management. The framework formulates the problem as a multi-agent Markov Decision Process and employs a cooperative MADRL approach to handle high-dimensional continuous action spaces. The system demonstrates superior performance compared to traditional optimization approaches, particularly in terms of adaptability to UEs' mobility, robustness to resource changes, and flexibility to dynamic computation demands.

The MATD3 framework adopts a centralized training with decentralized execution strategy, where evaluation critic networks obtain global views during training while individual UAVs execute policies based on local observations during deployment. This approach achieves significant improvements in total system cost reduction while maintaining scalability across different numbers of UAVs and user equipment. However, the work does not systematically compare different training paradigms or investigate the trade-offs between centralized and fully decentralized learning approaches.

Foerster et al. [5] and Gupta et al. [6] have established foundational work in cooperative multi-agent reinforcement learning, providing theoretical frameworks that have influenced subsequent UAV coordination research. However, these approaches have not been specifically evaluated for UAV task offloading scenarios or compared across different training strategies.

C. Federated Learning and Decentralized Training

The challenge of data privacy, communication overhead, and training efficiency in distributed systems has led to the exploration of federated learning approaches in edge computing environments.

Wang et al. [7] introduce FADE (Federated Deep reinforcement learning-based cooperative edge caching), a framework that enables base stations to cooperatively learn shared predictive models while keeping training data localized on individual IoT devices. The approach uses Double Deep Q-Network (DDQN) as the underlying reinforcement learning algorithm and demonstrates 92% loss reduction in the first 100 training steps compared to centralized approaches, along with 60% improvement in system payment efficiency. The framework addresses privacy concerns and communication overhead by sharing only model parameters rather than raw data.

While FADE focuses on edge caching rather than computation task offloading, it provides valuable insights into the benefits and challenges of federated learning in distributed edge computing scenarios. The work demonstrates that decentralized training can achieve comparable or superior performance to centralized approaches while addressing practical deployment constraints.

D. Research Gaps and Motivation

Despite the significant progress in UAV-assisted mobile edge computing and deep reinforcement learning, several critical research gaps remain:

Training Strategy Comparison: Existing work has not systematically compared global versus individual training strategies for A3C in multi-UAV environments. While Zhao et al. [4] demonstrate the effectiveness of centralized training with decentralized execution, and Wang et al. [7] show benefits of federated learning for edge caching, there lacks a comprehensive analysis of pure centralized versus fully decentralized training paradigms specifically for task offloading optimization.

A3C Algorithm Exploration: Limited research has explored the potential of Asynchronous Advantage Actor-Critic (A3C) algorithms for multi-UAV task offloading, despite A3C's inherent advantages in parallel learning and policy gradient methods. Most existing work focuses on DQN variants [1] or actor-critic methods like TD3 [4], leaving A3C's performance characteristics unexplored in this domain.

Cross-Environment Generalization: Current approaches primarily evaluate performance within specific environmental configurations without systematic analysis of generalization capabilities across diverse operational scenarios. This limits the practical applicability of trained models in real-world deployments where UAVs encounter varying environmental conditions.

Statistical Rigor: Many existing studies lack comprehensive statistical analysis of performance differences between training approaches, making it difficult to draw definitive conclusions about optimal strategies for different scenarios.

This work addresses these gaps by providing a systematic comparison of A3C global versus individual training strategies, comprehensive statistical analysis across multiple environmental configurations, and evaluation of cross-environment generalization capabilities in multi-UAV task offloading scenarios.

III. Methodology

A. Problem Formulation

The multi-UAV task offloading problem is formulated as a sequential decision-making process where each UAV agent must choose from three possible actions for incoming computational tasks: (1) local processing, (2) offloading to mobile edge computing (MEC) servers, or (3) task discard. The objective is to maximize the cumulative reward while minimizing energy consumption and processing delays.

The state space includes:

- Available computation units (normalized)
- Remaining epochs for task completion
- MEC server computation units and processing times
- Queue status (computation units and processing times)
- Success indicators for local and offload operations
- Context features (agent velocity and computation capacity)

The action space consists of discrete actions: $A = \{0, 1, 2\}$ representing LOCAL, OFFLOAD, and DISCARD respectively.

The reward function incorporates multiple factors including energy costs, processing delays, and failure penalties:

$$R = -\alpha \cdot E_{local} - \beta \cdot T_{offload} - \gamma \cdot D_{transmission} + \text{rewards} - \text{penalties} \tag{1}$$

where α, β, and γ are weighting coefficients for local processing energy cost, offload time cost, and transmission delay respectively.

B. A3C Architecture

The A3C implementation employs both feedforward and recurrent neural network configurations to handle the sequential nature of the task offloading decisions.

1) Network Architecture: The actor-critic architecture consists of:

- **Actor Network**: Outputs action probabilities using a softmax policy
- **Critic Network**: Estimates state values for advantage calculation
- **Hidden Layers**: 128 neurons with ReLU activation functions
- **Recurrent Component**: GRU layers for sequence modeling (when enabled)

For the recurrent variant (RecurrentActorCritic), the network processes sequences of observations through GRU layers before the final actor and critic heads, enabling the model to maintain memory of past decisions and environmental states.

2) Training Strategies: We evaluate two distinct training paradigms:

A3C Global Training: In this centralized approach, all workers share updates to a single global model. Each worker interacts with its assigned environment and computes gradients based on the advantage actor-critic loss:

$$L = L_{policy} + \beta \cdot L_{value} - \alpha \cdot H(\pi) \tag{2}$$

where L_{policy} is the policy loss, L_{value} is the value function loss, and $H(\pi)$ is the policy entropy term for exploration.

The global model aggregates updates from all workers asynchronously, promoting knowledge sharing across different environmental conditions and worker experiences.

Individual Worker Training: In this decentralized approach, each worker develops its own specialized policy independently. Workers train separate neural networks without sharing parameters, allowing for environment-specific adaptation and specialized behavior development.

C. Environment Configuration

The experimental framework utilizes a custom UAV environment with the following key parameters:

- Maximum computation units: 200
- Maximum computation units for cloud: 1000
- Maximum epoch size: 100
- Maximum queue size: 20
- Agent velocities: 50 units

Five distinct environmental configurations are evaluated, each representing different operational scenarios with varying computational demands, network conditions, and resource availability patterns.

D. Performance Metrics

Our evaluation framework includes the following key metrics:

- Episode-level total rewards
- Convergence characteristics across training episodes
- Statistical significance of performance differences
- Cross-environment generalization performance

IV. Experimental Setup

A. Implementation Details

B. Environment Configurations

Five distinct environmental configurations were evaluated, each representing different operational scenarios with varying computational demands and network conditions.

C. Training Parameters

V. Results and Analysis

A. Performance Comparison

B. Episode-level Analysis

C. Distribution Analysis

D. Cross-Environment Performance

E. Statistical Significance

VI. Discussion

VII. Conclusion

References

[1] T. Song, "Drqn-based task offloading in uav-assisted mobile edge computing environments with hidden channel conditions," in *2024 International Conference on Information and Communication Technology Convergence (ICTC)*. IEEE, 2024, pp. 881–884.

[2] L. Wang, K. Wang, C. Pan, W. Xu, N. Aslam, and A. Nallanathan, "Deep reinforcement learning for uav-enabled wireless communications with edge computing," *IEEE Transactions on Cognitive Communications and Networking*, vol. 7, no. 1, pp. 131–141, 2020.

[3] J. Liu, Q. Zhang, Z. Chen, and Q. Ni, "Multi-uav collaborative task offloading and resource allocation in edge computing," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 10, pp. 12 394–12 405, 2020.

[4] N. Zhao, Z. Ye, Y. Pei, Y.-C. Liang, and D. Niyato, "Multi-agent deep reinforcement learning for task offloading in uav-assisted mobile edge computing," *IEEE Transactions on Wireless Communications*, vol. 21, no. 9, pp. 6949–6962, 2022.

[5] J. Foerster, G. Farquhar, T. Afouras, N. Nardelli, and S. Whiteson, "Counterfactual multi-agent policy gradients," *Proceedings of the AAAI conference on artificial intelligence*, vol. 32, no. 1, 2018.

[6] J. K. Gupta, M. Egorov, and M. Kochenderfer, "Cooperative multi-agent control using deep reinforcement learning," *International conference on autonomous agents and multiagent systems*, pp. 66–83, 2017.

[7] X. Wang, C. Wang, X. Li, V. C. M. Leung, and T. Taleb, "Federated deep reinforcement learning for internet of things with decentralized cooperative edge caching," *IEEE Internet of Things Journal*, vol. 7, no. 10, pp. 9441–9455, 2020.