Comparative Analysis of A3C Global vs Individual Training Strategies for Multi-UAV Task Offloading Optimization

Author NameDepartment of Computer Science
University Name
City, State, Country
Email: author@university.edu

Abstract—This paper presents a comprehensive performance analysis of Asynchronous Advantage Actor-Critic (A3C) training strategies for multi-UAV task offloading optimization in edge computing environments. We compare two distinct training paradigms: A3C global model training versus individual worker training across multiple heterogeneous environments. Our experimental framework evaluates performance metrics including reward distribution, convergence characteristics, and cross-environment generalization capabilities. Through statistical analysis of episode-level performance data, we demonstrate the effectiveness of different training approaches under varying environmental conditions. The results provide insights into optimal training strategies for distributed UAV systems and highlight the trade-offs between centralized and decentralized learning approaches in multi-agent reinforcement learning scenarios.

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.

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.

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.

The main contributions of this work include:

- Comprehensive performance comparison between A3C global and individual training strategies
- Statistical analysis of reward distributions and convergence characteristics across multiple environments
- Evaluation framework for cross-environment generalization in multi-UAV systems
- Insights into optimal training paradigms for different operational scenarios

II. Related Work

A. UAV Task Offloading

B. Multi-Agent Reinforcement Learning

C. A3C in Distributed Systems

III. Methodology

A. Problem Formulation

B. A3C Architecture

The A3C architecture employed in this study consists of both feedforward and recurrent neural network configurations. The actor-critic framework utilizes separate networks for policy (actor) and value function (critic) estimation.

1) Network Architecture:

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

A3C Global Training: A centralized approach where all workers share updates to a single global model. This strategy promotes knowledge sharing across different environmental conditions and worker experiences.

Individual Worker Training: A decentralized approach where each worker develops its own specialized policy independently. This allows for environment-specific adaptation but may limit knowledge transfer.

C. Environment Configuration

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

Acknowledgment

The authors would like to thank...

References