DRL for Optimal UAV Routing in IoT Environments: A Comprehensive Exploration and Implementation

M. Tech 3rd Semester C3 Presentation

Dipankar Karmakar (MML2022003)
Under the Supervision of
Dr. Nabajyoti Mazumdar

Indian Institute Of Information Technology, Allahabad
Machine Learning and Intelligent Systems
Department of Information Technology

Presentation Outline

Introduction

Literature Review

The Advantages of DRL over Traditional Heuristics

Demonstrating the Power of DRL: A Survey of Related Works

System Model & Proposed Methodology

Simulation Results

Timeline : Preview of work in Upcoming Semesters

Conclusion

References

Introduction

Motivation of The Study

Age of Connected Devices

- In recent years, the widespread use of 5G communication and **Internet of Things (IoT)** has led to the emergence of various new services and applications, including but not limited to Augmented/Virtual Reality (AR/VR), autonomous driving, and video tracking. These services typically require high computational power and low latency.[1]
- The number of IoT devices such as wearables and sensors is predicted to exceed **14.7 billion** on the internet by 2023, as per **Cisco's report**. It is necessary to process more than 50% of network traffic at the network edges, which have a data processing delay requirement of less than 10ms. [1]

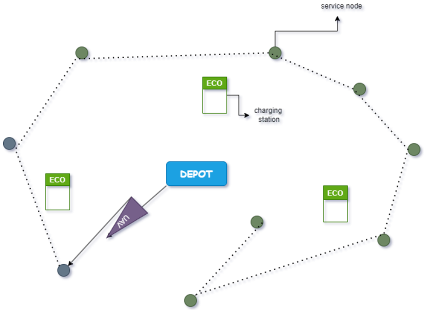
Need of the Study

The Emergence of Mobile Edge Computing:

- **Mobile Edge Computing (MEC)** is a promising approach to tackle these problems by moving computing and storage resources to the edges of the network, including Base Stations (BSs), Access Points (APs), or Unmanned Aerial Vehicles (UAVs). [2]
- **MEC (Mobile Edge Computing)** can offer several advantages such as decreasing delays in transmission and processing, conserving energy on mobile devices, and ensuring privacy and security by transferring computational tasks to the edge servers. [2]

Identifying the Challenge: A Brief Overview

- **MEC-enabled UAV** use cases require optimized trajectory and routing plans to overcome limited charging capacity.
- Optimize UAV route for quick target retrieval by minimizing distance and maximizing speed, while recharging at city charging stations through Smart Grids and Renewable Energy.
- **Minimize total UAV travel distance** while respecting power constraints, as it sequentially visits each target and may need to recharge at a charging station, as depicted in the given **figure**. [3]



A route is established for a UAV to cover all monitoring objectives, with multiple stops for recharging at different stations.

Literature Review

Literature Table 1

Author's Name	Implemented Strategy
Shetty et al. [4]	Divided the strategic routing of UAVs into two stages: target allocation and TSP-dependent route planning for each UAV.
Casbeer and Holsapple [5]	Transformed the assignment of UAVs into a VRP with precedence limitations and resolved it by utilizing a column generation technique.

Literature Table 2

Author's Name	Implemented Strategy
Thibbo-tuwawa et al [6]	Addressed the task of planning missions for unmanned aerial vehicles (UAVs) by incorporating factors such as battery life and payload weight into the **Capacitated Vehicle Routing Problem (CVRP)**.
Guerrero and Bestaoui [7]	Implemented a **Zermelo-TSP algorithm** in order to calculate the most efficient path for a UAV, while also taking into account the impact of wind.

Literature Table 3

Author's Name	Implemented Strategy
Wen et al. [8]	Aimed to reduce both the overall travel time and the number of UAVs required for the homogeneous fleet in an expanded CVRP that incorporates distance and weight as key factors.
Coelho et al. [9]	Presented a green UAV routing problem with multiple objectives, which permits the drones to recharge at **designated charging stations**.

Research Gaps

Shortcomings of Previous Approaches

- The majority of current approaches to **UAV routing** rely on **traditional heuristics**, which require a lot of effort for **algorithmic development**. This can result in subpar performance and inadequate computation, particularly when dealing with **large-scale issues**. To address these challenges, alternative methods are required.
- The **limitations of current methodologies** can be addressed through the use of **Deep Reinforcement Learning**, which has the potential to outperform classic heuristic methods and other learning-based approaches.

The Advantages of DRL over Traditional Heuristics

DRL's Key Attributes for Effective Problem-Solving

Here are some of the **key features** of **DRL (Deep Reinforcement Learning)** which makes it more powerful than other traditional heuristics in solving the problem.

A. Exploration and Exploitation: DRL algorithms can balance the exploration of new trajectories with the exploitation of known good trajectories by using an exploration strategy, such as **epsilon-greedy**, to encourage the UAV to try out new trajectories during training.

Contd...

B. Long-term planning: DRL can handle long-term planning by optimizing a **cumulative reward signal** that incentivizes the UAV to visit charging stations and edge nodes strategically to maximize its overall performance.

C. Generalization: DRL can generalize its learning to handle new scenarios that it has not encountered during training by learning a policy that can adapt to new situations.

D. Adaptation to changing environments: DRL can generalize its learning to handle new scenarios that it has not encountered during training by learning a policy that can adapt to new situations.

Contd...

E. Policy optimization: DRL can optimize a policy that maps the current state of the UAV to an action that maximizes its expected reward. This allows the UAV to learn the best actions to take in different scenarios based on its **past experiences**.

F. Learning from experience: DRL learns from its own experience by adjusting its trajectory based on the feedback it receives from the environment. This allows the UAV to learn from its own mistakes and improve its performance over time.

Demonstrating the Power of DRL: A Survey of Related Works

Literature Table

Author's Name	Problem Statement	Implemented Strategy
Jingxuan Chen et al. [10]	This paper proposes **resource allocation** for **multi-UAV-aided Mobile Edge Computing** using a novel approach in a still nascent and under-studied area.	**UMAP** algorithm optimizes UAV movement, MU association, and power control to minimize system cost, using **MDP and DDPG** with constraint punishment in the reward.
Jian Yang et al. [11]	This paper proposes task offloading in Mobile Edge Computing (MEC) using **multi-agent deep reinforcement** learning to enable **cooperative decision-making** among agents for optimized task allocation.	A **multi-agent deep reinforcement** learning-based cooperative task offloading approach for Mobile Edge Computing (MEC) that optimizes **task allocation** by allowing agents to make decisions collaboratively.
Zunliang Wang et al. [12]	This paper addresses the problem of routing in **UAV swarm networks** by proposing a **multi-agent reinforcement learning** approach that enables drones to learn how to efficiently route data packets through a network.	The proposed solution in this paper is a **multi-agent reinforcement learning** approach for UAV swarm networks that utilizes **Q-learning and prioritized experience replay** to train agents to make routing decisions based on network conditions and achieve optimal performance.

Contd...

Author's Name	Problem Statement	Implemented Strategy
Peng, H., & Wang [13]	The paper addresses the development of an energy harvesting reconfigurable **intelligent surface** for Unmanned Aerial Vehicles (UAVs) using robust **deep reinforcement learning**. The key problem statement involves optimizing **energy harvesting and surface reconfiguration strategies** to enhance UAV communication performance in a dynamic environment.	The implemented strategy involves leveraging robust deep reinforcement learning to dynamically adjust the configuration of an intelligent surface on a UAV for **optimal energy harvesting**, aiming to enhance wireless communication performance in varying environmental conditions.
Zhang et al. [14]	The paper addresses the development of an **energy- and cost-efficient transmission strategy** for Unmanned Aerial Vehicle (UAV) trajectory tracking control. The key problem is optimizing the UAV's communication strategy using a **deep reinforcement learning approach** to minimize energy consumption and operational costs during trajectory tracking.	The implemented strategy involves employing deep reinforcement learning to optimize the transmission strategy of a UAV during trajectory tracking. This aims to achieve energy and cost efficiency by dynamically adapting the communication parameters based on the UAV's operational context.

System Model & Proposed Methodology

Problem Formulation

- **Objective:** Mathematically formulate UAV routing with multiple charging stations influenced by.
- Given:
  - S - Charging stations.
  - U - UAVs.
  - G - Geographical space.
  - A - Actions for UAVs.
  - S - State space.
  - R - Rewards.
- Goal: Find optimal policy π^* **maximizing cumulative reward.**

Contd...

- **Deep Q Network:** $Q(s,a)$: Q-value function.
- Optimal $Q^*(s,a)$: Maximum expected cumulative reward.
- Update rule:

$$Q(s,a) \leftarrow Q(s,a) + \alpha \left(R + \gamma \max_{a'} Q(s',a') - Q(s,a) \right)$$

- **Double Q Learning:** Mitigates overestimation bias.
- Two Q-value functions Q_1 and Q_2.
- Update rule for $Q1$:

$$Q1(s,a) \leftarrow Q1(s,a) + \alpha \left(R + \gamma Q2(s', \arg\max_{a'} Q1(s',a')) - Q1(s,a) \right)$$

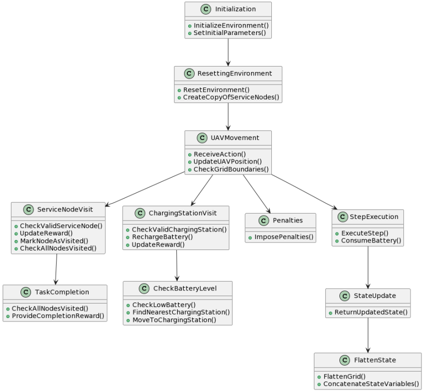
- Update rule for $Q2$:

$$Q2(s,a) \leftarrow Q2(s,a) + \alpha \left(R + \gamma Q1(s', \arg\max_{a'} Q2(s',a')) - Q2(s,a) \right)$$

System Model

Grid World Environment Setup:

- Initialize the grid with specified dimensions and place the depot
- Randomly deploy charging stations and service nodes within the grid
- Set initial parameters for the UAV: battery level, reward, and position
- Define the state space, including a flattened grid representation and the current battery level



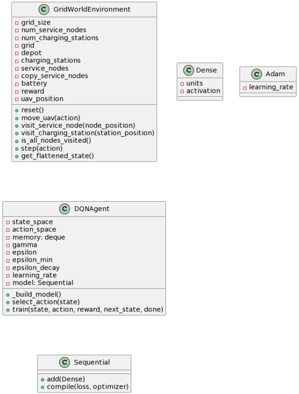
Flowchart of Grid World Environment Setup

Contd...

Deep Q-Learning Algorithm:

- Build a Q-network model with input dimension corresponding to the state space
- Initialize an experience replay buffer for efficient learning from past experiences
- **while** Training **do**
    - Select actions based on an epsilon-greedy exploration strategy
    - Execute actions in the environment and observe the resulting state, reward, and done flag
    - Store experiences in the replay buffer
    - Sample a minibatch from the replay buffer for training
    - Update the Q-network using gradient descent and the Bellman equation
    - Decay the exploration rate (ϵ) over time
- **end while**

Contd...



GridWorldEnvironment

- grid_size
- num_service_nodes
- num_charging_stations
- grid
- depot
- charging_stations
- service_nodes
- copy_service_nodes
- battery
- reward
- uav_position

- reset()
- move_uav(action)
- visit_service_node(node_position)
- visit_charging_station(station_position)
- is_all_nodes_visited()
- step(action)
- get_flattened_state()

Dense

- units
- activation

Adam

- learning_rate

DQNAgent

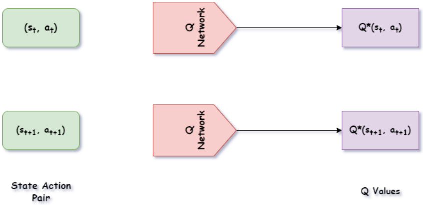
- state_space
- action_space
- memory: deque
- gamma
- epsilon
- epsilon_min
- epsilon_decay
- learning_rate
- model: Sequential

- _build_model()
- select_action(state)
- train(state, action, reward, next_state, done)

Sequential

- add(Dense)
- compile(loss, optimizer)

Components of DQN algorithm



(s_t, a_t)

Q Network

$Q^*(s_t, a_t)$

(s_{t+1}, a_{t+1})

Q Network

$Q^*(s_{t+1}, a_{t+1})$

State Action Pair

Q Values

Basic architecture of DQN

Contd...

Reward Calculation:

$$\text{Reward} = R_{\text{vis\_serv}} + R_{\text{vis\_charg}} - R_{\text{batt\_con}} + R_{\text{task\_com}}$$

Integration of Environment and Algorithm:

```
Initialize the grid world environment and the DQN agent
for Episode in Training do
    Reset the environment to its initial state
    while Episode not done do
        Select actions from the DQN agent
        Execute actions in the environment
        Train the DQN agent using experiences from the episode
    end while
end for
```

Proposed Methodology

Mathematical Formulation:

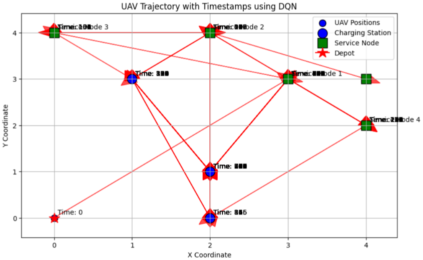
$$Q(s,a) \leftarrow Q(s,a) + \alpha \left[r + \gamma Q(s', \arg\max_{a'} Q(s',a')) - Q(s,a) \right] \tag{1}$$

DQN with Double Q Learning:

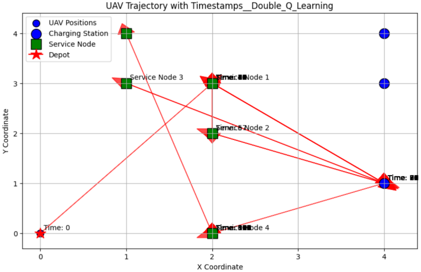
- Initialize two Q-networks with random weights: Q_1 and Q_2
- **for** *episode* $\leftarrow 1$ to N_{episodes}
  - Reset the environment
  - **while** not done
    - Select action using epsilon-greedy strategy
    - Execute action, observe reward and next state
    - Update Q-values using the Double Q Learning update equation
  - **end while**
- **end for**

Simulation Results

Visualization of UAV Trajectory

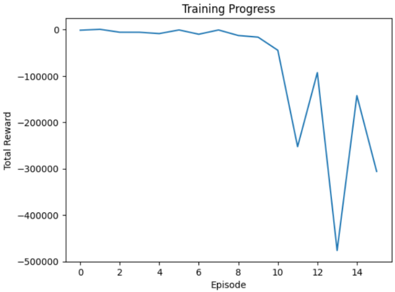


UAV Trajectory using DQN

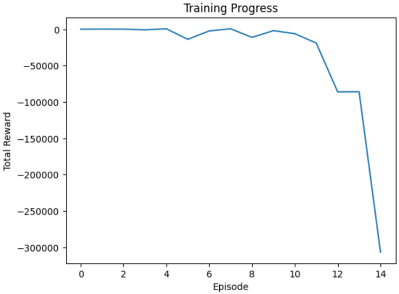


UAV Trajectory using DQN with Double-Q-Learning

Cumulative Rewards over episodes

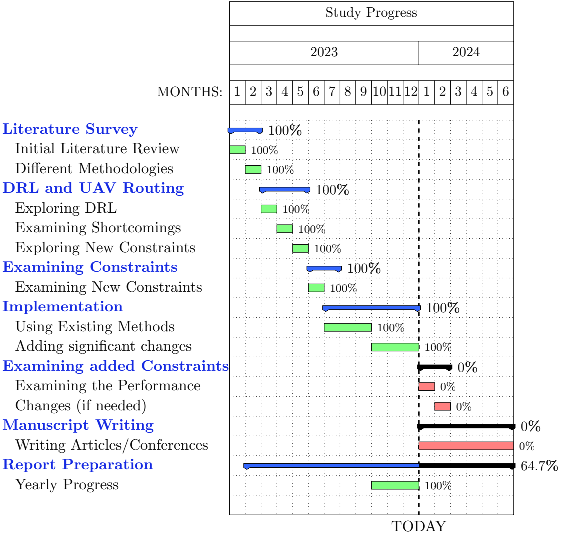


Rewards using DQN



Rewards using DQN with Double-Q-Learning

Timeline


Study Progress
2023
2024
MONTHS:
1
2
3
4
5
6
7
8
9
10
11
12
1
2
3
4
5
6
Literature Survey
100%
Initial Literature Review
100%
Different Methodologies
100%
DRL and UAV Routing
100%
Exploring DRL
100%
Examining Shortcomings
100%
Exploring New Constraints
100%
Examining Constraints
100%
Examining New Constraints
100%
Implementation
100%
Using Existing Methods
100%
Adding significant changes
100%
Examining added Constraints
0%
Examining the Performance
0%
Changes (if needed)
0%
Manuscript Writing
0%
Writing Articles/Conferences
0%
Report Preparation
64.7%
Yearly Progress
100%
TODAY

Conclusion

Key Takeaways

- **Adaptability:** DRL models showcase adaptability to dynamic and complex environments, adjusting their strategies based on learned experiences.
- **Learning from Interaction:** DRL agents learn from interaction with the environment, enabling them to discover optimal policies through trial and error.
- **Generalization:** Trained DRL models exhibit a degree of generalization, providing effective solutions for unseen scenarios beyond the training environment.
- **End-to-End Learning:** DRL frameworks facilitate end-to-end learning, allowing the model to directly map from raw sensory input to actions, reducing the need for handcrafted features.
- Link to github repo - **click here**

References I

[1] 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. 6949–6960, 2022.

[2] N. N. Ei, S. W. Kang, M. Alsenwi, Y. K. Tun, and C. S. Hong, “Multi-uav-assisted mec system: Joint association and resource management framework,” in *2021 International Conference on Information Networking (ICOIN)*, Jeju Island, Korea (South), 2021, pp. 213–218.

[3] M. Fan, Y. Wu, S. Li, Y. Zhang, and S. Cui, “Deep reinforcement learning for uav routing in the presence of multiple charging stations,” *IEEE Transactions on Vehicular Technology*, 2022.

[4] V. K. Shetty, M. Sudit, and R. Nagi, “Priority-based assignment and routing of a fleet of unmanned combat aerial vehicles,” *Computers & Operations Research*, vol. 35, no. 6, pp. 1813–1828, 2008.

[5] D. W. Casbeer and R. W. Holsapple, “Column generation for a uav assignment problem with precedence constraints,” *International Journal of Robust and Nonlinear Control*, vol. 21, no. 12, pp. 1421–1433, 2011.

[6] A. Thibbotuwawa, G. Bocewicz, P. Nielsen, and B. Zbigniew, “Planning deliveries with uav routing under weather forecast and energy consumption constraints,” *IFAC-PapersOnLine*, vol. 52, no. 13, pp. 820–825, 2019.

[7] J. A. Guerrero and Y. Bestaoui, “Uav path planning for structure inspection in windy environments,” *Journal of Intelligent & Robotic Systems*, vol. 69, no. 1, pp. 297–311, 2013.

[8] T. Wen, Z. Zhang, and K. K. Wong, “Multi-objective algorithm for blood supply via unmanned aerial vehicles to the wounded in an emergency situation,” *PloS one*, vol. 11, no. 5, p. e0155176, 2016.

[9] B. N. Coelho, V. N. Coelho, I. M. Coelho, L. S. Ochi, R. Haghnazar, D. Zuidema, M. S. Lima, and A. R. da Costa, “A multi-objective green uav routing problem,” *Computers & Operations Research*, vol. 88, pp. 306–315, 2017.

[10] J. Chen, X. Cao, P. Yang, M. Xiao, S. Ren, Z. Zhao, and D. O. Wu, “Deep reinforcement learning based resource allocation in multi-uav-aided mec networks,” *IEEE Transactions on Communications*, 2022.

References II

[11] J. Yang, Q. Yuan, S. Chen, H. He, X. Jiang, and X. Tan, "Cooperative task offloading for mobile edge computing based on multi-agent deep reinforcement learning," *IEEE Transactions on Network and Service Management*, 2023.

[12] Z. Wang, H. Yao, T. Mai, Z. Xiong, X. Wu, D. Wu, and S. Guo, "Learning to routing in uav swarm network: A multi-agent reinforcement learning approach," *IEEE Transactions on Vehicular Technology*, 2022.

[13] H. Peng and L. C. Wang, "Energy harvesting reconfigurable intelligent surface for uav based on robust deep reinforcement learning," *IEEE Transactions on Wireless Communications*, 2023.

[14] M. Zhang, S. Wu, J. Jiao, N. Zhang, and Q. Zhang, "Energy and cost-efficient transmission strategy for uav trajectory tracking control: A deep reinforcement learning approach," *IEEE Internet of Things Journal*, vol. 10, no. 10, pp. 8958–8970, 2022.

Thank You!