DOI: 10.13009/EUCASS2023-663
Aerospace Europe Conference 2023 – 10TH EUCASS – 9TH CEAS

Mission Planning of Manned-Unmanned Aircraft Teaming based on Reinforcement Learning: Suppression of Enemy Air Defences(SEAD) Mission

Hyeon Jun Lee, Bo Hoon Moon*, Chan Kwag*, Aye Aye Maw* and Jae-Woo Lee*†*
Konkuk Aerospace Design·Airworthiness Research Institute (KADA), Konkuk University
120, Neungdong-ro, Gwangjin-gu, Seoul, Republic of Korea
rain9138@gmail.com · qhgns5714@gmail.com · gleam1021@naver.com · ayeayemaw@konkuk.ac.kr · jwlee@konkuk.ac.kr

Abstract
This study explores the development of centralized mission planning for Unmanned Aerial Vehicles (UAVs) in collaboration with manned aircraft. We adopt a Proximal Policy Optimization (PPO) trained single agent to simulate a Suppression of Enemy Air Defenses (SEAD) scenario. Our goal is to master optimal mission strategies. Tested under various environmental conditions, our model demonstrates a 78% success rate in neutralizing enemy defenses across 100 tests. The significant success of our model underlines its potential application in future warfare scenarios, representing a substantial progression in the domain of aerial warfare and reinforcement learning application.

1. Introduction

Dating back to the advent of the experimental unmanned bomb aircraft, the Kettering Bug, during World War I, unmanned systems have played a significant role in military operations. As the years passed and technology progressed, these unmanned systems have seen extensive utilization in various theaters of war. From the B-17 unmanned bombers and Goliath unmanned bomb vehicles in World War II to the drones utilized in the Middle Eastern conflict and by the U.S. Air Force and Navy in the Vietnam and Gulf Wars, the role and capabilities of these unmanned systems have continually evolved.

One significant transformation propelled by technological advancements has been the shift in the relationship dynamics between humans and unmanned systems. What began as a unilateral operator-system relationship has evolved into a complex and cooperative strategy known as 'Manned-Unmanned Teaming (MUM-T)'. This evolution has been characterized by the growing importance of these systems in enhancing situational awareness, increasing lethality, and improving survivability on the battlefield [1].

In the context of modern, complex battlefield scenarios, effective implementation of MUM-T strategies is paramount. Unmanned Aerial Vehicles (UAVs), once primarily used for surveillance purposes, are now entrusted with more complex tasks that necessitate a high degree of collaboration with manned aircraft. Suppression of Enemy Air Defenses (SEAD) missions, for instance, require sophisticated mission planning techniques that can adapt to dynamic battlefield environments and optimize the use of mixed resources.

This research project seeks to address the development of such advanced mission planning techniques, specifically focusing on centralized planning for UAVs operating in collaboration with manned aircraft during SEAD missions. Our objectives in this pursuit are threefold:

1) Design an efficient centralized battlefield operation architecture to enhance coordination between manned and unmanned systems in dynamic combat environments.
2) Construct a reinforcement learning environment that uses battlefield monitoring data to simulate real-world scenarios, thereby facilitating robust AI agent training.
3) Implement the PPO algorithm to train a single-agent model, enabling it to learn the optimal mission planning strategy effectively.

DOI: 10.13009/EUCASS2023-663

The implications of this research are significant and far-reaching. From an operational perspective, the development of such a centralized mission planning system could revolutionize the planning and execution of military missions, leading to a significant enhancement in operational efficiency and a reduction in mission failures. On a theoretical level, the research contributes to the broader field of reinforcement learning by providing empirical evidence of the effectiveness of the PPO algorithm in handling complex, real-world scenarios. Consequently, the insights gained from this research have the potential to guide the future development of military technology, particularly with regard to the planning, control, and deployment of unmanned systems in various military operations.

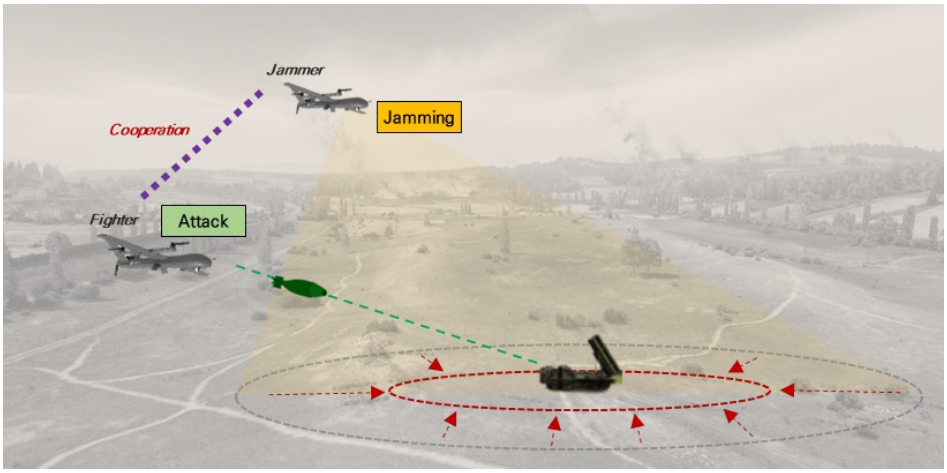

Figure 1: MUM-T Mission with 2 UAVs in SEAD Mission

2. Literature Review and Background

2.1 MUM-T, Autonomous Control Levels (ACL), and Mission Planning Strategies

Efficient data gathering and swift decision-making processes are vital in modern warfare scenarios. MUM-T has gained traction as it combines the strengths of manned and unmanned aircraft to enable strategic information gathering and execution, boosting combat efficiency. MUM-T has been designated as a core component of future warfare strategies by the U.S. Department of Defense[1], and several nations are actively developing it [2][3].

The framework for enabling MUM-T involves scenario development, workload reduction for operators, and enhancement of UAVs' autonomy. Scenario development focuses on employing both manned and unmanned aircraft in precise battlefield environments. This development is based on the analysis of current mission statuses of manned and unmanned aircraft using Model Based Systems Engineering (MBSE) [4] or detailed procedural breakdowns [5][6]. Workload reduction research concentrates on quantifying and mitigating the decision-making difficulty and the number of decisions during mission execution. Technologies and interfaces are designed to alleviate operator workloads through autonomous path planning, hierarchical mission planning, among others [7].

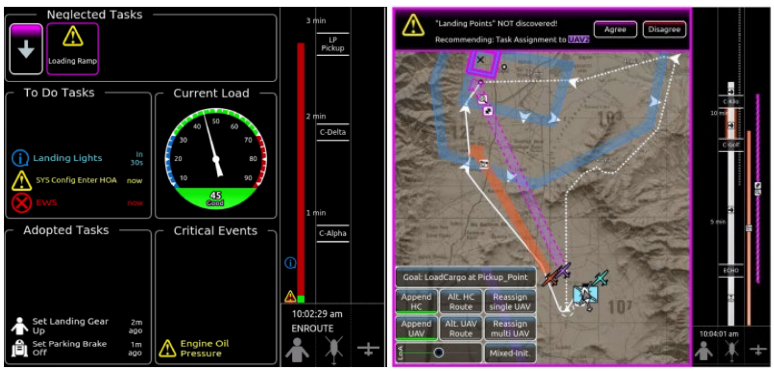

Figure 2: Transparency elements in the mission planning interface [7]

DOI: 10.13009/EUCASS2023-663

Autonomy, as per the AFRL ACL[8], is key to determining the UAVs' ability to independently perform the Observe, Orient, Decide, and Act (OODA) loop. Mission planning can either be decentralized, with each UAV independently making decisions [9], or centralized, where a single entity like a GCS makes decisions based on data from all UAVs [10]. A framework utilizing Leaders and Sub-Leaders has been proposed to address processing and communication challenges [11].

Table 1: AFRL ACL Chart [8]

Level	Level Descriptor	Observe	Orient	Decide	Act
		Perception / Situational Awareness	**Analysis / Coordination**	**Decision Making**	**Capability**
5	Real Time Multi Vehicle Cooperation	Sensed awareness / Local sensors to detect external targets (friendly and threat) fused with off board data	All below with Prognostic Health Mgmt; Group diagnosis and resource management	On board trajectory replanning / Optimizer for current and predictive conditions; Collision avoidance	Group accomplishment of tactical plan as externally assigned; Air collision avoidance; Possible close air space separation (1:200yards); Formation in non-threat conditions

The choice between centralized and decentralized planning depends on the autonomy level of the UAV and the operational environment. This study assumes an ACL Level 5 for the involved UAVs, implying their capability for evasion maneuvers and autonomous flight. However, for group operations and strategy establishment, a centralized architecture has been implemented.

2.2 Application of Reinforcement Learning in UAV Mission Planning

RL, where an agent learns to make decisions to maximize rewards through interaction with its environment, has significant applications in UAV mission planning. The PPO algorithm, an RL technique, can be applied to everything from lower-level controls like attitude control and route planning to higher-level controls such as mission point planning and optimal action set determination [12].

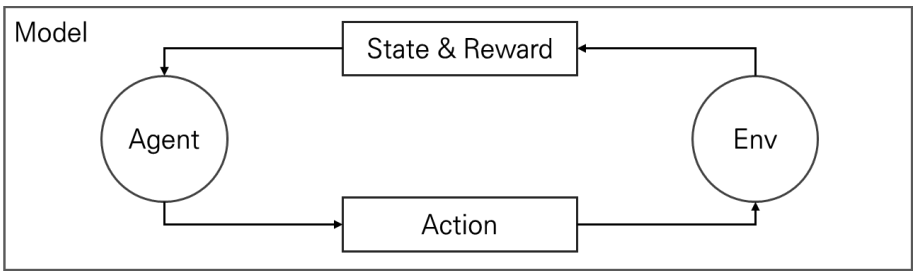

Figure 3: Reinforcement Learning

Yue et al. applied Deep RL to MUM-T mission planning, specifically for shooting down SAMs using Jammers and Fighters, controlling each UAV's heading and velocity to move towards the target point based on pre-determined Jamming and Attack Points. Zhan et al. applied PPO and MAPPO algorithms to multi-UAV strike missions but did not consider explicit cooperation between UAVs [13].
In contrast, our research implements a centralized architecture for higher-level controls in UAV mission planning, differentiating UAVs as Jammers and Fighters. A cooperative scenario is learned through the PPO algorithm, which involves shooting down SAMs and moving towards the next destination. The agent learns the optimal route and Jamming Point for the Jammer and the optimal route and Attack Point for the Fighter, which will be explained in detail in Section 3.2.

DOI: 10.13009/EUCASS2023-663

3. Methods

3.1 Centralized Mission Planning Architecture

Centralized Mission Planning Architecture refers to an advanced technological architecture enabling efficient coordination and management of UAVs in complex and dynamic combat scenarios. This architecture collects data from various sources of information, evaluates the situation in real-time, and plans and executes optimal strategies to maximize the success potential of the overall mission [14].

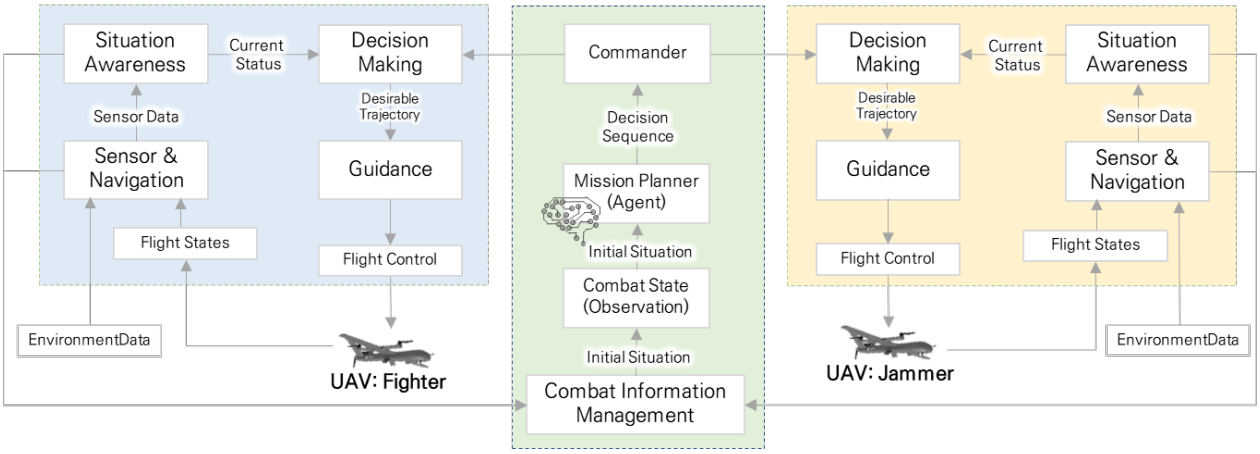

Figure 4: Centralized Mission Planning

The main components of this architecture are as follows:

1) **Combat Info Management:** This component continually monitors the current combat situation and tracks information to provide real-time battlefield intelligence. Information sources vary widely, including various sensors, sensor networks, and human observation, enabling a deep understanding of the dynamic and complex combat environment. This corresponds to the process of collecting information about the environment in reinforcement learning, providing the first step for an effective learning process.
2) **Combat State (Observation):** At this stage, battlefield information is provided to the agent. Various information collected in the field is processed in real-time and delivered to the reinforcement learning agent. This allows the agent to understand the current situation through integrated battlefield situation awareness, predict future possibilities, and decide on the next action.
3) **Mission Planner (Agent):** As the core element of the central, this reinforcement learning-based agent makes optimal actions based on incoming real-time combat situation data. This decision process is carried out by a pre-trained reinforcement learning model, which learns how to achieve the objectives of the mission in a complex environment.
4) **Commander:** Lastly, the agent's decision is passed onto the Commander for execution. The actions decided by the agent are delivered as commands to the actual UAVs, enabling specific tasks such as movement, target detection, and attack.

Therefore, Centralized Mission Planning Architecture realizes the strategy of collecting and processing data from various sources of information, planning and adjusting UAV's actions adaptive to real-time battlefield conditions. This enables real-time strategic decision-making and quick response, enhancing overall combat efficiency and survivability.

3.2 Construction of Reinforcement Learning Environment

We have developed a tailored reinforcement learning environment for the MUM-T problem. In this environment, we have deployed a single Fighter UAV, a Jammer, and a SAM system, each with predefined attack ranges and jamming distances. The primary objective of the mission is to collaboratively engage in jamming operations, neutralize the targeted SAM system, and subsequently eliminate it by maneuvering the Fighter UAV. Successful completion of the mission is determined by reaching the designated Goal Point.
We constructed a custom reinforcement learning environment for MUM-T in the context of UAV mission planning. In our MUM-T environment, we deployed one Fighter UAV, a Jammer, and SAM systems, each with defined attack

DOI: 10.13009/EUCASS2023-663

ranges and jamming distances. The ultimate objective of the mission is to perform cooperative jamming with the Jammer, rendering the SAM incapable of attacking, and subsequently destroying the SAM by maneuvering the Fighter UAV. Successful completion of the mission is achieved when the UAV reaches the final destination, referred to as the Goal Point.

To develop the environment, we utilized the Gym library, an open-source framework for reinforcement learning environments. The space in which the UAVs can move is represented as a 2D grid. Since the lower-level control aspects such as heading and velocity of the UAVs are assumed to be handled autonomously at AFRL ACL Level 5, the centralized mission planning framework focuses on the higher-level control responsible for planning mission-related values, namely waypoints and mission points, based on the information of multiple UAVs and the battlefield state. To facilitate the learning process, we discretized the mission space into a 30x30 grid, consisting of a total of 900 cells.

The action space for each UAV is defined as a discrete multi-action space, enabling each agent to independently select actions. The Fighter UAV and Jammer have five possible actions: left, right, up, down, and attack. Discretizing the action space simplifies learning and control [Figure 5, 6].

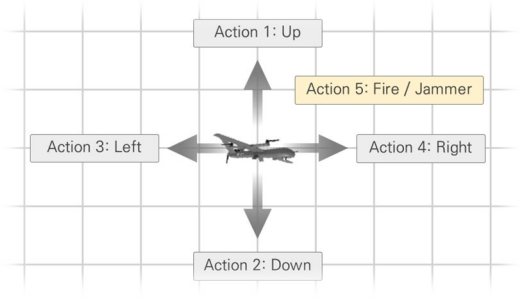

Figure 5: UAV's Action

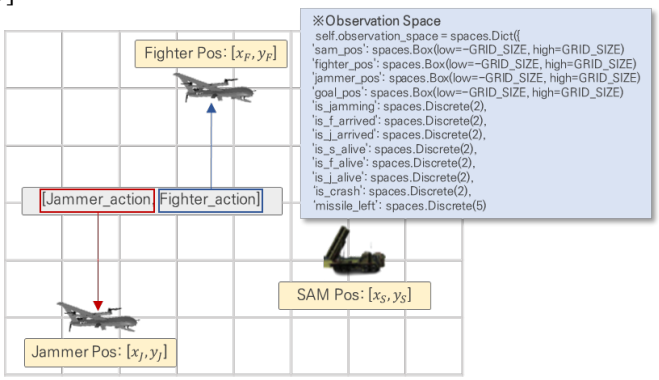

Figure 6: Discrete Multi-action Space and Observation Space

At each time step, the agents move within the grid environment according to their chosen actions. We impose boundary conditions (penalties) to prevent the UAVs from moving outside the grid boundaries. Additionally, we handle potential collisions between the Fighter and Jammer by detecting collisions and assigning penalties accordingly.

To address the collaboration aspect among the UAVs, we model specific functionalities and interactions between the agents. When the Jammer engages in jamming and the SAM is not within the attack range, a penalty is incurred. However, if the SAM is within the attack range, successful jamming leads to a reward, rendering the SAM inoperable. The Fighter has a total of five attack opportunities and failing to attack (when the SAM is not within the attack range) results in losing one attack opportunity and receiving a penalty. On the other hand, if the SAM is within the defined attack range, the SAM is neutralized, and a reward is given. Importantly, the Fighter cannot attack if it is not engaged in jamming, as the attack range of the Fighter is shorter than the jamming distance.

3.3 Training Method of the Agent using PPO Algorithm

In this study, we utilized PPO, a type of RL algorithm and a variant of the Advantage Actor Critic algorithms. The objective function of PPO, known as the Surrogate Objective, aims to update the policy in a way that minimizes the difference between the current policy and the previous policy [15][17].

The Surrogate Objective is represented as follows:

$$Surrogate\ Function = \hat{E}_t\left[\frac{\pi_\theta(a_t \mid s_t)}{\pi_{\theta old}(a_t \mid s_t)}\hat{A}_t\right] \quad (1)$$

In the formula, a a_t and s_t denote the action and state at time step t, respectively, π_θ represents the policy, and $\hat{A}_t$ stands for the estimator of the advantage function at timestep, while $\hat{E}_t$ refers to the expected value of the function. At this point, when we refer to the ratio of the previous policy to the current policy as $r_i(\theta)$, it can be represented as follows:

DOI: 10.13009/EUCASS2023-663

$$r_i(\theta) = \left[\frac{\pi_\theta(a_t \mid s_t)}{\pi_{\theta old}(a_t \mid s_t)}\right] \tag{2}$$

The PPO algorithm employs a strategy known as Clipping to restrict the similarity between the current and previous policies. Clipping refers to keeping the value of $r_i(\theta)$ between 1-ϵ and 1+ϵ, and it can be represented as follows:

$$\text{clip}\,(r_i(\theta), 1-\epsilon, 1+\epsilon) = \begin{cases} 1+\epsilon, \text{ if } r_i(\theta) \geq 1+\epsilon \\ 1-\epsilon, \text{ if } r_i(\theta) \leq 1-\epsilon \\ r_i(\theta), \text{ otherwise} \end{cases} \tag{3}$$

By employing this Clipping technique, we can represent the optimization problem with the new objective function $L_t^{clip}(\theta)$ and θ as follows:

$$\begin{gathered} L_T^{clip}(\theta) = \min\{r_i(\theta)A^{\pi_{OLD}}(x_t, u_t), \text{clip}\,(r_i(\theta), 1-\epsilon, 1+\epsilon)A^{\pi_{OLD}}(x_t, u_t)\} \\ \theta \leftarrow \arg\max_\theta \sum_{t=0}^{\infty} E_{x_0:u_t \sim p_\theta OLD}(x_0{:}\,u_t)\left[L_T^{clip}(\theta)\right] \end{gathered} \tag{4}$$

1) $A^{\pi_{OLD}}(x_t, u_t) > 0$: In state x_t, the selected action u_t is in a better state compared to the average. Hence, the new policy will have a higher probability value than the policy that selected that action. This increases the value of $r_i(\theta)$, albeit limiting it to 1+ϵ
2) $A^{\pi_{OLD}}(x_t, u_t) < 0$: In state x_t, the selected action u_t is in a worse state compared to the average. Therefore, the new policy will have a lower probability value than the policy that selected that action. This decreases the value of $r_i(\theta)$, albeit limiting it to 1-ϵ

Algorithm 1 PPO, Actor-Critic Style [16]

for iteration = 1,2, ... **do**
  for actor = 1,2, ... , N **do**
    Run policy $\pi_{\theta_{\text{old}}}$ in environment for T timesteps
    Compute advantage estimates $\hat{A}_1, \ldots, \hat{A}_T$
  end for
  Optimize surrogate L wrt θ, with K epochs and minibatch size $M \leq NT$
$\theta_{\text{old}} \leftarrow \theta$
end for

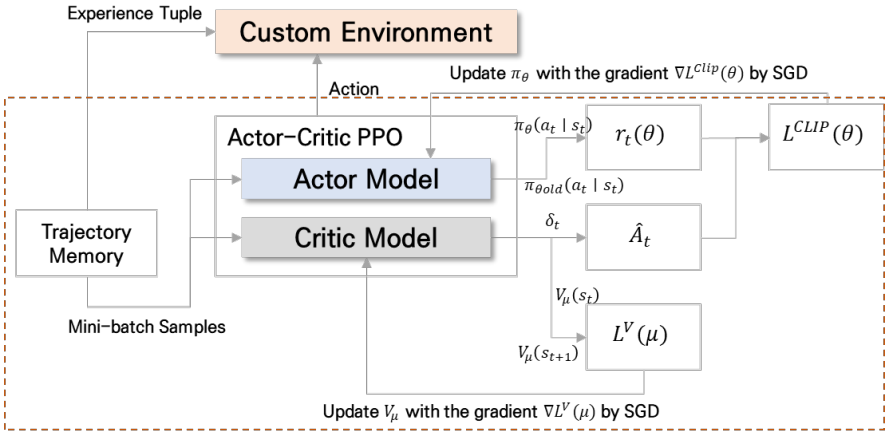

Figure 7: General Architecture of PPO based Training [18]

DOI: 10.13009/EUCASS2023-663

4. Training Results and Analysis

In this section, we present the results obtained from the application of our proposed mission planning strategy on SEAD missions. The results were achieved by simulating a variety of SEAD mission scenarios within our reinforcement learning environment and observing the performance of our PPO-trained agent.

4.1 Training Setup

In our research, we employed the PPO algorithm to facilitate effective learning for the Fighter and the Jammer. Although the learning scenario is configured for the Fighter to attack the SAM, there exist several significant constraints.
Firstly, the Fighter is limited to having a maximum of only five missiles. Moreover, if the Jammer does not precede in jamming the SAM, the Fighter is not permitted to carry out an attack. Additionally, it is necessary to find a path that prevents the two entities from colliding, and they are not allowed to move beyond a designated space. Lastly, the agent must plan its mission utilizing only the information available from its observable points.
To overcome these constraints, we designed a reward system as detailed in Table 3. This system is composed of two components, Penalty and Reward. The Penalty component enables the agent to learn swiftly and in the correct direction, while the Reward component is applied to actions such as attack and jamming, encouraging the discovery of optimal policies under different situations. Through this method, we anticipate that the agent can learn efficiently and achieve its objectives while satisfying the constraints.

Table 2 Reward

Penalty	Reward
1) Step 2) Crash 3) Mission Fail	1) Conservation 2) Survival 3) Arrival 4) Attack

.
In the PPO algorithm, several key hyperparameters significantly influence performance. The learning rate determines the step size for model parameter updates; a lower rate fosters stable yet slower training, while a higher rate can hasten convergence but may lead to instability. The discount factor, or gamma, arbitrates the importance of immediate rewards versus future ones, with the optimal value being goal dependent. The batch size influences the number of samples utilized in each update, affecting both the speed of learning and memory requirements. Smaller batch sizes may slow down learning but are less memory-intensive, while larger ones could speed up learning at the expense of more memory. Lastly, the number of steps (n_steps) per training step also has a significant effect on both learning speed and memory usage. In our study, we adjusted and compared several parameters to find the optimal set. The results of this are presented in Table 4.

Table 3: Hyperparameter Setting

Parameter	Value
Leaning Rate	0.003
Gamma	0.99
N_steps	1024
Batch Size	128

Another significant element in reinforcement learning is the 'Observation,' which signifies the current state of the environment or the representation of the state. This serves as critical information that the agent uses to make decisions. A trained agent understands the environment based on the Observation Space used during the learning process, and links this understanding to the actual battlefield situation and the determination of the optimal policy. The following description illustrates the types of Observation Space and corresponding values utilized in our research [Table 5].

DOI: 10.13009/EUCASS2023-663

Table 4: Observation Spaces

Type	Data Format	Code
Sam Position	Integer	'sam_pos': spaces.Box(low=-GRID_SIZE, high=GRID_SIZE, shape=(2,), dtype=np.int32)
Fighter Position	Integer	'fighter_pos': spaces.Box(low=-GRID_SIZE, high=GRID_SIZE, shape=(2,), dtype=np.int32)
Jammer Position	Integer	'jammer_pos': spaces.Box(low=-GRID_SIZE, high=GRID_SIZE, shape=(2,), dtype=np.int32)
Goal Position	Integer	'goal_pos': spaces.Box(low=-GRID_SIZE, high=GRID_SIZE, shape=(2,), dtype=np.int32)
Is Fighter Alive	True / False	'is_f_alive': spaces.Discrete(2)
Is Jammer Alive	True / False	'is_j_alive': spaces.Discrete(2)
Number of Weapons	Integer	num_wp': spaces.Box(low=0, high=NUM_W, shape=(1,), dtype=np.int32)
Is SAM Alive	True / False	'is_s_alive': spaces.Discrete(2),
Is Jamming	True / False	'is_jamming': spaces.Discrete(2)
Is Fighter Arrived	True / False	'is_f_arrived': spaces.Discrete(2)
Is Jammer Arrived	True / False	'is_j_arrived': spaces.Discrete(2)
Crashed	True / False	'is_crash': spaces.Discrete(2)

4.2 Training Results

In Section 4.2, we provide a detailed analysis of the training results from our research. The outcomes are presented in terms of the Episode Length Mean, Episode Reward Mean, and Value Loss.

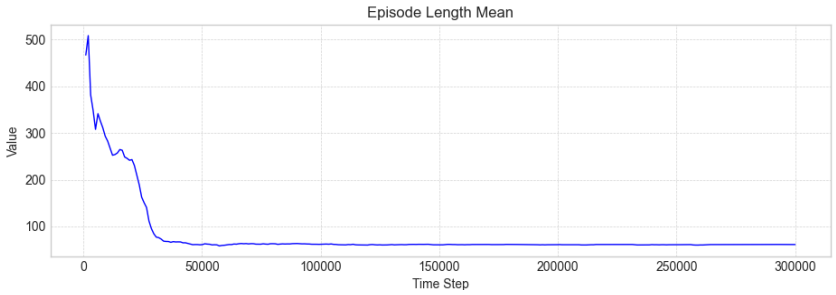

Figure 8: Episode Length Mean

"Episode Length Mean" in reinforcement learning indicates the average number of steps taken by the agent before it reaches a terminal state. This metric is utilized to assess the progress and efficiency of the agent's learning. Initially, the episode length was over 500 but gradually decreased to below 100, and by the end of the training, it registered at 61.1[Figure 8].

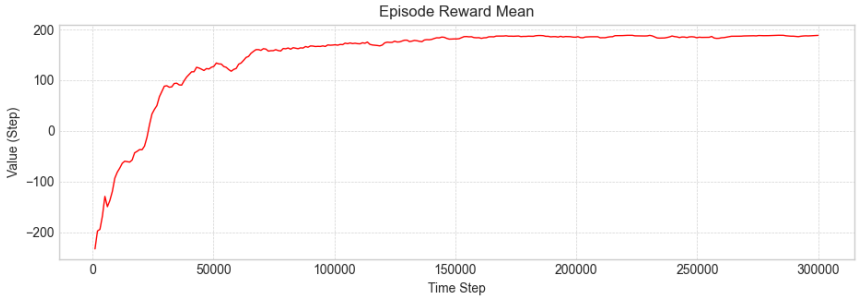

Figure 9: Episode Reward Mean

In reinforcement learning, the "Episode Reward Mean" represents the average total reward an agent accumulates throughout an episode. This metric serves as a crucial indicator of the agent's performance, with a higher average reward typically signifying that the agent is learning actions leading to more favorable outcomes in the given environment. In our scenario, the agent receives a reward of 200 upon successful completion of the mission and incurs

DOI: 10.13009/EUCASS2023-663

a penalty of -0.2 for each step taken. Taking this into account, the convergence of the reward function towards 188 validates the effective progress of the training.

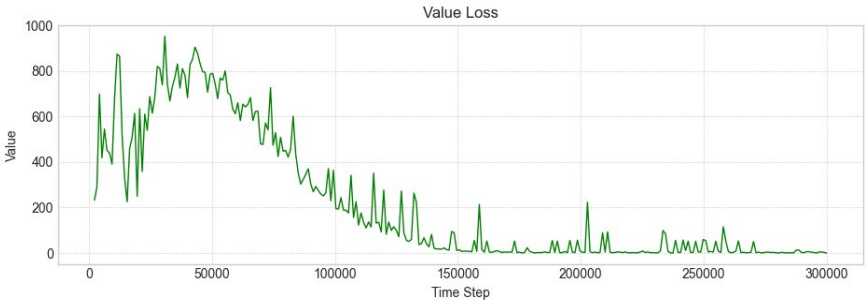

Figure 10: Value Loss

In reinforcement learning, "Value Loss" signifies the difference between the agent's predicted and actual returns. A smaller value loss indicates more accurate predictions of returns, thereby reflecting the agent's effective learning. As seen in Figure 10, the value loss decreases as the training progresses, with the average over 10,000 time steps converging approximately to 1.2.

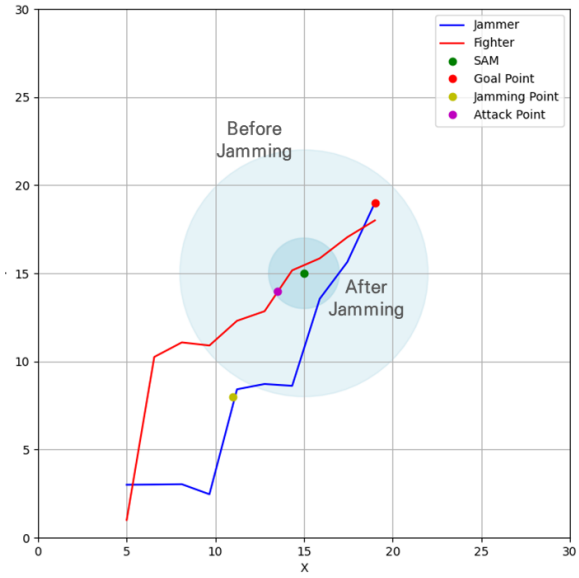

Figure 11: The Smooth Interpolated Trajectory of a Successful Mission

Figure 11 represents the simulation results using the trained model. The trajectories have been processed with smooth interpolation to ensure that overlapping paths are not visible. The figure includes the Jamming Point and Attack Point. The larger circle represents the attack range of the SAM before jamming, while the smaller circle represents the SAM's range after jamming. As evident from the results, each UAV has planned its path to avoid collision with other UAVs while successfully completing the mission.

4.3 Performance Evaluation

To evaluate the performance of the trained model, we assessed its responsiveness to changes in the environment. This was done by randomly varying the positions of the SAM, the goal, and the starting points of the two UAVs. The termination criterion for the tests was set at 100 steps, and a successful evaluation was based on the model successfully intercepting the SAM and passing through the goal point within 100 steps. The results are as follows:

Table 5: Evaluation Result

Test Cases	Rate of Success
100	78%

DOI: 10.13009/EUCASS2023-663

5. Conclusions

In this study, we have designed and validated a centralized mission planning strategy for UAVs engaged in SEAD operations. Leveraging the capabilities of the PPO algorithm, we trained a model to discern and implement the optimal strategy for cooperative SAM neutralization by Fighter and Jammer units.

Our training outcomes showcased considerable improvements across various performance indicators. The Episode Length Mean fell from over 500 to a mere 61.1, evidencing the agent's efficiency in mission execution. The Episode Reward Mean stabilized around 188, underscoring the agent's learned proficiency in selecting rewarding actions while considering penalties. Concurrently, the Value Loss diminished over training iterations, indicating the agent's enhanced accuracy in forecasting returns.

We further examined the adaptability and robustness of our model under environmental modifications. By randomizing the positions of SAMs, goals, and starting points, we evaluated the model's capability to successfully neutralize SAMs and reach the goal within limited steps. The results manifested high success rates under the specified constraints.

The implications of our developed centralized mission planning strategy for real-world applications are noteworthy. By effectively integrating and leveraging manned and unmanned platforms, military operations could benefit from superior situational awareness, augmented lethality, and enhanced survivability. The integration of AI-driven mission planning systems holds the potential to curtail mission failures, optimize resource allocation, and ultimately safeguard valuable resources and lives.

Nonetheless, the scope for further research and enhancements remains. Future work could delve into more intricate and dynamic battlefield environments, incorporate advanced multi-agent frameworks for precise collaboration between Fighter and Jammer units, and contemplate communication limitations for realistic decision-making.

In conclusion, our research furnishes a comprehensive solution for mission planning in SEAD operations by harnessing the potential of reinforcement learning and the PPO algorithm. The outcomes affirm the efficacy of the proposed strategy in enhancing mission efficiency and success rates. This work significantly contributes to the military technology domain, emphasizing the criticality of integrating manned and unmanned systems for optimal mission outcomes.

Funding: This research was supported by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education (No. 2020R1A6A1A03046811)
This work is supported by the Korea Agency for Infrastructure Technology Advancement(KAIA) grant funded by the Ministry of Land, Infrastructure and Transport (Grant RS-2022-00143965).

References

[1] Office of the Secretary of Defense. (2005). Unmanned Aircraft Systems Roadmap 2005-2030. U.S. Department of Defense.
[2] U.S. Department of Defense. (2014). Unmanned Systems Integrated Roadmap, FY2013-2038.
[3] Roth, G. (2020). Transparency for a Workload-Adaptive Cognitive Agent in a Manned-Unmanned Teaming Application. IEEE.
[4] Michelson, S. (2019). Concepts for Manned Unmanned Teaming Behaviors in Model Based Systems Engineering. NDIA Ground Vehicle Systems Engineering and Technology Symposium.
[5] Kim, J.-H., Seo, W., Choi, K., & Ryoo, C.-K. (2019). Analysis of SEAD Mission Procedures for Manned-Unmanned Aerial Vehicles Teaming. Journal of the Korean Society for Aeronautical & Space Sciences, 47(9), 678-685.
[6] The United States Naval War College. (2011). Joint Military Operations Reference Guide.
[7] Roth, G., Schulte, A., Schmitt, F., & Brand, Y. (2020). Transparency for a Workload-Adaptive Cognitive Agent in a Manned–Unmanned Teaming Application. IEEE Transactions on Human-Machine Systems, 50(3), 225-233. https://doi.org/10.1109/THMS.2019.2914667.
[8] Sholes, E. C. (2007). Evolution of a UAV Autonomy Classification Taxonomy. 2007 IEEE Aerospace Conference, 1-16.
[9] Ponda, S., Redding, J., Choi, H.-L., How, J. P., Vavrina, M., & Vian, J. (2010). Decentralized Planning for Complex Missions with Dynamic Communication Constraints. 2010 American Control Conference, Marriott Waterfront, Baltimore, MD, USA.
[10] Hwang, N. E., Kim, H. J., & Kim, J. G. (2023). Centralized Mission Planning for Multiple Robots Minimizing Total Mission Completion Time. Applied Sciences, 13, 3737. https://doi.org/10.3390/app13063737.

DOI: 10.13009/EUCASS2023-663

[11] Zhou, X., Wang, W., Wang, T., Li, X., & Li, Z. (Year). A Research Framework on Mission Planning of the UAV Swarm. System of System Engineering Group, College of Information System and Management, National University of Defense Technology.
[12] Bøhn, E., Coates, E. M., Moe, S., & Johansen, T. A. (2019). Deep Reinforcement Learning Attitude Control of Fixed-Wing UAVs Using Proximal Policy Optimization. 2019 International Conference on Unmanned Aircraft Systems (ICUAS), Atlanta, GA, USA, 523-533. doi: 10.1109/ICUAS.2019.8798254.
[13] Yue, L., Yang, R., Zhang, Y., Yu, L., & Wang, Z. (2022). Deep Reinforcement Learning for UAV Intelligent Mission Planning. Complexity, 2022, 3551508. https://doi.org/10.1155/2022/3551508
[14] Maw, A.A.; Tyan, M.; Nguyen, T.A.; Lee, J.-W. iADA*-RL: Anytime Graph-Based Path Planning with Deep Reinforcement Learning for an Autonomous UAV. Appl. Sci. 2021, 11, 3948. https://doi.org/10.3390/app11093948
[15] Zhan, G., Zhang, X., Li, Z., Xu, L., Zhou, D., & Yang, Z. (2022). Multiple-uav reinforcement learning algorithm based on improved ppo in ray framework. Drones, 6(7), 166.
[16] Schulman, J., Wolski, F., Dhariwal, P., Radford, A., & Klimov, O. (2017). Proximal Policy Optimization Algorithms. arXiv preprint arXiv:1707.06347v2 [cs.LG].
[17] Liu, Z., Li, J., Zhang, P., Ding, Z., & Zhao, Y. (2022). An AGC Dynamic Optimization Method Based on Proximal Policy Optimization. Frontiers in Energy Research, 10, 947532. https://doi.org/10.3389/fenrg.2022.947532
[18] Raşchka, A., Hill, A., Gleave, A., Kanervisto, A., Ernestus, M., & Dormann, N. (2021). Stable-Baselines3: Reliable Reinforcement Learning Implementations. Journal of Machine Learning Research, 22, 1-8.