Wildfire Contention and Suppression using Drones: A Multi-Agent Deep Reinforcement Learning Approach

Maxime Collignon, Adolfo Perrusquía, Antonios Tsourdos, Weisi Guo

***Abstract*— Wildfires present serious risks to both natural ecosystems and smart living environments due to their highly destructive and inherently unpredictable behavior. In recent years, unmanned aerial vehicles (UAVs) have been increasingly deployed for wildfire detection and suppression in order to mitigate ecological damage and protect wildlife. However, many existing reinforcement learning (RL)-based methods depend on overly simplified environmental representations that do not adequately reflect the complex and stochastic nature of fire propagation. To overcome this limitation, this work introduces a cooperative multi-UAV framework for wildfire containment based on a multi-agent Deep Q-Network (MADQN). The proposed method incorporates a tailored observation space alongside a temperature-driven reward function to improve environmental perception and promote effective coordination among agents. Comprehensive simulation studies are performed to evaluate the performance of the approach, demonstrating its effectiveness in wildfire suppression and containment tasks.**

I. Introduction

Climate change is driving a sustained rise in global temperatures [1], intensifying the frequency and severity of extreme weather events such as storms, heatwaves, and floods [2]–[6]. Among these, wildfires have emerged as one of the most destructive and recurrent natural disasters in recent decades [7], [8]. Despite the significant advances in detection and response technologies, wildfire management remains a formidable challenge. The unpredictable and rapidly changing dynamics of fire spread, coupled with the hazardous conditions created by smoke, heat, and limited visibility, make firefighting and search-and-rescue (SAR) operations extremely dangerous. Rescuers must frequently make rapid decisions in uncertain environments, often with incomplete situational awareness, placing their lives at considerable risk [9]. As climate change amplifies the scale and intensity of wildfires, the demand for faster, safer, and enhanced autonomy in emergency response becomes critical.

Traditional approaches to wildfire response rely heavily on human coordination supported by aerial firefighting aircraft (e.g., Canadair) and ground teams [10]. While effective in localized scenarios, these methods are constrained by limited endurance, delayed response times, and human safety considerations. Recent advances in robotics and artificial intelligence, however, have introduced new possibilities for autonomous unmanned aerial vehicles (UAVs) to assist or even replace human responders in high-risk environments. UAVs can provide real-time monitoring, rapid deployment, and high manoeuvrability. However, to be truly effective in wildfire suppression or SAR operations, they must also be able to cooperate intelligently as a swarm under uncertain and dynamic conditions.

Maxime Collignon, Adolfo Perrusquía, Antonios Tsourdos, and Weisi Guo are with the Faculty of Engineering and Applied Sciences, Cranfield University, MK43 0AL Bedford, UK. email: adolfo.perrusquia-guzman@cranfield.ac.uk

The coordination of multiple UAVs in disaster scenarios poses several open research challenges, including collective perception, decentralized decision-making, communication robustness, and adaptive planning in uncertain environments. These challenges are exacerbated in wildfire contexts, where the environment changes continuously and unpredictably. Addressing these issues requires autonomous control frameworks capable of learning, adapting, and optimizing behaviors collaboratively. Multi-agent Deep Reinforcement Learning (MADRL) offers a promising path toward scalable, data-driven coordination for fire suppression and rescue missions.

Motivated by these challenges, this work proposes a learning based multi-agent UAV framework for wildfire suppression and containment. Unlike conventional rule-based or heuristic control approaches, our method leverages MADRL and a temperature-based perception system to enable UAVs to autonomously learn coordinated actions that minimize fire spread and mitigate damage across uncertain environments.

A. Related work

The deployment of robotic and aerial swarms for search and rescue (SAR) missions has gained increasing attention in recent years [11]. In particular, [12] outlines several key challenges that still limit the large-scale adoption of robot and UAV swarms in operational SAR scenarios. These challenges are grouped into four main research directions: (i) perception and planning, (ii) communication, (iii) human-robot interaction [13], and (iv) cost-to-performance trade-offs. Among these, this work concentrates on perception and planning, with the goal of enabling autonomous UAV swarm behavior for SAR tasks in wildfire-affected environments.

A variety of methodological approaches have been explored to address SAR-related coordination and decision-making problems. For example, Monte Carlo Tree Search (MCTS) has been applied to multi-robot systems to support planning in SAR missions [14]. More recently, reinforcement learning (RL) and multi-agent reinforcement learning (MARL) techniques have been widely adopted to train individual agents or coordinated swarms capable of operating in complex SAR settings [15], [16]. In [17], a RL-based maritime UAV is developed for post-disaster SAR operations, where trajectory optimization is used to improve mission efficiency compared to conventional approaches.

Several MARL-based strategies have also been proposed for target search and tracking in challenging environments. In [18], a cooperative multi-robot system is trained using adversarial MARL to improve strategic target discovery in cluttered terrains. Similarly, [19] introduces a MARL framework for multi-agent SAR (MASAR) in dynamic and partially observable environments, where target behavior is stochastic and continuously evolving, as often observed in disaster scenarios. This approach demonstrates improved performance over traditional MARL baselines. In [20], a heterogeneous MARL framework is designed for maritime SAR operations, enabling coordinated trajectory planning and resource allocation among UAVs and rescue vessels while maintaining robustness to communication failures. Additional research has focused on improving scalability and robustness in partially observable SAR settings. In [21], a distributed MARL architecture (MADAC) is proposed for aerial search missions, incorporating Long Short-Term Memory (LSTM) networks [22] to mitigate the effects of imperfect communication and incomplete environmental information. In the context of urban disaster response, [23] introduces ResQ, a MARL-based system that optimizes the deployment of rescue agents to efficiently locate victims in rapidly changing environments.

MARL techniques have also been extended to UAV-based wildfire monitoring and suppression tasks. In [24], a combination of Q-learning and Value Decomposition Networks (VDN) is used to train UAV swarms for wildfire spread tracking, achieving superior performance compared to existing benchmark methods. Furthermore, [25] proposes a Multi-Agent Deep Q-Network (MADQN) approach for coordinated fire suppression, where UAV agents learn to strategically deploy fire retardants in simulated wildfire scenarios, outperforming heuristic suppression techniques [26].

II. Methodology

The proposed methodology adopts a MADQN architecture and a relative complex environment model of wildfires under variable density vegetation. Each agent has the same neural network architecture and takes their own decisions. However, each agent shares its policy to the other agents. The actions applied by the interaction agents-environment are separated in the environment model and they are not related to the policy of the agents. Similarly, the communication between agents is modelled as an interaction between the Q-network and the environment. Here, the environment provides information regarding the local observations of the agents, the global information that includes fire location and positions of the agents, and the communication of the positions of the nearest agents. In the sequel of this section we describe each element of the proposed methodology in detail.

A. Wildfire Environment Model

The wildfire environment is based on the model introduced in [27], which simulates fire spread and thermal diffusion over a heterogeneous forest grid. For clarity, we briefly summarize the main components below.

1) Vegetation Density: We represent the forest as a two-dimensional grid $G \in \mathbb{N}_N^2$ of size $N \times N$, where each cell corresponds to a forest patch. Each cell (i, j) has an associated vegetation density $d_{ij} \in [0, 1]$, generated by applying a Gaussian filter of width σ_d to an initial uniform random field. This produces smooth regions of varying vegetation density, which directly influence local fire propagation. Cells with higher d_{ij} values are more flammable and facilitate faster fire spread. We set $\sigma_d = 10$ to achieve a realistic spatial heterogeneity.

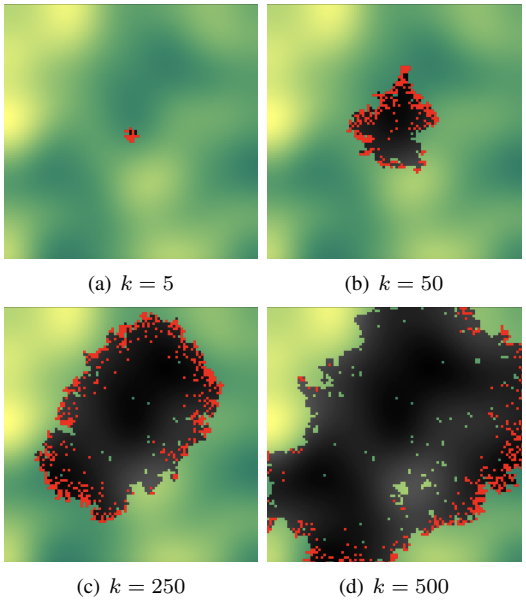
2) Fire Propagation: Fire propagation follows a local stochastic process over the four-neighbour topology of the grid. Each cell X can be in one of three states:

$$s_X^k \in \{\mathcal{H}, \mathcal{F}, \mathcal{D}\} = \{\text{Healthy}, \text{ Burning}, \text{ Burned}\}.$$

At each time step k:

- A burning cell ignites each healthy neighbour with probability $\gamma_X = \alpha d_X$, where α is the base fire transmission rate and d_X is the local vegetation density.
- A burning cell extinguishes with probability β, after which it becomes permanently burned.
- Burned cells remain inert and cannot reignite.

This local process defines a Markov chain at the cell level, and the global wildfire dynamics arise from the joint transitions of all cells in G. We use $\alpha = 0.25$ and $\beta = 0.1$, which yield thin and discontinuous fire fronts consistent with heterogeneous forest landscapes [27]. Fig. 1 shows a simulation example of how fire spreads based on the proposed vegetation density model.



(a) $k = 5$ (b) $k = 50$ (c) $k = 250$ (d) $k = 500$

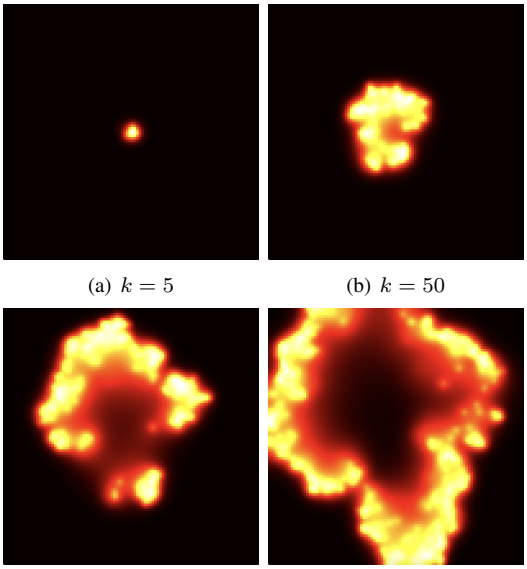
Fig. 1. Wildfire simulation

3) Temperature Field: To emulate thermal observations captured by UAV-mounted infrared sensors, a temperature grid T evolves alongside the fire states. At each step, burning cells are set to a maximum temperature T_{max}, while heat diffuses spatially using a Gaussian kernel with standard

deviation σ_T and cools over time with factor $c \in [0, 1]$. This process can be expressed as

$$T^{k+1} = c\left(T_{\max} \mathbb{1}_{\mathcal{F}}(s^{k+1}) + \mathbb{1}_{\mathcal{H}\cup\mathcal{D}}(s^{k+1}) \odot T^k\right) * w_{\sigma_T},$$

where $*$ denotes the 2D convolution, $\odot$ stands to the Hadamard product, and $\mathbb{1}_{\mathcal{F}}(s)$ is the binary mask of burning cells. We set $T_{\max} = 100$, $\sigma_T = 1$, and $c = 0.965$ to achieve a realistic heat diffusion and cooling profile. This simplified model reproduces plausible temperature gradients near the fire front without the computational cost of solving full heat equations. Fig. 2 shows the temperature map of the previous fire propagation example.



(a) $k = 5$ (b) $k = 50$

(c) $k = 250$ (d) $k = 500$

Fig. 2. Wildfire temperature anomaly.

B. Multi-Agent Deep Q-Network

In this work, we adopt a multi-agent Deep Q-Network (MADQN) formulated within a centralized training with decentralized execution (CTDE) paradigm. This framework serves as the learning backbone for promoting coordinated behavior among agents, making it particularly well suited to the wildfire suppression scenario considered here. The central principle of CTDE is to exploit global state and shared experiential data during training, while ensuring that each agent operates independently at execution time. Specifically, during deployment, each agent relies on its own instance of the trained policy to make decisions in a decentralized manner. This paradigm has been widely adopted across both value-based methods (e.g., DQN) and policy-gradient approaches such as REINFORCE and proximal policy optimization (PPO) [28], [29]. The proposed MADQN architecture employs a shared Q-network and a shared target network across all agents. Training is performed using mini-batches of transitions sampled from a centralized replay buffer. At each timestep, every agent stores its experience tuple $\langle s_m^k, a_m^k, r_m^k, s_m^{k+1}\rangle$ into this shared memory, enabling collective learning from the aggregated experiences of all agents. As a result, the replay buffer captures a unified representation of multi-agent interactions, which is then used to update the neural networks. During execution, each agent receives a synchronized copy of the learned parameters θ and follows an identical policy π_θ for action selection. While acting independently, agents may still exchange and utilize global environmental information, including the full state representation s, to support coordinated decision-making. Both the Q-network and target network consist of two fully connected hidden layers with 512 and 256 neurons, respectively, using ReLU activation functions. Additionally, we incorporate pre-training adjustments inspired by [27]. In particular, the initial fire region is expanded to a 3×3 area, and vegetation maps are regenerated to avoid low-density regions near ignition points. This modification effectively mitigates the occurrence of short-lived fire dynamics during training, leading to more stable learning behavior.

1) Fire suppression and contention problem definition: Let define $\mathcal{E}_{N,\sigma_d,\alpha,\beta,T_{\max},\sigma_T,c}$ as a given environment where a fire starts. We define $\kappa_{1,\mathcal{E}}$ as the proportion of cells of G that are burnt after the end of the fire i.e., after an infinite number of steps:

$$\kappa_{1,\mathcal{E}} = \lim_{k\to\infty} \frac{1}{N^2} \sum_{X\in G} \mathbb{1}_{\mathcal{D}}(s_X^k). \tag{1}$$

Here, the fire suppression and contention problem is to train M agents using MADQN to determine an optimal policy Π_{1,θ_1} that minimizes $\mathbb{E}[\kappa_{1,\mathcal{E}}]$. As previously mentioned, the MADQN episode starts with a 3×3 burning square in the centre of the grid, and the simulation ends when there are no remaining burning cells in the grid, that is, when the fire is either suppressed or all the cells are burnt. In the sequel of this paper, we interchangeably refer to the "score" as the quantity to be minimized if it corresponds to $\kappa_{1,\mathcal{E}}$ or the quantity to be maximized if it corresponds to $1 - \kappa_{1,\mathcal{E}}$.

2) Agents modelling: We consider a high-level navigation problem in a large grid environment. The agents initially start at random positions located on the edges of the grid, namely on the four borders of the environment. The agents have the ability to drop retardant only if the ground temperature under the UAVs is above a pre-defined threshold. We define $\mathcal{M}^k$ as the set that contains the positions occupied by the agents at step k. For an agent at cell X, the corresponding space is $I_1(X) = \{X + (i, j) \mid (i, j) \in \{-1, 0, 1\}^2\}$.

If an agent mitigates a fire in a given burning cell, then this cell becomes a dead cell. Here, the incorporation of the agents into the environment implies that we need to modify the Markov Process of the fire propagation and consider the effect of the agents as well. This can be modelled as follows,

$$\begin{aligned} p^k_{X,\mathcal{F}\to\mathcal{D}} &= \max\left(\beta, \mathbb{1}\left(\sum_{X\in I_1(X)} \mathbb{1}_{\mathcal{M}^k}(X) \geq 1\right)\right) \\ &= \max\left(\beta, \mathbb{1}\left(I_1(X) \cap \mathcal{M}^k \neq \varnothing\right)\right) \qquad (2) \\ p^k_{X,\mathcal{F}\to\mathcal{F}} &= 1 - \max\left(\beta, \mathbb{1}\left(I_1(X) \cap \mathcal{M}^k \neq \varnothing\right)\right) \qquad (3) \end{aligned}$$

These probabilities mean that a burning cell requires one or more agents in its I_1 space to be extinguished, or that

it naturally extinguishes with probability β given by (2). In other terms, if the intersection of $I_1(X)$ and the set containing the agents' positions is not the empty set, that is, there is one or more agents within $I_1(X)$, then the cell is "cleared" to contain the fire with probability 1. Otherwise, the probability of transitioning to state $\mathcal{D}$ remains β as given in (2). Finally, it is assumed that the agents have no influence on the temperature near burning cells. In that case, the drones are only removing heat sources.

3) State space and Action space: For the state space, let define $m \in [\![1,\ M]\!]$ as the agent's index. We write X_m^k to denote the position of agent m. At step k, the state space, which is equivalent to the neural network input, consists in five features given by

- The initial position of the fire ($X_{\mathcal{F}}^0$) relative to the position of the agent X_m^k: $u_m^k = X_{\mathcal{F}}^0 - X_m^k$. This data is available in real-life scenarios [30]. This feature is useful for the agents as it allows them to know in which direction they need to move at the beginning of an episode;
- Let $m_c^k = \underset{m' \in \mathbb{N}_M \setminus m}{\arg\min} ||X_{m'}^k - X_m^k||_2$ be the closest agent to m. The distance to m_c^k denoted as $v_m^k = ||X_{m_c}^k - X_m^k||_2$ and the position of m_c^k relative to the one of m denoted as $u_{c,m}^k = X_{m_c}^k - X_m^k$ are two features of the state space model. This permits the agents to be aware of the proximity to other agents. In this case, they could be less efficient compared with the case where they operate in separate areas;
- A "help" feature denoted as

$$h_m^k = \mathbb{1}\left(\max_{X \in I_1(X_{m_c}^k)} T^k(X) > \frac{T_{\max}}{2}\right) \cdot \mathbb{1}\left(\max_{X \in I_1(X_m^k)} T^k(X) < \frac{T_{\max}}{5}\right)$$

  is designed. Here, the help feature h_m^k is equal to 1 if the maximum temperature in the I_1 space of the closest agent m_c^k is superior to $T_{\max}/2$, whilst the temperature of the m agent is less than $T_{max}/5$. This means that the agent will be able to know if his closest neighbour needs help and if the agent itself can provide help. In this case, we want the agent to move towards his closest neighbour. This can lead to better results if the given agent is not currently suppressing fire, whilst his closest neighbour might be struggling in high temperature zones;
- We propose to take the logarithm of the temperature as data for our local relative temperature map, so that the exponential nature of the temperature is diminished. The following input is proposed: a map of neighbouring logarithmic temperatures relative to the one of the agent's cell, in the grid $I_1(X_m^k)$. This map is exactly the size of $I_1(X_m^k)$. The corresponding matrix is

$$t_m^k = \left(\ln T^k(i,j) - \ln T^k(X_m^k)\right)_{(i,j) \in I_1(X_m^k)} \quad (4)$$

Therefore, the feature vector of the state space model can be summarized as

$$s_m^k = [u_m^k \ \ v_m^k \ \ u_{c,m}^k \ \ h_m^k \ \ t_m^k]^\top \quad (5)$$

The action space consists in 8 possible moves: up, down, left, right, upper left, upper right, lower left, lower right.

4) Reward function: At step k, for agent m, we split the reward into three components,

- A reward r_{clean}^k obtained when the agent extinguishes fire on a cell in $I_1(X_m^k)$. Here, we award the agent with $+\frac{1}{5}$ per cleaned cell, and punish with $-\frac{1}{10M}$ if no cell has been cleaned,
- A reward r_{temp}^k obtained if the agent moved to a cell which is hotter than its previous one. We award the agent with $+\frac{1}{10M}$ in this case,
- A reward r_{help}^k obtained when the closest agent needs help and when m can provide help ($h_m^k = 1$). If the agent moves towards its closest neighbour, we award the agent with $+\frac{1}{M}$. Otherwise, it receives a negative reward of $-\frac{1}{M}$.

5) Hyperparameters: We perform hyperparameter tuning with a *Random Search* over specific ranges of parameter values. This optimization is carried out in the DQN hyperparameters: $(lr, \gamma, \epsilon, \text{batch\_size})$. The final combination of hyperparameters that leads the best performance results were: learning rate $lr = 10^{-4}$, discount factor $\gamma = 0.98$, ε-probability of 0.15, and batch size of 32.

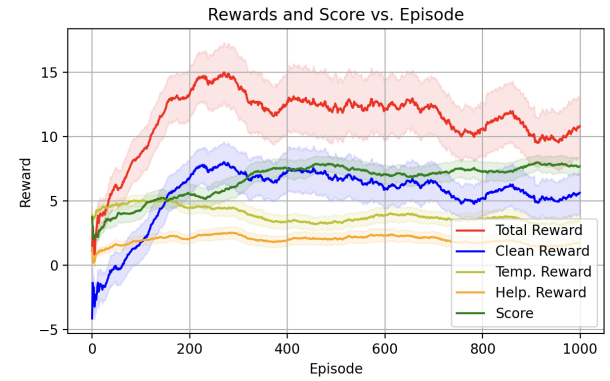
III. Results and Discussion

In this section, we train the proposed MADQN under the proposed forest environment with variable vegetation. Unless specified otherwise, the trainings are performed with 5 agents ($M = 5$) and the results correspond to the $M = 5$ case. We similarly study the influence of some of the simulation parameters, e.g., the number of agents.

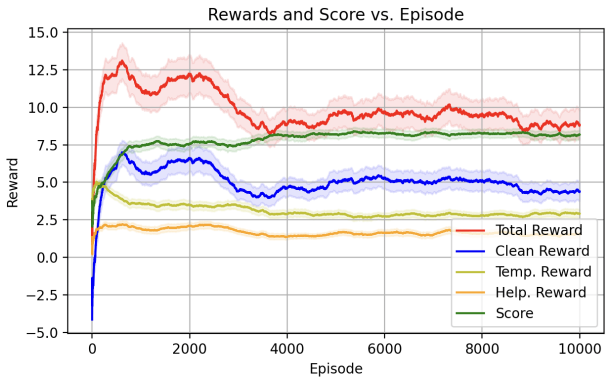
A. Reward curve

Fig. 3(a) displays the reward curve over the first $1,000$ epochs, using a moving average of window size 100. Fig. 3(b) shows the reward curve over $10,000$ epochs with a moving average of window size 500. In both graphs, we plot the total reward and the respective reward components.

The different window sizes account for the different maximum values. We additionally plot in both graphs the score curve, i.e., the score obtained by the agents at each episode multiplied by 10 (so that it ranges from 0 to 10). Both graphs enable identifying an initial surge of the return in the first 300 episodes. The reward slightly decreases afterwards before it stabilizes within a small neighbourhood. However, we can observe that the score exhibits an asymptotic and increasing performance. This means that the training of the MADQN is working properly and it is efficient given the constraints of the algorithm. On the other hand, as the training advances, the agents gain efficiency and reduce the number of burned cells at the end of each episode. As a result, the episodes are becoming shorter. Nevertheless, shorter episodes mean that agents have less time to earn reward by, for instance,



(a) $1,000$ epochs



(b) $10,000$ epochs

Fig. 3. Reward and score curves

cleaning fire on cells or moving towards cells with higher temperatures. Consequently, the reward curve is artificially decreasing and this is not tantamount to a drop of the performances. Indeed, the agents are in fact more efficient as they earn more reward per step at the end of the training than at the beginning.

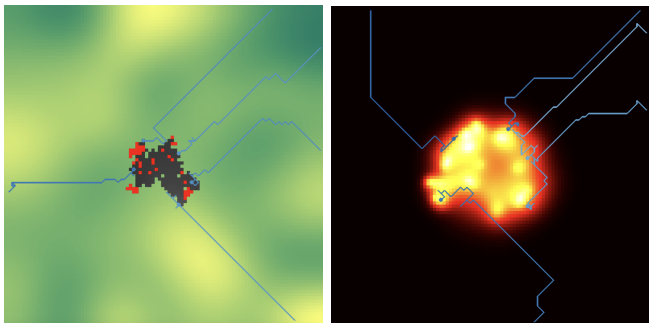
We also observe that the different components of the reward increase at different paces. The clean reward increases fast and ends very high, while the help and temperature rewards rise to lower maxima. Again, this is predictable since we normalized the reward to give more importance to the clean reward, which is the most important element of the total reward. The temperature and help rewards increase as well as can be seen in Fig. 3(b).

Fig. 4 shows a simulation of fire suppression and contention of the proposed MADQN after training.

From the results, we observe that the UAVs learn to get as close as possible to the fire at the beginning of an episode. Then, they try to stay not too close between them, in order to maximize efficiency. They finally move around the fire front to extinguish fire. As the fire front is discontinuous, the UAVs learn to quit a zone of high temperature just after extinguishing all the burning cells of this zone. This is done to find other areas where fire is spreading.

B. Ablation studies

We analyse the influence of both the *help* feature h_m^k (and the corresponding reward) and the logarithm in the local



(a) Classical map (b) Temperature map

Fig. 4. Example of a simulation where RL-based agents work to extinguish a wildfire.

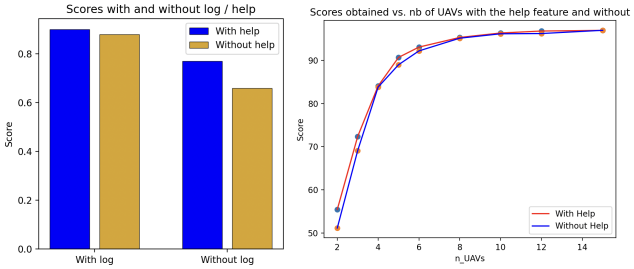
temperature map forwarded to the agent. We hypothesize that these elements are two of the main contributions for the proposed solution. Indeed, h_m^k introduces cooperation between UAVs, which is a key component when addressing natural disasters and catastrophes using intelligent or heuristic-based robotic agents. Then, the logarithm in the local temperature map is hypothesized to be essential to the approach, because without this element the agents will not understand the temperature data. To verify these hypotheses, we train three different models using MADQN described as follows: (i) a model without the *help* feature but with the logarithmic input; (ii) a model with the *help* feature but without the logarithmic input; and (iii) a model without both elements. The obtained results are summarized in Table I and the scores are given in Fig. 5(a).

TABLE I
ABLATION STUDIES OF THE MADQN. FINAL SCORES

Help \ Log	**Yes**	**No**
Yes	0.90	0.77
No	0.88	0.65

The results verify the hypotheses by showing how the score of the MADQN is degraded when one or both elements are missing. We also note that the *help* feature is less useful than the logarithmic input as it only enables the RL-trained agents to score 2% better, compared with 13% for the logarithm. However, it still has an interesting property in enhancing the agents' cooperation skills.

We conduct a further ablation study in terms of how much the number of agents influence in the final score. Fig. 5(b) shows the results of this ablation study. We note that the score increases with the number of agents and observe that the agents always score better with the *help* feature, no matter how many they are. Nonetheless, the difference between the two scores is higher when the agents are not numerous (three or less agents). Agents behave this way perhaps because scoring well requires more cooperation when there are a few agents, as the areas covered by all the UAVs are way smaller in this case.



(a) Logarithm in the temperature v.s. *help* feature

(b) No. of agents v.s. *help* feature

Fig. 5. Ablation studies scores

IV. Conclusion

In this work, we introduce a UAV-based wildfire suppression and containment strategy operating in a forest simulation environment. The environment accounts for spatially heterogeneous vegetation density, enabling a more faithful representation of wildfire propagation dynamics. A temperature-based sensing layer is incorporated to provide more reliable indicators of critical fire regions. A MADQN framework is designed to coordinate suppression and containment actions under a sparse reward structure. This formulation encourages efficient exploration while promoting cooperative behaviour among UAV agents in identifying and mitigating high-risk areas. Simulation experiments demonstrate the effectiveness of the proposed method.

Future work will focus on enriching the environmental model by integrating additional physical factors such as wind dynamics and humidity variations. Furthermore, we aim to investigate hybrid approaches that combine reinforcement learning with heuristic guidance to enhance convergence speed and improve overall wildfire mitigation performance.

References

[1] S. F. B. Tett, P. A. Stott, M. R. Allen, W. J. Ingram, and J. F. B. Mitchell, "Causes of twentieth-century temperature change near the earth's surface," *Nature*, vol. 399, no. 6736, pp. 569–572, 1999.

[2] A. A. Deo, D. Ganer, and G. Nair, "Tropical cyclone activity in global warming scenario," *Natural Hazards*, vol. 59, pp. 771–786, 2011.

[3] S. C. Chapman, N. W. Watkins, and D. A. Stainforth, "Warming trends in summer heatwaves," *Geophysical Research Letters*, vol. 46, no. 3, pp. 1634–1640, 2019.

[4] H. J. Smith, "Increasing heatwave activity," *Science*, 2019. [Online]. Available: https://api.semanticscholar.org/CorpusID:135140046

[5] Y. Hirabayashi, R. Mahendran, S. Koirala, L. Konoshima, D. Yamazaki, S. Watanabe, H. Kim, and S. Kanae, "Global flood risk under climate change," *Nature Climate Change*, vol. 3, pp. 816–821, 2013.

[6] C. Gruffeille, A. Perrusquía, A. Tsourdos, and W. Guo, "Disaster area coverage optimisation using reinforcement learning," in *2024 International Conference on Unmanned Aircraft Systems (ICUAS)*. IEEE, 2024, pp. 61–67.

[7] Y. Liu, J. A. Stanturf, and S. L. Goodrick, "Trends in global wildfire potential in a changing climate," *Forest Ecology and Management*, vol. 259, pp. 685–697, 2010.

[8] A. Kumar, A. Perrusquía, S. Al-Rubaye, and W. Guo, "Wildfire and smoke early detection for drone applications: A light-weight deep learning approach," *Engineering Applications of Artificial Intelligence*, vol. 136, p. 108977, 2024.

[9] P. Withen, "Climate change and wildland firefighter health and safety," *NEW SOLUTIONS: A Journal of Environmental and Occupational Health Policy*, vol. 24, pp. 577 – 584, 2015.

[10] M. El Debeiki, S. Al-Rubaye, A. Perrusquía, C. Conrad, and J. A. Flores-Campos, "An advanced path planning and uav relay system: Enhancing connectivity in rural environments," *Future Internet*, vol. 16, no. 3, p. 89, 2024.

[11] A. Perrusquía, W. Guo, B. Fraser, and Z. Wei, "Uncovering drone intentions using control physics informed machine learning," *Communications Engineering*, vol. 3, no. 1, p. 36, 2024.

[12] D. S. Drew, "Multi-agent systems for search and rescue applications," *Current Robotics Reports*, vol. 2, pp. 189–200, 2021.

[13] W. Yu and A. Perrusquia, *Human-Robot Interaction Control Using Reinforcement Learning*. John Wiley & Sons, 2021.

[14] C. A. Baker, S. Ramchurn, W. Teacy, and N. R. Jennings, "Planning search and rescue missions for uav teams," in *ECAI 2016*. IOS Press, 2016, pp. 1777–1782.

[15] E. Bildik, A. Tsourdos, A. Perrusquía, and G. Inalhan, "Decoys deployment for missile interception: A multi-agent reinforcement learning approach," *Aerospace*, vol. 11, no. 8, p. 684, 2024.

[16] X. Kong, Y. Xing, A. Tsourdos, Z. Wang, W. Guo, A. Perrusquia, and A. Wikander, "Explainable interface for human-autonomy teaming: A survey," *arXiv preprint arXiv:2405.02583*, 2024.

[17] B. Ai, M. Jia, H. Xu, J. Xu, Z. Wen, B. Li, and D. Zhang, "Coverage path planning for maritime search and rescue using reinforcement learning," *Ocean Engineering*, vol. 241, p. 110098, 2021.

[18] A. Rahman, A. Bhattacharya, T. Ramachandran, S. Mukherjee, H. Sharma, T. Fujimoto, and S. Chatterjee, "Adversar: Adversarial search and rescue via multi-agent reinforcement learning," in *2022 IEEE International Symposium on Technologies for Homeland Security (HST)*, 2022, pp. 1–7.

[19] X. Cao, M. Li, Y. Tao, and P. Lu, "Hma-sar: Multi-agent search and rescue for unknown located dynamic targets in completely unknown environments," *IEEE Robotics and Automation Letters*, vol. 9, no. 6, pp. 5567–5574, 2024.

[20] C. Lei, S. Wu, Y. Yang, J. Xue, and Q. Zhang, "Maritime search and rescue leveraging heterogeneous units: A multi-agent reinforcement learning approach," in *2023 IEEE/CIC International Conference on Communications in China (ICCC)*, 2023, pp. 1–6.

[21] V. Sadhu, C. Sun, A. Karimian, R. Tron, and D. Pompili, "Aerial-deepsearch: Distributed multi-agent deep reinforcement learning for search missions," *2020 IEEE 17th International Conference on Mobile Ad Hoc and Sensor Systems (MASS)*, pp. 165–173, 2020.

[22] B. Fraser, A. Perrusquía, D. Panagiotakopoulos, and W. Guo, "A deep mixture of experts network for drone trajectory intent classification and prediction using non-cooperative radar data," in *2023 IEEE Symposium Series on Computational Intelligence (SSCI)*. IEEE, 2023, pp. 1–6.

[23] Z. Yang, L. Nguyen, J. Zhu, Z. Pan, J. Li, and F. Jin, "Coordinating disaster emergency response with heuristic reinforcement learning," in *2020 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, 2020, pp. 565–572.

[24] A. Viseras, M. Meissner, and J. Marchal, "Wildfire front monitoring with multiple uavs using deep q-learning," *IEEE Access*, pp. 1–1, 2021.

[25] R. N. Haksar and M. Schwager, "Distributed deep reinforcement learning for fighting forest fires with a network of aerial robots," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2018, pp. 1067–1074.

[26] E. Thellier, A. Perrusquía, and A. Tsourdos, "Scalable and generalizable path planning for robotic navigation using transformer-based heuristic learning," *Information Sciences*, p. 123149, 2026.

[27] M. Collignon, A. Perrusquía, A. Tsourdos, and W. Guo, "Search and rescue operations in wildfires using unmanned aerial vehicles: A multi-agent deep reinforcement learning approach," *Neurocomputing*, p. 131211, 2025.

[28] R. Azzam, I. Boiko, and Y. Zweiri, "Swarm cooperative navigation using centralized training and decentralized execution," *Drones*, vol. 7, no. 3, 2023.

[29] T. Rashid, M. Samvelyan, C. S. De Witt, G. Farquhar, J. Foerster, and S. Whiteson, "Monotonic value function factorisation for deep multi-agent reinforcement learning," *Journal of Machine Learning Research*, vol. 21, no. 178, pp. 1–51, 2020.

[30] A. Genovese, R. D. Labati, V. Piuri, and F. Scotti, "Wildfire smoke detection using computational intelligence techniques," in *2011 IEEE International Conference on Computational Intelligence for Measurement Systems and Applications (CIMSA) Proceedings*, 2011, pp. 1–6.