Towards Self-Adaptive Software for Wildfire Monitoring with Unmanned Air Vehicles

Enrique Vílchez[1[0009−0009−7548−9641]], Javier Troya[1[0000−0002−1314−9694]], and Javier Cámara[1[0000−0001−6717−4775]]

ITIS Software, Universidad de Málaga, Spain
{enriquev, jtroya, jcamara}@uma.es

Abstract. Wildfires have evolved significantly over the last decades, burning increasingly large forest areas every year. Smart cyber-physical systems like small Unmanned Air Vehicles (UAVs) can help to monitor, predict, and mitigate wildfires. In this paper, we present an approach to build control software for UAVs that allows autonomous monitoring of wildfires. Our proposal is underpinned by an ensemble of artificial intelligence techniques that include: (i) Recurrent Neural Networks (RNNs) to make local UAV predictions about how the fire will spread over its surrounding area; and (ii) Deep Reinforcement Learning (DRL) to learn policies that will optimize the operation of the UAV team.

Keywords: wildfire monitoring · artificial intelligence · UAVs

1 Introduction

Every year, large extensions of terrain are burned by wildfires. In Spain, 2022 has been the worst in the last 15 years, with 267,939.64 hectares of forest devastated [3]. The use of human-supervised, software-controlled Small Unmanned Air Vehicles (UAVs) like drones is considered as a promising approach to effectively monitor and mitigate the impact of wildfires without the limitations of human operators (limited ability to handle complexity, long reaction time, etc.).

There are multiple approaches that already employ techniques such as Deep Reinforcement Learning on analogous applications that endow drones with autonomous behavior [2] [1]. However, these approaches are currently limited because they: (i) do not consider key factors like smoke, which can have a remarkable disruptive effect on the monitoring task of UAVs, and (ii) are largely reactive and not equipped to anticipate how the fire will spread in the short term. This second limitation is particularly relevant because it may result in situations where drone behavior may either be too slow to react on time to avoid drone damage, or too conservative (and thus suboptimal to effectively monitor areas that are relatively close to the fire).

In this paper, we describe a work-in-progress that aims at overcoming such limitations by proposing an approach that allows: (i) defining a more realistic model of the wildfire propagation behavior that incorporates a smoke model; and (ii) a combination of machine learning techniques that will be used to anticipate the forest fire spread and proactively adapt the behavior of the UAV team to it.

2 Problem description

The core of our problem description is based on an analogous one for aircraft-class drones [2]. However, our work extends the environment description to consider smoke, and adapts the problem to the specific context of quadcopter UAVs.

Wildfire modeling A forest area is represented as a matrix of $N \times N$ cells, where each cell $c \in C$ incorporates two time-dependent variables that represent: (i) the amount of burnable fuel in the cell $F : C \to \mathbb{N}$; and (ii) whether the cell is burning ($B : C \to \{0, 1\}$). We assume a discrete notion of time, and for simplicity, we represent the fuel value of a variable associated with a cell c at time instant t as e.g., $F_t(c)$. When one time unit elapses, each cell can change in three ways:

- If cell c is burning and there is remaining fuel ($F_t(c) > 0 \wedge B_t(c) = 0$), fuel is decreased according to a burning rate parameter β (consumption speed).
- When $F_t(c) = 0$, the fire is extinguished in that cell ($B_{t+1}(c) = 0$).
- When there is fuel remaining and the cell is not burning ($F_t(c) > 0 \wedge B_t(c) = 0$), we define a probability $p_\iota : C \to [0, 1]$ of the cell getting ignited that is inversely proportional to the proximity to other burning cells.[1]

A concrete example of how wildfire propagates through a grid where $N = 25$ can be seen in Figure 1.

Wind and smoke modeling Wind is modeled as the bias of a cell igniting probability, depending on the direction (north, east, south, west) and strength. For instance, if the wind blows north, any cell (c') north of a burning cell c will see its ignite probability $p_\iota(c)$ increased according to a wind speed parameter μ, while cells south of c will see their ignite probability decreased by μ.

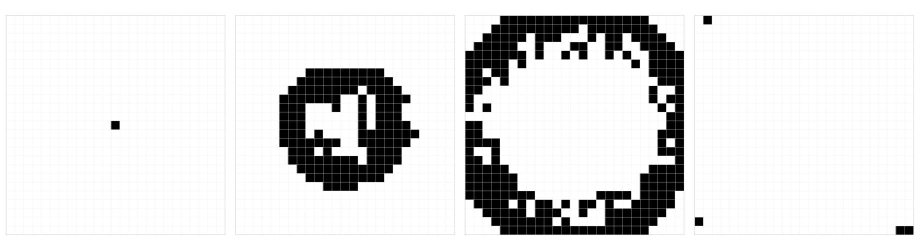

Fig. 1. Wildfire propagation in a 25x25 grid: (a) $t = 0$ (b) $t = 5$ (c) $t = 10$ (d) $t = 15$

Smoke is modeled analogously to wind, with three variables for each cell: (i) $S : C \to \{0, 1\}$ indicates whether the cell contains smoke or not, (ii) $H : C \to \mathbb{N}$ indicates smoke height, and (iii) $DC : C \to \mathbb{N}$ is a dispelling smoke counter.

[1] Readers interested in details can refer to [2].

Smoke appears only if there is fire in a cell. After smoke appears over a burning cell c, it gains height each time unit by an ascension rate ($H_{t+1}(c) = H_t(c) + \Phi$). When the cell stops burning, the smoke stops gaining height, and the dispelling counter starts to discount ($DC_{t+1}(c) = DC_t(c) - \gamma$ if $B_t(c) = 0 \wedge S_t(c) = 1$) according to a discount rate γ. When $DC(c) = 0$, the smoke in that cell disappears ($S(c) = 0$). Moreover, smoke can be spread over cells with or without fire. For a given cell c, there is a probability to get smoke based on the weighted multiplication of contiguous cells smoke level.

Unmanned Air Vehicles We assume that the UAV is in constant flight and can move at a constant speed measured in cells per time unit. The available actions to a UAV are changing altitude, moving north, east, south, and west.

The goal of a UAV is to maximize its covered forest area, i.e., to monitor as many cells on fire as possible. At the same time, the UAV should avoid the smoke (which hampers its ability to effectively monitor its surroundings), as well as getting too close to the fire to avoid damage. Globally, the goal for the UAV team is to maximize the overall monitored forest area.

3 Self-Adaptive Wildfire Monitoring

We envisage a solution to the problem described in Section 2 based on MAPE-K [**?**], considered to be one of the most successful approaches to engineering self-adaptive software that has at its core a set of models used to support reasoning at run time about *when* (analysis) and *how* (planning) to best adapt the system. For our solution, we intend to instantiate MAPE-K making use of components that implement both RNNs and DRL to inform the analysis and planning stages of the MAPE loop, respectively, as illustrated in Figure 3. The *managed system* layer (bottom) is the UAV's software, which includes components to control both sensors and actuators. The *managing system* layer (top) incorporates the various stages that endow the system with autonomous, self-adaptive behavior:

1. The monitor stage aggregates sensor information and provides it to the RNN, which predicts the next elements of a sequence of observations (e.g., about the state of the terrain, other drones) based on previous ones, providing to the analyze stage a prediction of how the fire will spread.
2. Based on the output provided by the RNN component, the analyze stage determines whether the current situation demands an action from the UAV (e.g., because it may be too close to the fire). If analyze determines that there is a need for adaptation, then the plan stage is triggered.
3. When the plan stage is triggered, it employs DRL for decision-making. Concretely, the DRL component receives as input a local observation of the area, and a prediction of how it will partially evolve in the next time units (obtained from the RNN component). The DRL generates a decision by calculating the weighted reward of the inputs, together with variables concerning about other UAVs state and generates a plan as output, that consists in moving the respective UAV to a new position in order to maximize the monitored fire area, which is passed on to the execute stage.

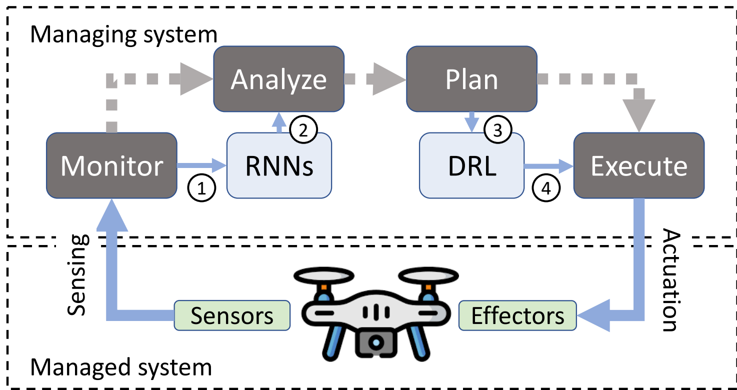

Fig. 2. An overview of our self-adaptive wildfire monitoring approach

4. Finally, the `execute` stage receives the plan and enacts its execution by co-ordinating the effectors embedded at the managed system layer.

4 Conclusions and Future Work

We have presented a work-in-pogress approach to build self-adaptive software for wildfire monitoring with UAVs that is supported by an ensemble of AI techniques aimed at anticipating the evolution of the environment to optimize UAV team operations. In future work, we plan on moving towards more realistic scenarios and refining our RNN and DRL architectures to make them more precise.

References

1. Jain, P., Coogan, S.C., Subramanian, S.G., Crowley, M., Taylor, S., Flannigan, M.D.: A review of machine learning applications in wildfire science and management. Environmental Reviews **28**, 478–505 (2020), `https://cdnsciencepub.com/doi/10.1139/er-2020-0019`
2. Julian, K.D., Kochenderfer, M.J.: Distributed wildfire surveillance with autonomous aircraft using deep reinforcement learning. Journal of Guidance, Control, and Dynamics **42**, 1768–1778 (10 2018), `https://arxiv.org/abs/1810.04244v1`
3. Ministerio para la transición ecológica y el reto demográfico: Avance informativo de incendios forestales del 1 de enero al 31 de diciembre de 2022. (2023), `https://www.miteco.gob.es/es/biodiversidad/temas/incendios-forestales/avanceinformativo31dediciembre2022_tcm30-560521.pdf`

Acknowledgements This work was partially funded by the Spanish Government (FEDER/Ministerio de Ciencia e Innovación–Agencia Estatal de Investigación) under projects TED2021-130523B-I00 and PID2021-125527NB-I00.