Design of a wildfire monitoring system using fleets of Unmanned Aerial Vehicles

Rafael Bailon-Ruiz

THÈSE

En vue de l'obtention du

DOCTORAT DE L'UNIVERSITÉ DE TOULOUSE

Délivré par : *l'Institut National des Sciences Appliquées de Toulouse (INSA de Toulouse)*

Présentée et soutenue le *24/09/2020* **par :**

Rafael BAILÓN RUIZ

Design of a wildfire monitoring system using fleets of Unmanned Aerial Vehicles

---

JURY

OUIDDAD LABBANI-IGBIDA	XLIM Limoges	Présidente du Jury
NOURY BOURAQADI	IMT Lille Douai	Examinateur
CHARLES LESIRE	ONERA Toulouse	Examinateur
LUIS MERINO	Universidad Pablo Olavide	Rapporteur
LUCILE ROSSI	Université de Corse	Rapporteure
SIMON LACROIX	LAAS-CNRS	Directeur de Thèse

École doctorale et spécialité :
EDSYS : Robotique et Informatique
Unité de Recherche :
Laboratoire d'Analyse et d'Architecture des Systèmes (LAAS-CNRS)
Directeur de Thèse :
Simon LACROIX
Rapporteurs :
Luis MERINO et *Lucile ROSSI*

Abstract

Wildfires, also known as forest or wildland fires, are uncontrolled vegetation fires occurring in rural areas that cause tremendous damage to the society, harming environment, property and people. The firefighting endeavor is a dull, dirty and dangerous job and as such, can greatly benefit from automation to reduce human exposure to hazards. Aerial remote sensing is a common technique to obtain precise information about a wildfire state so fire response teams can prepare countermeasures. This task, when performed with manned aerial vehicles, expose operators to high risks that can be eliminated by the use of autonomous vehicles. This thesis introduces a wildfire monitoring system based on fleets of unmanned aerial vehicles (UAVs) to provide firefighters with timely updated information about a wildland fire. We present an approach to plan trajectories for a fleet of fixed-wing UAVs to observe a wildfire evolving over time. Realistic models of the terrain, of the fire propagation process, and of the UAVs are exploited, together with a model of the wind, to predict wildfire spread and plan UAV motion. The approach tailors a generic Variable Neighborhood Search method to these models and the associated constraints. The execution of the planned monitoring mission provides wildfire maps that are transmitted to the fire response team and exploited by the planning algorithm to plan new observation trajectories. Algorithms and models are integrated within a software architecture allowing for execution under scenarios with different levels of realism, with real and simulated UAVs flying over a real or synthetic wildfire. Mixed-reality simulation results show the ability to plan observation trajectories for a small fleet of UAVs, and to update the plans when new information on the fire are incorporated in the fire model.

Keywords

Fleets of UAVs, Remote sensing, Wildfire monitoring, Multi-robot planning, Mixed-reality simulation.

Résumé

Les feux de forêt sont des incendies de végétation incontrôlés qui causent des dégâts importants à l'environnement, aux biens et aux personnes. Les actions de lutte contre de tels feux sont risqués et peuvent par conséquent bénéficier de techniques d'automatisation pour réduire l'exposition humaine. La télédétection aérienne est une technique qui permet d'obtenir des informations précises sur l'état d'un feu de forêt, afin que les équipes d'intervention puissent préparer des contremesures. Avec des véhicules aériens habités, elle expose les opérateurs à des risques élevés, qui peuvent être évités par l'utilisation de véhicules autonomes. Cette thèse présente un système de surveillance de feux de forêt basé sur des flottes de véhicules aériens sans pilote (UAV) afin de fournir aux pompiers des renseignements précis et à jour sur un feu de forêt. Nous présentons une approche pour planifier les trajectoires d'une flotte de drones à voilure fixe afin d'observer un feu de forêt évoluant au fil du temps. Des modèles réalistes du terrain, du processus de propagation du feu et des drones sont exploités, ainsi qu'un modèle du vent, pour prédire la propagation des feux de forêt et planifier le mouvement des drones. L'approche présentée adapte une méthode générique de recherche à voisinage variable (VNS) à ces modèles et les contraintes associées. L'exécution de la mission d'observation planifiée fournit des cartes des feux de forêt qui sont transmises à l'équipe d'intervention et exploitées par l'algorithme de planification pour déterminer de nouvelles trajectoires d'observation. Les algorithmes et les modèles sont intégrés dans une architecture logicielle permettant l'exécution dans des scénarios avec différents niveaux de réalisme, avec des drones réels et simulés survolant un feu de forêt réel ou synthétique. Les résultats de simulation mixte montrent la capacité de planifier les trajectoires d'observation d'une petite flotte de drones et de mettre à jour les plans lorsque de nouvelles informations sur l'incendie sont incorporées dans le modèle de propagation de feu.

Mots-clés

Flottes de drones, Télédétection, Surveillance de feux de forêt, Planification multirobot, Simulation à réalité mixte

Acknowledgments

First of all, I am very grateful to my supervisor Simon Lacroix for his mentoring and for the very valuable advice he has given to me during the last years, first as a master intern and then as a PhD student. My sincere gratitude to Arthur Bit-Monnot for his assistance at the beginning of the thesis to pave the path of my research.

I would like to thank the partners of the FireRS consortium for their work towards making the project possible and for their warm welcome during my multiple visits to Vigo and Porto. Special thanks to Maria Costa, Keila Lima and Paulo Dias of the Underwater Systems and Technology Laboratory of the University of Porto for their help to integrate my software with the UAV platform.

My warmest thanks to the all the members of RIS team that I had the opportunity to meet during my time at LAAS. Your friendliness is the source of an amazing work environment that encouraged me every day. I can not forget the much-needed relaxing atmosphere of the engaging coffee break conversations and the short after-lunch card playing sessions.

To my friends and family, especially to my father Rafael and my brother Miguel Ángel, always proud of my accomplishments.

Finally, my deepest gratitude goes to Estelle for her unconditional love and support. Thank you for being by my side in the PhD journey.

Contents

I Context **1**

1 Introduction **3**

1.1 Overview 3
1.2 Contributions 4
1.3 Outline 5

2 Review of UAV-based wildfire monitoring systems **7**

2.1 Brief overview of UAV technology development 7
2.2 Relevant publications about UAVs and wildfires 9
2.2.1 Single UAV systems 11
2.2.2 Multiple UAV systems 13
2.3 Autonomy level classification scheme 15
2.3.1 Main metrics 16
2.3.2 Secondary concerns 18
2.4 Discussion 20
2.5 The FireRS system 24
2.5.1 Objectives 24
2.5.2 Operational view 24
2.5.3 Approach 25

II Models **27**

3 Wildfire models **29**

3.1 Introduction 29
3.1.1 Parts of a wildfire 30
3.1.2 Environmental factors of wildfire propagation 31
3.2 Modeling wildfire behavior 34
3.2.1 Limitations 35
3.2.2 Wildfire situation assessment 36
3.3 Design of a wildfire simulator software 37
3.3.1 Wildfire model integration 37
3.3.2 Input data 41
3.3.3 Building the wildfire map 44
3.3.4 Illustrations 46
3.4 Conclusion 46

4 UAV models **51**
4.1 UAV motion model 51
4.1.1 Computation of flat flight trajectories without wind 52
4.1.2 Computation of flat flight trajectories under constant wind . . 53
4.2 Perception model 55
4.2.1 Assessing visible cells 55
4.2.2 Assessing the utility of perceiving a given cell 57
4.3 Conclusion 59

III Algorithms **61**

5 Situation assessment: Update of the wildfire status belief **63**
5.1 Introduction 63
5.2 Fire mapping from aerial infrared imagery 65
5.2.1 Detection 66
5.2.2 Mapping 66
5.2.3 Fusion of local wildfire maps 67
5.3 Estimation of the current wildfire situation 68
5.3.1 Fusion of observed and predicted wildfire maps using an image warping algorithm 69
5.3.2 Illustrations and analysis 70
5.4 Conclusion 71

6 Planning algorithms for wildfire monitoring **77**
6.1 Characteristics of the wildfire monitoring problem 77
6.2 Related work 78
6.2.1 Existing approaches solving the Orienteering Problem 79
6.2.2 Aircraft motion models for Vehicle Routing Problems 79
6.3 Problem formulation 80
6.4 Variable Neighborhood Search planner 81
6.4.1 Basic VNS 81
6.4.2 Improved algorithm 83
6.4.3 Definition of the neighborhoods 84
6.5 Illustrations and analysis 89
6.5.1 VNS implementation details 89
6.5.2 Demonstration of the observation planner over selected wildfire scenarios 91
6.6 Conclusion 96

IV Integration **99**

7 Wildfire monitoring system integration **101**
7.1 System architecture 101

7.1.1 Overview of the FireRS SAOP system architecture 102
7.1.2 UAV control software 104
7.1.3 Communication infrastructure 104
7.1.4 Subsystems 108
7.2 Mixed-reality simulation of wildfire monitoring 111
7.2.1 Implementation of a mixed-reality wildfire mapping framework 112
7.3 Field tests and results 114
7.3.1 Test scenarios 114
7.3.2 Demonstration campaigns 117
7.4 Conclusion 120

8 Conclusion 125
8.1 Summary 125
8.2 Discussion and future work 125
8.2.1 Computation of 3D flight trajectories 126
8.2.2 Improvements to wildfire situation assessment 128
8.2.3 Planning improvements for long-lasting UAV wildfire monitoring missions 128
8.2.4 System supervision 130
8.2.5 Robust integration of independent simulators 130

Bibliography 131

List of Figures

2.1 Number of *UAV* related publications by year 8
2.2 Number of *UAV and wildfire* related publications by year 9
2.3 The FiRE project system architecture 11
2.4 Aerial view of a wildfire from different angles 14
2.5 Multiple sensor and communication schemes for wildfire monitoring 16
2.6 Examination of the wildfire remote monitoring systems reviewed in this survey 23
2.7 The FireRS system global architecture 25

3.1 The fire tetrahedron 30
3.2 Satellite image of a wind driven wildfire 31
3.3 Parts of a wind-driven wildfire 32
3.4 Illustration of the heat transfer mechanisms of an uphill wind-driven surface fire 34
3.5 Illustration of wildfire propagation, wind field, elevation and fuel maps 38
3.6 Illustration of fuel types 40
3.7 Illustration of the wildfire perimeter shape 40
3.8 Digital elevation map of a $25\,\mathrm{km} \times 25\,\mathrm{km}$ area in the french Hautes-Pyrénées department 42
3.9 Surface wind maps obtained with WindNinja using the terrain depicted in Figure 3.8 43
3.10 Land cover map of the region described in Figure 3.8 45
3.11 Wildfire propagation map in a $5\,\mathrm{km}$ by $5\,\mathrm{km}$ area 46
3.12 Illustration of a wildfire propagation from one ignition point 47
3.13 Illustration of the wildfire propagation of Figure 3.12 with different wind conditions. 47
3.14 Illustration of the propagation of concomitant wildfires. 48
3.15 Illustration of the propagation of concomitant wildfires. 48

4.1 UAV reference axes and rotation angles. 52
4.2 The six possible optimal standard Dubins path configurations linking two oriented points. 53
4.3 Trochoidal path performed by a UAV turning at a constant rate under the effect of a constant wind field. 54
4.4 Depiction of the effect of wind over Dubins paths 55
4.5 Illustration of the UAV perception model 56
4.6 Illustration of the perception model outputs resulting from a UAV trajectory 57

5.1 Diagram of the proposed situation assessment process. 64

5.2 Mapping of a 2D point to a 3D earth location 66
5.3 Mapping of the fire detected in a picture, taken from a UAV, to a coarser raster map. A cell is considered on fire if it is covered by a majority of pixels on fire. 67
5.4 Deformation of a checkerboard pattern image using thin-plate image warping. 69
5.5 Situation assessment scenario 1 72
5.6 Situation assessment scenario 2 73
5.7 Situation assessment scenario 3 74
5.8 Situation assessment scenario 4 75
5.9 Situation assessment scenario 5 76

6.1 The waypoint insertion process 88
6.2 Illustration of the local path optimization process 89
6.3 Illustration of the planning test situation 1: One wildfire 91
6.4 Illustration of the planning test situation 2: One large wildfire 92
6.5 Illustration of the planning test situation 3: Three wildfires 92
6.6 Illustration of plan to observe the wildfire of situation 1 with one UAV 93
6.7 Illustration of plan to observe the wildfire of situation 1 with three UAVs 93
6.8 Illustration of plan to observe the wildfire of situation 2 with one UAV 94
6.9 Illustration of plan to observe the wildfire of situation 2 with three UAVs 94
6.10 Illustration of plan to observe the wildfires of situation 3 with one UAV 95
6.11 Illustration of plan to observe the wildfires of situation 3 with three UAVs 95
6.12 Illustration of an observation plan with strong wind blowing from the south 97
6.13 Illustration of an observation plan with strong wind blowing from the west. 97

7.1 FireRS SAOP wildfire monitoring concept architecture 102
7.2 FireRS SAOP system architecture. Orange boxes represent data messages, red boxes are commands, and blue ovals are the main processes.103
7.3 Illustration of the Neptus command and control software while operating autonomously a UAV 105
7.4 FireRS SAOP integration of the fleet of UAVs and satellite external network infrastructure 106
7.5 FireRS SAOP implementation details. Similar color and shape conventions as in Figure 7.2 apply: orange boxes are ROS topics and blue ovals, ROS nodes. Ground control station components are depicted in purple. 107
7.6 Screenshot of the FireRS SAOP human-machine interface 110
7.7 Mixed-reality wildfire mapping architecture 113

7.8 Illustration of Blender showing a Morse environment of a real rural area digital elevation map. 114
7.9 Illustration of Blender showing a synthetic wildfire 115
7.10 Diagram of the components involved in the INIT scenario 116
7.11 Diagram of the components involved in the RUN scenario 117
7.12 Diagram of the components involved in the MONITOR scenario . . . 118
7.13 Demonstration sites in Spain 118
7.14 Still image of a real fire from the infrared video feed provided by the wildfire ground sensor located in the fire emergency training center . 119
7.15 Screenshot of the satellite communication system interface displaying a wildfire alarm and the SAOP HMI showing the corresponding predicted wildfire map 120
7.16 Details of the demonstration flying field near the city of Porto (Portugal) 121
7.17 Scenario RUN situation during the Porto demonstration flight. 122
7.18 Screenshot of the SAOP HMI after running the MONITOR scenario in Portugal 122
7.19 Observed wildfire map of the same fire depicted in Figure 7.18 30 minutes later 123

8.1 Illustration of the three types of Dubins 3D paths 127

List of Tables

2.1 Synthesis of the reviewed UAV-based wildfire remote sensing systems. 21
2.1 Synthesis of the reviewed UAV-based wildfire remote sensing systems. 22

5.1 Wind conditions of the situation assessment scenarios 71

List of Algorithms

1 Basic Variable Neighborhood Search algorithm 82
2 The shuffling algorithm pseudo-code 84
3 Pseudo-code of the proposed Variable Neighborhood Search (VNS) algorithm 85
4 Pseudo-code of the *projectFF* function 86

Part I

Context

CHAPTER 1

Introduction

1.1 Overview

Wildfires, also known as forest or wildland fires, are uncontrolled fires that occur on rural areas and whose suppression requires large amounts of human and technical efforts. Once an ignition starts, it is essential to declare it as soon as possible so the outbreak can be rapidly controlled and suppressed. However, wildland fire fighters usually lack precise timely updated information about the situation because its location is remote or difficult to access. When the outbreak is not controlled on time and the wildfire continues to grow, its danger increases and the suppression costs boom. Currently, data obtained on an active wildfire perimeter is either imprecise, when acquired from a far safe distance; late, when satellite imagery is used; or risky and expensive, if aerial means are used.

The popularity of Unmanned Aerial Vehicles (UAVs) for remote sensing applications has surged in recent years. Wildfire propagation monitoring is an activity that could greatly benefit from this technology as a way to overcome the limitations of traditional sensing techniques. A fleet of autonomous fixed-wing UAVs equipped with thermal infrared cameras can be deployed expeditiously at low cost and risk for human fire fighters. But as of today, many of those systems consists of single remote piloted aircraft, with little to no autonomy, and whose operation range is reduced.

The goal of this thesis is to develop a wildfire monitoring system using UAVs with an aim to provide wildland firefighters with complete up-to-date maps of an ongoing wildfire event. Achieving such a system raises numerous challenges: wildfires are dynamic and uncertain, the time scale and distances involved are large, and UAV motion and perception are constrained. Planning wildfire observations for multiple vehicles require fine-tuned algorithms, that handle the trade-offs imposed by these restrictions, that should be flexible enough to accept the revision of the plans after the incorporation of new information.

The challenges of wildfire observation planning also have an impact on fire data processing. As measurements of every location and time are not available – the opposite would require many more UAVs to monitor every piece of land simultaneously – situation assessment algorithms are needed to fuse observations coming from different sources and to estimate the missing portions of wildfire perimeter knowledge.

The integration of the wildfire observation planning and situation assessment algorithms with the UAV control platform and user interaction depend on a comprehensive software architecture that orchestrates the data and control flows. Testing this architecture under real conditions faces an important challenge: wildfires are extraordinary events that cannot be replicated in a laboratory. Realistic wildfire simulations can be valid substitutes to the real ones for testing purposes during preliminary system design. Mixed-reality frameworks offer improved testing capabilities to combine synthetic wildfires with real or simulated software- and hardware-in-the-loop UAVs at the different stages of product design.

1.2 Contributions

The main contributions of this thesis reside in the analysis of the problem of wildfire monitoring using fleets of autonomous UAVs and the design and implementation of the principal components of wildfire remote surveillance system:

- Study the state of the art as an overview of the past, current and future trends of wildfire remote sensing from an autonomy perspective.
- Propose and implement a simple strategy of wildfire situation assessment to produce estimations of the current and future wildfire situation.
- Formalize the problem of wildfire observation with multiple UAVs and develop a planning algorithm that exploits existing fixed-wing UAV, wildfire propagation, geographic and wind models.
- Design a software architecture that integrates the situation assessment and observation planning algorithms and allows testing them in a mixed real-synthetic environment.

The work presented in this thesis has been carried out within the scope of the Fire-RS project financed by the European Regional Development Fund through the Interreg V Sudoe program[1] in collaboration with researchers of the University of Vigo, Spain, and the University of Porto, Portugal. In particular, the design and implementation of the interface between the wildfire monitoring architecture and the UAV control system has been developed with the Underwater Systems and Technology Laboratory of the University of Porto[2]. Preliminary work on the observation planning algorithm has been done in cooperation with Arthur Bit-Monnot [Bit-Monnot 2018].

Scientific production Parts of this thesis have been published in the following international conferences:

[Bit-Monnot 2018] Arthur Bit-Monnot, **Rafael Bailon-Ruiz** and Simon Lacroix. A Local Search Approach to Observation Planning with Multiple UAVs.

[1] `https://www.interreg-sudoe.eu`
[2] `https://www.lsts.pt`

In Twenty-Eighth International Conference on Automated Planning and Scheduling, June 2018.

[Bailon-Ruiz 2018] **Rafael Bailon-Ruiz**, Arthur Bit-Monnot and Simon Lacroix. Planning to Monitor Wildfires with a Fleet of UAVs. In 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pages 4729–4734, Madrid, Spain, October 2018. IEEE

[Pérez-Lissi 2018] Franco Pérez-Lissi, Fernando Aguado-Agelet, Antón Vázquez, Pablo Yañez, Pablo Izquierdo, Simon Lacroix, **Rafael Bailon-Ruiz**, Joao Tasso, Andre Guerra and Maria Costa. FIRE-RS: Integrating Land Sensors, Cubesat Communications, Unmanned Aerial Vehicles and a Situation Assessment Software for Wildland Fire Characterization and Mapping. In 69th International Astronautical Congress, Bremen, Germany, October 2018.

[Bailon-Ruiz 2020] **Rafael Bailon-Ruiz** and Simon Lacroix. Wildfire Remote Sensing with UAVs: A Review from the Autonomy Point of View. In 2020 International Conference on Unmanned Aircraft Systems (ICUAS), Athens, Greece, September 2020. IEEE.

And the following paper is in preparation:

[Bailon-Ruiz 2020a] **Rafael Bailon-Ruiz**, Arthur Bit-Monnot and Simon Lacroix. Real-Time Wildfire Monitoring using Fleets of uavs: Models, Algorithms and Architecture. *(in preparation)*

1.3 Outline

This thesis is divided into four parts and organized in a total of eight chapters.

Part I introduces the reader to the subject of wildfire remote sensing with fleets of UAVs. It encompasses two chapters:

- **Chapter 1** (this chapter) gives a brief overview of the thesis subject, the outline and enumerates the technical and scientific contributions.
- **Chapter 2** is a review of wildfire remote sensing literature and an analysis from the perspective of system autonomy.

Part II characterizes the principal models used within the proposed wildfire monitoring system. It contains two chapters:

- **Chapter 3** provides a summary and a global analysis of existing wildfire propagation models and describes the integration of wildfire, geographic and weather models needed across the situation assessment and observation planning algorithms.

Chapter 4 introduces the various models used for observation planning: fixed-wing UAV motion, perception models, and an observation plan utility model.

Part III introduces wildfire situation assessment and observation planning algorithms:

Chapter 5 describes algorithms serving the purpose of fusion of fire observations and estimation of current and future wildfire maps using the model described in chapter 3.

Chapter 6 analyses the challenges of planning wildfire observations and introduces a planning algorithm that exploits the wildfire maps produced by the algorithms of chapter 5 and the models of chapter 4.

Part IV depicts the integration of the software introduced in the previous chapters:

Chapter 7 explain the software integration efforts and the architecture that integrates the models and algorithms of Part II and Part III. Then, it describes the design of a hybrid real-simulated simulation framework and presents the test campaigns performed with the wildfire monitoring system.

Chapter 8 concludes this thesis with comments on the work done and prospective developments.

CHAPTER 2

Review of UAV-based wildfire monitoring systems

This chapter analyses the state of the art on wildfire remote sensing using UAVs, an application context that has now gained significant interest. It reviews a selection of relevant publications, and introduces a classification scheme to synthesize them from an autonomy perspective. Three metrics are introduced: situation awareness, decisional ability, and collaboration ability. A discussion about the current state and the outlook of UAV systems for wildfire observation concludes the chapter.

2.1 Brief overview of UAV technology development

Unmanned Aerial Vehicles (UAVs), also known as Unmanned Aerial Systems (UAS) or Drones, are uninhabited flying machines remotely piloted by a human, another machine, or by themselves.

Designs of UAVs have been imagined since the ancient era, with a text of 425 B.C. describing a mechanical bird created by Archytas the Tarantine that was reported to fly about 200m [Valavanis 2007]. In the 15th century, Leonardo Da Vinci designed some aircraft that retains some of the ideas of today's helicopters even though it was not able to fly at the time. During the following centuries, multiple engineers tried to fly their own designs with little to no success due to immature engine, aerodynamic and control technologies.

Linked to the evolution of manned aviation, numerous aircraft designs appeared in the first half of the 20th century, but it was not until its last decades when UAVs took their place in aviation, first as a military technology. Improved computing power and miniaturization of electronic devices, better and smaller communication equipment and sensors made their operation for reconnaissance and intelligence missions very valuable.

With the beginning of the 21st century came the transition to civil UAVs. Discontinuation of GPS *Selected Availability* (degradation of GPS signal quality for civil use) in 2000 provided precise positioning anywhere on earth to everyone. Further miniaturization in electronics and the advent of MEMS technologies resulted in smaller, less power hungry Central Processing Units and lighter Inertial Measurement Units. Combined with improved energy storage density of batteries, UAVs where able to continue their path to autonomy.

At the heart of UAVs is the *autopilot*: the software that controls the UAV flight, including *guidance*, *navigation* and *control* (*GNC*), and communications among oth-

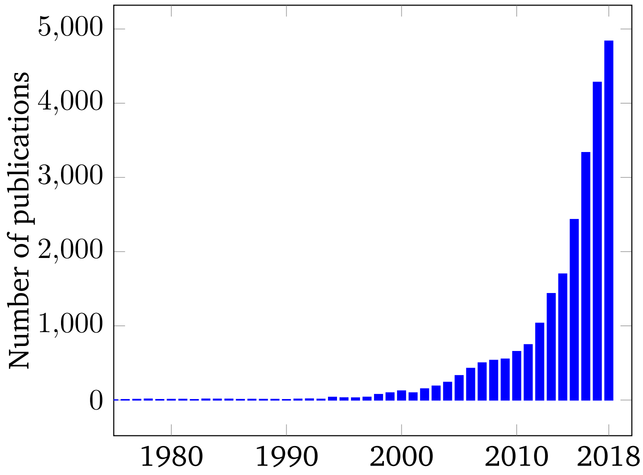


Figure 2.1: Number of *UAV* related publications by year.
Obtained using the search query *TS=((UAV$ OR drone$ OR "unmanned aerial" OR "unmanned aircraft" OR RPAS OR "uninhabited aerial" OR "uninhabited aircraft") NOT bee$ NOT workerbee$ NOT apis-mellifera NOT HIV)* in the *Web Of Science* database.

ers. During the last 15 years many autopilots have appeared thanks to the involvement of research institutions and developer communities following the free open-source software movement trail. Some examples are the Paparazzi autopilot (2003)[1] by the French Civil Aviation School, the LSTS toolchain (2004)[2] by the University of Porto in Portugal, ArduPilot (2009)[3] originally targeting the Arduino microcontroller kit, and the PX4 autopilot (2012)[4] from the Swiss Federal Institute of Technology.

The interest on UAVs in the scientific community has risen in the last decades, as these vehicles have been made cheaper and more capable. A search for the keywords *drone* and its synonyms, filtering unrelated keywords, in the Web of Science database[5], covering thousands of journals, conference proceedings and other remarkable sources, returns a total of 24084 publications for the period 1975–2018. The distribution of these records by year (Figure 2.1) shows an accelerating growth since 2010 in the number of publications, with the total accumulated publication count increasing exponentially as a consequence. As observed by D.J. De Solla Price [Price 1963], father of modern scientometrics, this is a clear symptom of an emerging research topic.

The most developed application of UAV technology is remote sensing, primarily using image sensors, and more recently LIDARs, in a variety of application contexts: terrain mapping for mining industry, agriculture and flood risk assessment; monitoring and surveillance of long linear structures such as power lines, roads

[1] `https://wiki.paparazziuav.org/wiki/Main_Page`
[2] `https://lsts.fe.up.pt/toolchain`
[3] `https://ardupilot.org/`
[4] `https://px4.io/`
[5] `https://clarivate.com/webofsciencegroup/solutions/web-of-science/`

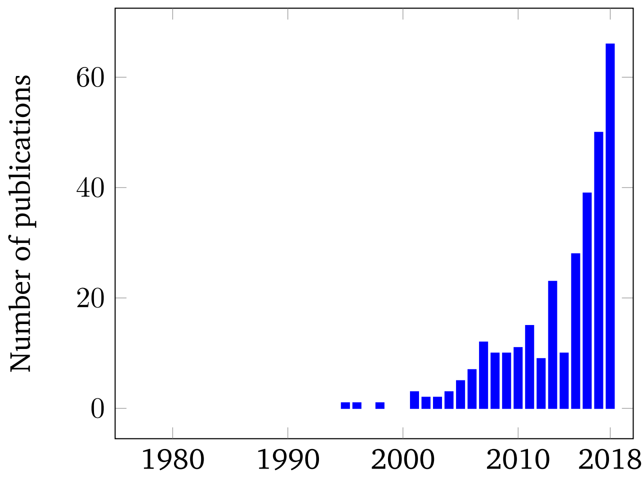


Figure 2.2: Number of *UAV and wildfire* related publications by year.
Obtained using the search query *TS=((fire OR wildland NEAR/1 fire OR "forest fire" OR wildfire$) AND ((UAV$ OR drone$ OR "unmanned aerial" OR "unmanned aircraft" OR RPAS OR "uninhabited aerial" OR "uninhabited aircraft") NOT(ant$ OR bee$ OR workerbee$ OR apis-mellifera OR HIV OR gun$ OR weapon$ OR "battle field" OR gunfire$)))* in the *Web of Science* database.

and railway infrastructures; large structure inspection and wildlife monitoring. In the context of forestry, drones have been proved useful for canopy mapping, forest management and wildfire tracking [Tang 2015]. The development of such applications is fostered by specific benefits of UAVs with respect to aircraft and satellite solutions [Torresan 2017]: low investment and operational costs, and the ability to provide timely and high-resolution data collection. The main downside is limited flight endurance, and hence a reduced extent over which data can be collected. However, this is the usual trade-off to which data acquisition systems are prone, that is data resolution versus spatial extent.

2.2 Relevant publications about UAVs and wildfires

The wildfire community is increasingly adopting Unmanned Aerial Systems as this technology shows promising applications in the field. As a consequence, the consideration of UAVs for wildfire remote sensing has risen in the recent years following the global trend already mentioned. A search query in the *Web of Science* including publications with contents related to UAVs and wildfires returns 308 results for the 1975 to 2018 range. These results, depicted in Figure 2.2, show a trend in the increase of the number of records similar to the one seen on UAVs, but naturally in much smaller numbers.

To our knowledge, only two surveys about wildfire remote sensing systems using UAVs have been published: [Yuan 2015] and [Twidwell 2016]. [Yuan 2015] focus on the analysis of the sensing hardware and algorithms, including a detailed classification of image sensor characteristics and fire detection algorithms. [Twidwell 2016] is an overview of the applicability of UAV technology into fire management operations in which the authors also provide a clear synthesis of the institutional and legal state of the art in the USA. Because system architecture and operational autonomy of the vehicles are secondary concerns in both reviews, we deem interesting to discuss the usage of UAVs in wildfire remote sensing from a robotic point of view: studying the interactions between vehicles, control operators and the environment. In other words, how systems are designed to autonomously act with little or without direct human involvement to gather information on wildfires.

As reviewed by [Twidwell 2016], the expansion of UAV applications in the fire domain area mainly pertains to the domain of wildfire remote sensing. Manned aerial wildfire monitoring is costly and very risky, specially when dealing with uncontrolled fires. Hence, transferring the aircraft operator from the air to the ground improves the cost-effectiveness and efficiency of wildfire fighting efforts [Christensen 2015], freeing resources for other firefighting-related duties. Other tasks beyond remote sensing can be achieved by UAVs, like the aerial ignition of prescribed fires [Beachly 2018], or even firefighting [Haksar 2018], the latter being yet to be developed in operational contexts, as it requires carrying huge quantities of water and fire retardant.

The most common mission in the wildfire remote sensing domain is fire mapping, which produces a map of an area highlighting the locations on fire at a particular time from geo-referenced aerial imagery. Additionally, it is possible to process the fire maps in order to determine the current fire perimeter and to provide an estimation of its position in unobserved areas. When mapping is performed in a continuous manner, for instance to track a fire perimeter to provide regular updates of the fire map, it is referred to as *monitoring*.

UAVs are able to generate rich and precise fire data with high resolution cameras, that can be used to characterize the fire geometry. Remote 3D reconstruction of a wildfire can give a lot of information to the firefighters, who can safely assess the fire severity in a particular location. Besides, collected data help researchers to better understand fire propagation. The work of [Martínez-de Dios 2011] and [Ciullo 2018] are examples of 3D flame reconstruction algorithms from aerial stereo footage. Aerial thermal infrared imaging can also be used in automated wildfire monitoring. In [Valero 2017], the proposed system is able to track wildfire perimeters from images acquired by a UAV and use this information to improve the parameters of a wildfire propagation simulator. Such wildfire prognosis capability could be integrated in an automated fire decision support tool, but means of this kind still have some way to go before their use in real operational situations.

The following subsections summarize and comment the most relevant publications related to wildfire observation systems using UAV technology. Those have been selected for the novelty of the approaches and the relative importance of the citation count. Then, a discussion about of the state of the art is provided with a focus on robotic intelligence and autonomy.

Publications are put into two distinct categories, chosen based on the number vehicles involved in the design: 1. Systems consisting on a *Single UAV*, and 2. sys-

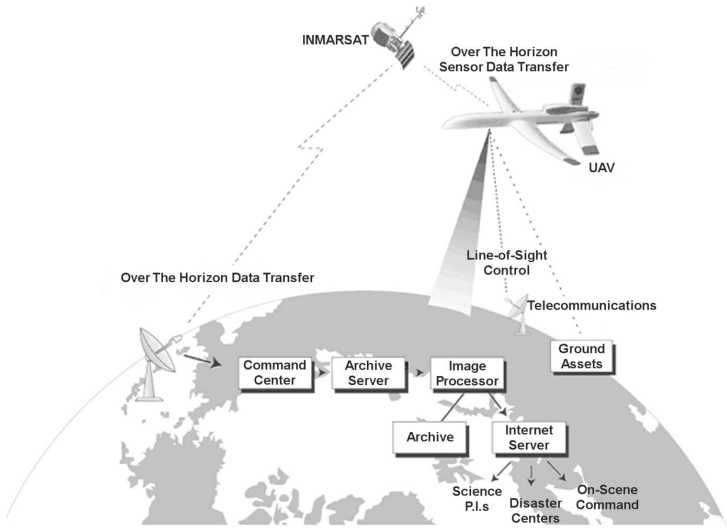


Figure 2.3: The FiRE project system architecture. A UAV captures thermal images of an ongoing fire and sends them to a command center through satellite communication. Images are geo-rectified as they arrive, then delivered to hazard response teams. *Excerpt from [Ambrosia 2003]*

tems built upon *Multiple UAVs*. This categorization reflects a leading trend in the state of the art with systems based on fleets of UAVs becoming more common as UAV technology becomes more powerful.

2.2.1 Single UAV systems

Preliminary work on unmanned aircraft technologies for wildfire remote sensing started in the early 2000's. This period is characterized by the usage of remotely piloted High Altitude and Long Endurance UAVs (HALE UAVs) exploited by research agencies as a complement to existing satellite monitoring systems. HALE UAVs feature great capabilities: they can fly for hours at high altitude and carry significantly heavy payloads, buy they are expensive systems and do not provide more precise data than satellites do. Small UAVs fly much closer to the ground potentially being able to sense more detailed information about wildfires.

The objective of the *FiRE* project [Ambrosia 2003] is to demonstrate the use of a remotely operated UAV equipped with a thermal scanner for wildfire mapping. Images are transmitted to the ground station via a satellite link and are then geo-rectified to produce a fire map in real time. The system was tested in a controlled burn site in 2001 using an ALTUS II UAV, producing 5 geo-rectified images with a $2.5m$ spatial resolution during a one hour flight. The architecture of the *FiRE* system, depicted in Figure 2.3, is probably the first instance of a complete wildfire monitoring system, serving as the foundation of more capable ones to come in the following years.

People from the same laboratory reported in [Ambrosia 2011] about new fire imaging missions flown by NASA and the US Forest Service with a HALE UAV between 2006 and 2010. This time, the UAV included a hyperspectral camera and sufficient computing power to process sensor data on-board. A fire hot-spot algorithm running on the UAV on-board CPU was able to detect burning areas by applying a threshold to selected infrared bands. With the help of a digital elevation model, image regions corresponding to fire were projected over the ground level to obtain a geo-rectified fire map. Finally, the fire map was sent in real time to a Wildfire Collaborative Decision Environment that integrated multiple geospatial sources allowing real-time collaborative manipulation of the information.

OSIRIS [Lewyckyj 2007] is a European project whose objective is to develop a High Altitude Platform for wildfire monitoring using a solar-powered HALE UAV. This UAV is able to follow a predefined flight plan that can be updated at any time from a Ground Control Station. While achieving the mission, the aircraft is able to capture high-resolution images that are transmitted in real time to the ground station through a satellite link. Then, the raw images can be forwarded to a Central Data Processing Center for additional processing, that archives all the georeferenced imagery that has been produced and allows the final users to consult the information. No automated assessment is performed on the images to detect fire spots or propagation perimeters. We could not assess whether this work has been validated in realistic conditions or not.

[Esposito 2007] describes the results of a project devoted to wildfire monitoring with a remote sensing system based on two platforms: a small UAV and a two-seat airplane. Different sensing instruments were available including thermal, multispectral and hyperspectral cameras.

The work of [Restas 2006] assesses the interest of using UAVs to observe wildfires from the point of view of a firefighting team. Once a fire is declared, the first task of the response crew is the reconnaissance of the situation. Given that the damage generated by a wildfire depends greatly on the response time, the earlier firemen have an understanding on the severity of the fire, the earliest they can apply the best countermeasures. Hence, air reconnaissance with UAVs being operated directly by the fire fighting crew seems for the author the most effective way to get early information about a starting wildfire. When the fighting efforts are concentrated on a specific area, UAVs can be used as a decision support tool for the least active portions of the front. After doing a simple fire reconnaissance test with a regular commercial UAV, the author concludes with a roadmap for three wildfire monitoring designs that firefighting crews would benefit from. Its last milestone consists of fleets of autonomous UAVs including blimps, helicopters and fixed-wing aircraft.

An autonomous wildfire monitoring system is depicted in [Skeele 2016], whose objective is to track a set of hot-spots as fast as possible. Given a realistic simulation of the expected evolution of a wildfire, a hot-spot is defined by the center of a cluster of locations where the fire is the fastest growing. The authors introduce a greedy algorithm that guides the UAV towards these hot-spots. Then, the algorithm is evaluated against a baseline strategy consisting in circling the current fire contour. A mixed-reality experiment, with a real UAV and a simulated fire has been performed to test the proposed algorithm.

2.2.2 Multiple UAV systems

The use of multiple UAVs reveals new forms of operation. The scale of a wildfire is often too large to be encompassed by a single UAV, which fleets of UAVs can cope with. But systems built upon multiple UAVs also imply new design challenges, as vehicles must communicate and collaborate in some way to exploit the full potential of the fleet. Hence, some task allocation or distributed control algorithms are necessary to operate the fleet.

The interest in systems based on multiple UAVs is supported by a recent publication [Moulianitis 2019] that evaluates different wildfire remote sensing schemes, combining land sensors, fixed-wing and rotary-wing UAVs, and satellites. The main interest of this work is the analysis of different sensing strategies regarding how they fit into the fulfillment of typical missions, *i.e.* fast and reliable wildfire detection and robust monitoring. The authors retain three unique strategies: the first one using satellite and land sensor imagery, the second one built upon fleets of drones and the third one combining fleets of UAVs with ground cameras. UAV-based systems are found to be the most useful configuration overall with respect to their criteria.

To the extent of our knowledge, the first publication devising the application of fleets of fixed-wing UAVs as a wildfire remote sensing platform is [Dovis 2001], in which the authors conceive a remote sensing architecture using UAVs as a foreseeable alternative to satellites. The fleet, flying at stratospheric level, carries optical payloads and has on-board processing capabilities. The work presented in this article is more centered into the design of the optical payload, but the introduction of the idea of a UAV fleet platform with on-board processing for autonomous detection and tracking of ground phenomena was innovative at that time.

The authors of [Merino 2005] propose a fire spot detection system in the context of the COMETS project. This system included for the first time a fleet of heterogeneous UAVs composed of three vehicles of two different kinds and with varying levels of autonomy: autonomous and remote controlled helicopters, and a remote piloted blimp. The fire spotting algorithm relies on the collaboration between UAVs featuring IR cameras and other UAVs carrying visible cameras to increase the detection probability. Once a fire spot is confirmed, the assessment is completed by observing it from different points of view. This system has been tested on the field and authors provide an extensive report on the results of the experiment. More recent publications from the same authors introduce improved results on fire contour extraction [Merino 2012] and 3D fire shape estimation from multiple ground and aerial views [Martínez-de Dios 2011].

Other innovative work using fleets of UAVs is introduced in [Casbeer 2006]. This publication depicts a collaborative wildfire monitoring system using a fleet of

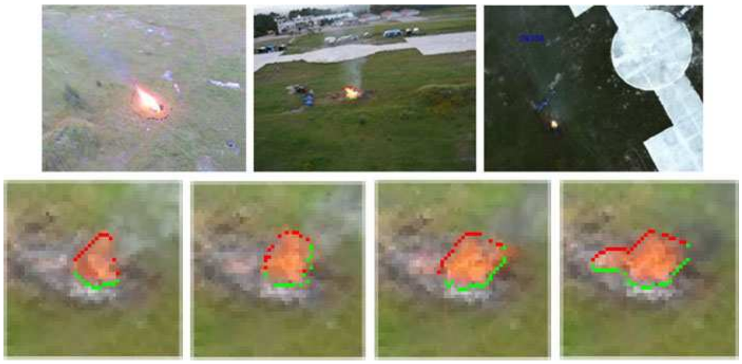


Figure 2.4: Aerial view of a wildfire from different angles *(top)*. Algorithms are able to locate the position of the fire if a digital elevation map is provided *(bottom)*. *Excerpt from [Merino 2012]*

LASE (Low Altitude Short Endurance) fixed-wing UAVs. The monitoring framework relies on a decentralized algorithm that makes each UAV track the front of a round shape wildfire to measure the total length of the perimeter. The UAVs fly in opposite directions and every time a vehicle meets another one, they exchange information about the length of the perimeter they have already tracked. Given sufficient time, the fleet agrees on the front length, equally distributing the portion of the front being monitored by every vehicle. The interest of the proposed system resides in its ability to work in limited communication scenarios, with information being sent from one agent to another, reaching at the end the ground station. This time, the algorithm was tested with the help of a wildfire simulator and a 6 degree of freedom simulator for the UAVs.

[Rabinovich 2018] describes a monitoring architecture to estimate the perimeter of a wildfire using a fleet of UAVs. Their model defines the fire front as a set of discrete control points that move outwards the center of the fire. Then, the border is observed by a fleet of UAVs equipped with a binary sensor capable of detecting whether a UAV is located over the wildfire or not. Observations are fed into a tailored Kalman filter that estimates the location of the perimeter control points, and associates to each of them an uncertainty value. A planning algorithm creates a plan for fleets of UAVs to observe the control points prioritizing those whose location is more uncertain.

[von Wahl 2010] proposes an early fire detection platform with ground sensors and a fleet of UAVs. The suggested scheme advocates for the use of small quadrotor drones for early fire verification and a blimp equipped with a smoke detector and a radiometer for hot-spot monitoring.

A novel swarm control algorithm for wildfire search and tracking is proposed in [Howden 2013]. This algorithm for fleets of microUAVs is based on a dynamic pheromone map, which combined with a particular heuristic, makes the swarm search for new fire spots while tracking existing ones in a distributed manner. Each UAV carries an infrared camera capable of detecting and geo-locating a fire. Periodically, pheromone maps are shared between agents with the objective of collaboratively building a complete fire map. This wildfire tracking algorithm has been tested in simulation over a synthetic oval-shaped fire.

[Pham 2017] proposes a distributed control framework for a fleet of rotary-wing UAVs that spatially distributes the vehicles to observe a wildfire. The algorithm tries to maximize the fire coverage, moving the vehicles to the most advantageous positions while the fire keeps spreading.

[Haksar 2018] applies a deep reinforcement learning strategy for a fleet of firefighting UAVs. The purpose of the proposed algorithm is to control the propagation of a wildfire by commanding the UAVs to, first, detect trees close to ignition and second, drop some retardant to stop fire spread.

A wildfire rate of spread estimation algorithm using data obtained by a fleet of rotary-wing UAVs is presented in [Lin 2015], and further developed in [Lin 2018] and [Lin 2019]. The estimation exploits a Kalman filter that combines past knowledge about the fire perimeter with measurements to predict the current contour position. Thanks to a wildfire simulator, the authors tested the algorithm inside a realistic environment.

[Julian 2019] introduces an autonomous wildfire surveillance system, using a deep reinforcement learning approach to control a fleet of UAVs. The problem is modeled after a Partially Observable Markov Decision Process for the UAV controller state and actions, and a stochastic wildfire model is used to generate the training environment. Two different strategies for generating actions for each UAV in the fleet are depicted. The first one directly derives actions from binarised images of the wildfire around each UAV (burning or not burning). A penalty system favors the aircraft to follow the fire front while avoiding extreme bank angles and flying close to other UAVs. The second strategy defines an ignition belief map for the whole fire, encoding for every cell whether it is on fire and the time elapsed since the last observation. Each time an UAV flies over an area, the belief map is updated and the fleet obtains a reward when a cell thought to be extinguished is seen on fire. Deep reinforcement learning is used to train the controllers for the two aforementioned strategies. The authors study the pros and cons of each approach and analyze their performance against each other under several distinct scenarios. The results show that both approaches are able to track wildfires, none of them outperforming the other in any case.

2.3 Autonomy level classification scheme

This section synthesizes the publications with respect to the *Autonomy* of the systems, defined as the ability of a robot to perform a given task with the least human involvement.

Several pieces of work have dealt with the issue of classifying unmanned aerial systems by their *Autonomy* abilities. A study by the US Air Force Research Laboratory [Clough 2002] proposes an Autonomous Control Level (ACL) metric di-

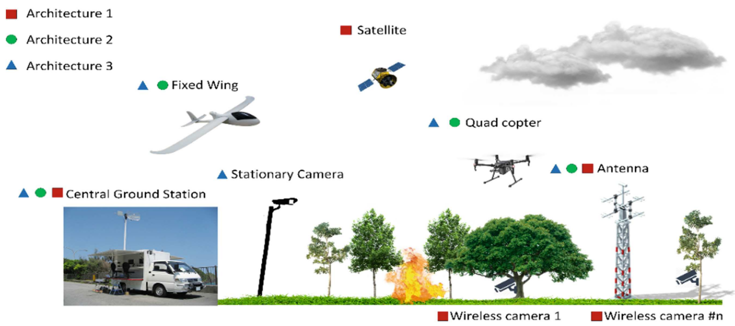


Figure 2.5: Due to the extension and the duration of wildfires, teams of vehicles, multiple sensors and communication schemes must be considered for a successful wildfire monitoring mission. *Excerpt from [Moulianitis 2019]*

vided in 10 levels, from *Remotely piloted* to *Fully Autonomous Swarms*, based on three metrics: Perception/Situational Awareness; Analysis/Decision Making; and Communication/Cooperation. The Autonomy Levels for Unmanned Systems (ALFUS) [Huang 2007] is a popular classification by the USA National Institute of Standards and Technology that takes in account three aspects of the overall completion of a mission: mission complexity, environmental complexity, and human independence. A working group of NATO has classified UAS autonomy in 4 levels [Protti 2007]: remotely controlled system, automated system, autonomous non-learning system, and autonomous learning system.

These three classification schemes have been established considering mostly military Unmanned Aerial Systems, and some of their definitions are not suitable to fully describe civil missions. The scales for the different autonomy levels depend on specific abilities such as threat detection and flight tactics that are not fully compatible with wildfire remote sensing particularities. Nevertheless, the principles that define these classifications can be adapted to the remote sensing context.

2.3.1 Main metrics

An important difference with respect to previous work is the enlargement of the analysis scope to the overall system. The classification is no longer restricted to the elements that are embedded into the UAVs but considers all the system devices, protocols and algorithms that constitute the whole system: satellites, ground sensors, ground control station, etc.

We propose the following three following metrics to establish a classification of UAV based wildfire monitoring systems:

Awareness Whether the system is able to provide a global understanding and analysis of the fire state and properties.

Decision Whether the system is able to decide which actions to perform and achieve them.

Collaboration Whether the system components communicate, share information or perform joint tasks.

The autonomy level of a system is described by these three main metrics, seen as mostly independent qualitative dimensions. The proposed classification does not provide numerical scores, but a qualitative synthesis of system strengths.

2.3.1.1 Awareness

The *Awareness* metric extends elementary fire perception abilities (e.g. fire detection) to the analysis of the observed data. It represents the ability to synthesize observations into a computerized understanding of the observed phenomenon.

The Awareness metric is divided into three levels:

No data analysis Acquired data is disseminated with no further analysis.

Feature extraction The system is able to detect some features of a wildfire from sensor data – for example, detecting fire hot-spots.

Situation assessment The system is able to exploit sensor data and models to provide additional insights about the wildfire, such as estimation of missing data or prediction of wildfire evolution.

2.3.1.2 Decision

The *Decision* metric defines level of intervention of the users on the definition of the actions to be carried out by an UAV.

This metric presents two distinct stages of increasing levels of autonomy, depending on whether the UAV *adapts* its actions with respect to the environmental or not. At the first stage, UAVs are able to navigate by themselves following a human-made plan specifying how to perform the mission. For instance: "*Take off, then go over point A, take a picture, and land at point B*". At the second stage, the mission plan is not fully specified by an operator: it is automatically generated from high-level requirements and constraints provided by the users. For example: "*Find fires in this region*" or "*monitor this fire perimeter*".

These two stages are detailed in the following four autonomy levels of the *decision* metric:

Remotely piloted The UAV is directly piloted by an operator.

Manual planning The UAV is able to navigate autonomously given a plan provided by a human operator.

Autonomous planning The UAV is able to navigate autonomously a trajectory generated by a computer taking in account the current or expected environment.

Adaptive autonomous planning The UAV is able to navigate autonomously a planned trajectory and modify it while flying according to changes in the environment.

2.3.1.3 Collaboration

The *Collaboration* metric evaluates the existence and the autonomy level of interaction between several UAVs. Given the spatial and time extents of wildfires, the benefits of multiple vehicles for observation missions are clear. Nevertheless, having a fleet of vehicles inevitably requires taking in account a collaboration dimension in the design, as the control algorithms must be adapted to this configuration. In this respect, two classes of designs can be found: one approach is to have a fleet of UAVs with independently allocated tasks executing actions in parallel to achieve the mission. For instance, the monitoring of a $10\,\mathrm{km}^2$ region can be done with 10 UAVs patrolling areas of $1\,\mathrm{km}^2$. Another design approach is to make UAVs work together, exploiting synergies to achieve a common objective. Good examples of this strategy are [Casbeer 2006] and [Merino 2005]. The first is an instance of swarm design, that yields the estimation of a fire perimeter length with a fleet of vehicles having limited sensing thanks to local information exchange. In the second case, UAVs carrying different sensors observe the same fire from different locations to increase the accuracy of the detection.

This metric is divided into the following three levels:

One vehicle No collaboration.

Distributed task Multiple UAVs achieve independently allocated tasks.

Cooperative task Multiple UAVs for which one vehicle task execution requires the interaction with one or more other vehicles.

2.3.2 Secondary concerns

There are other aspects playing a complementary role into the characterization of wildfire remote sensing systems. These concerns do not take part directly of the autonomy definition, but are still worth to be mentioned as they help put those designs into context.

2.3.2.1 On-board versus off-board data processing

On-board processing means that the vehicles are capable to process sensor data and to produce a synthesis of the information on their own. On the contrary, off-board processing defines a vehicle that only acts as the carrier of a sensor, transmitting raw data or storing it for further retrieval. The distinction between on-board and off-board computation is not sharp, as data processing efforts may be shared between the vehicles and the ground station.

The choice of a data processing configuration comes out as a technical consequence of multiple combined factors: 1. The ratio between the performance of the communication link and the size of the data that needs to be transmitted, 2. The computing power that can be embedded into the UAVs, and 3. Mission-specific requirements.

2.3.2.2 UAV airframe type

UAVs may be classified by their structure and method of lift, some airframes being more suitable for specific types of mission than others.

Lighter-than-air vehicles are able to fly during long periods of time and can control their trajectories, but they navigate very slowly and are prone to winds. This kind of vehicle is suitable for static surveillance missions as used in [Merino 2012], taking pictures from a high point, and long term observation of extinguished areas like in [von Wahl 2010], but not for wildfire perimeter tracking when wind is one of the main cause of the fire propagation.

Rotary-wing aircraft can take off and land vertically and hover, but they require the continuous action of the rotors to fly, reducing their endurance. Quadrotors, and multirotor aircraft in general, have recently gained some popularity thanks to the introduction of relatively cheap commercial off-the-shelf models.

In contrast, fixed-wing planes have to be continuously moving forward, they are not able to hover and are non-holonomic. Nevertheless, they are very useful as their airspeed and endurance are high. These two reasons make them the airframe of choice for fire mapping and tracking missions.

New UAV designs with hybrid airframes are capable of vertical take-off and landing and gliding flight, combining the advantages of rotary-wing and fixed-wing aircraft designs, but are not yet widespread.

2.3.2.3 Mission type

In the context of wildfires, UAV missions are devoted to information gathering, be it *detection*, *mapping*, *monitoring* or *tracking* of either *hot-spots* or *fire perimeters* (also referred to as *fire front*). Fire *suppression*, one of the objectives of [Haksar 2018], is the sole mission that does not fall into the remote sensing category.

Detection missions consist in finding the existence of fire and its coordinates. The objective of a mapping mission, typically performed during the early stages of a wildfire, is to build a map of the fire extent. Depending on the kind of sensor used, visible or thermal, this can be for the fire front or hot-spots. When mapping is done continuously for a region as a whole, the mission type is monitoring; or tracking, if a particular feature of the wildfire is being followed. Surveillance is a variant of the monitoring mission type where UAVs are used to observe an area before or after a fire is declared.

2.3.2.4 Field tests

The maturity of research and development can be assessed on whether it has been tested in real conditions or not. Among all the systems reviewed, those that have been field tested have single vehicle configurations (with [Merino 2005] as the sole exception). This fact is consistent with the analysis of [Tang 2015] relating current legal issues of UAV usage. It can be expected that in the coming years there is going to be a push for the authorization of multiple UAV operations, as technology gets more mature.

2.4 Discussion

Table 2.1 synthesizes the work reviewed in section 2.2 with respect to the metrics introduced in section 2.3. Publications are sorted by year and grouped into one record when they describe parts of the same system. Additionally, Figure 2.6 shows a condensed pairwise comparison of the most common wildfire remote monitoring system configurations with respect to the defined autonomy level metrics. The charts highlight the most frequent levels and those that are not popular.

As expected, we observe an overall increasing level in autonomy in the recent years. In particular, every publication after [Merino 2012] in 2012 describes a system featuring some kind of Situation Assessment, that is either creating some sort of map of the wildfire location, or recovering the fire geometry: Situation Assessment is naturally the first step to autonomous wildfire monitoring. Furthermore, most of recent work in the literature reach the autonomous planning decision level with multiple UAVs. This is a consequence of the nature of wildfires, which extend both in space and time, and of the requirements of the operators, who need as timely and as complete as possible information. As a result, it is likely that fleets of autonomous UAVs are going to be the standard for future wildfire remote sensing developments.

If the recent trend in publications shows that using multiple UAVs are possibly the best fit to observe wildfires, such technology is not completely ready yet. There is a general lack of extensive field experiments, that may be due to a couple of overcoming challenges. The first is legal: flight restrictions require a dedicated safety pilot for every vehicle, which is an issue for fleets of autonomous UAVs. The second may be due to autonomy challenges: reaching the collaboration level of autonomy requires an advanced level of decisional autonomy. As illustrated in Figure 2.6c, no system with multiple UAVs is remotely piloted.

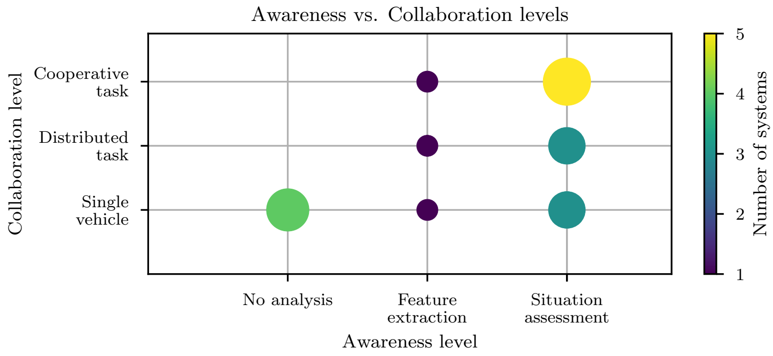
Besides, Figure 2.6b illustrates well that autonomous planning systems must rely on situation assessment algorithms. Finally, one can state that the future of wildfire remote sensing is dependent on a general increase in the autonomy of UAV technology. Additional field testing will be required in order to certify collaborative UAV activities, for which system-oriented designs that explicitly account for the ground control station and operator involvement are deemed necessary.

Table 2.1: Synthesis of the reviewed UAV-based wildfire remote sensing systems.

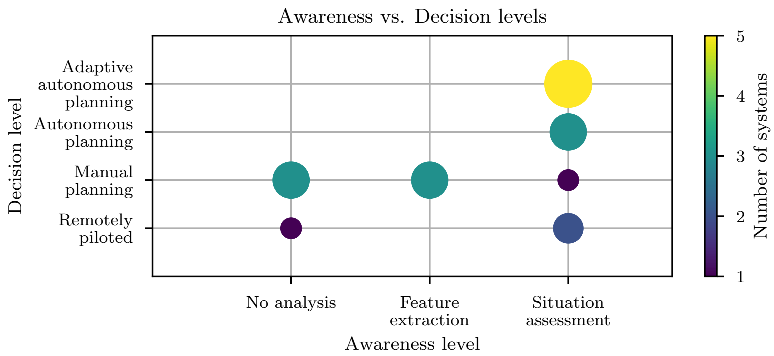
Publication	Mission type	Decision level	Awareness level	Collaboration level	Information Processing	Airframe	Fielded
[Dovis 2001]	Detection	Manual planning	Feature extraction	Cooperative task	On-board	HALE fixed-wing	No
[Ambrosia 2003]	Mapping	Manual planning	No analysis	Single vehicle	Off-board	HALE fixed-wing	Yes
[Casbeer 2006]	Front tracking	Adaptive autonomous planning	Situation Assessment	Cooperative task	On-board	LASE fixed-wing	No
[Restas 2006]	Surveillance	Remotely piloted	No analysis	Single vehicle	No	Quadcopter	Yes
[Esposito 2007]	Mapping	Manual planning	No analysis	Single vehicle	Off-board	LASE fixed-wing	Yes
[Lewyckyj 2007]	Mapping	Manual planning	No analysis	Single vehicle	Off-board	HALE fixed-wing	?
[von Wahl 2010]	Hot-spot surveillance	Manual planning	Feature extraction	Distributed task	Off-board	Quadcopter and blimp	No
[Ambrosia 2011]	Detection and mapping	Manual planning	Feature extraction	Single vehicle	On-board	HALE fixed-wing	Yes
[Merino 2005] [Martínez-de Dios 2011] [Merino 2012]	Hot-spot and front detection and flame geometry	Autonomous planning	Situation Assessment	Cooperative task	Off-board	Helicopter and airship	Yes
[Howden 2013]	Detection and mapping	Adaptive autonomous planning	Situation Assessment	Cooperative task	On-board	Rotary-wing	No
[Skeele 2016]	Front tracking	Adaptive autonomous planning	Situation Assessment	Single vehicle	On-board	Quadcopter	Yes

Table 2.1: Synthesis of the reviewed UAV-based wildfire remote sensing systems.

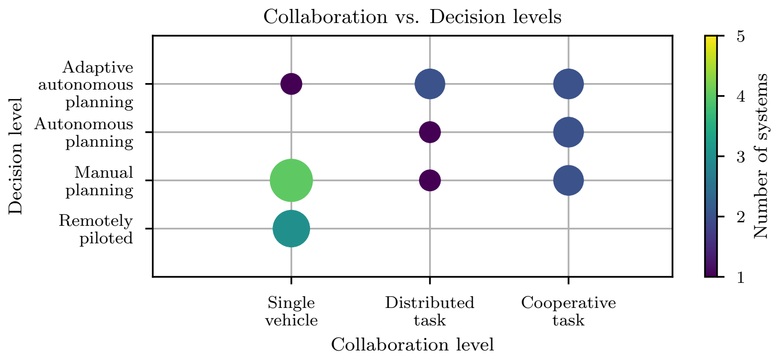
Publication	**Mission type**	**Decision level**	**Awareness level**	**Collaboration level**	**Information Processing**	**Airframe**	**Fielded**
[Valero 2017]	Mapping	Remotely piloted	Situation Assessment	Single vehicle	Off-board	Quadcopter	Yes
[Pham 2017]	Monitoring	Adaptive autonomous planning	Situation Assessment	Distributed task	On-board	Quadcopter	No
[Rabinovich 2018]	Front tracking	Autonomous planning	Situation Assessment	Distributed task	Off-board	LASE fixed-wing	No
[Ciullo 2018]	Flame geometry mapping	Remotely piloted	Situation Assessment	Single vehicle	Off-board	Quadcopter	Yes
[Haksar 2018]	Detection and suppression	Adaptive autonomous planning	Situation Assessment	Distributed task	On-board	Quadcopter	No
[Pérez-Lissi 2018] [Bit-Monnot 2018] [Bailon-Ruiz 2018]	Monitoring	Autonomous planning	Situation Assessment	Cooperative task	Off-board	LASE fixed-wing	Yes
[Julian 2019]	Mapping	Autonomous planning	Situation Assessment	Cooperative task	Off-board	LASE fixed-wing	No
[Lin 2015] [Lin 2018] [Lin 2019]	Front tracking	Manual planning	Situation Assessment	Cooperative task	Off-board	Quadcopter	No
FireRS [Bit-Monnot 2018] [Bailon-Ruiz 2018] [Pérez-Lissi 2018]	Monitoring	Autonomous planning	Situation Assessment	Cooperative task	Off-board	Fixed-wing LASE	Yes



(a) Awareness vs. collaboration autonomy



(b) Awareness vs. decision autonomy



(c) Collaboration vs. decision autonomy

Figure 2.6: Examination of the wildfire remote monitoring systems reviewed in this survey. Charts provide aggregated pairwise comparisons of the awareness, decision and collaboration autonomy levels highlighting the most common configurations. The size and the color of the circles indicate the number of contributions.

2.5 The FireRS system

The work presented in this manuscript has been achieved in the context of the FireRS project, funded by the European Union through the Interreg Sudoe V program. FireRS (acronym for wildFIRE Remote Sensing) aims to provide emergency agencies and coordination centers with an innovative tool for detecting and managing wildfires. Using a fleet of UAVs, ground sensors, and a constellation of picosatellites, FireRS will provide firefighters with real-time information on the fire status: its position, its perimeter, and its evolution.

2.5.1 Objectives

The purpose of FireRS is to design a wildfire detection and monitoring tool for fire management and suppression teams.

The objective is to build a perception machine that provides firefighting crews with timely updated information about the evolution of a wildfire. FireRS is meant to be a tool allowing the precise detection and comprehensive monitoring of wildfires and the production of propagation forecasts.

The vision for the FireRS project is to help wildland firefighters during several stages of the wildfire suppression efforts, by *(i)* reducing the time necessary to confirm and locate an ignition after the first alert without the need of putting human in danger, and *(ii)* providing a timely updated map of the fire perimeter in a safe, cost-effective, manner even in situations where current procedures are not able to do so.

Note that it is out of the scope of FireRS to provide instructions to the fire management team about suppression strategies.

2.5.2 Operational view

The FireRS platform, depicted in Figure 2.7, is based on a fleet of fixed-wing UAVs, a network of land sensors for fire detection, and constellation of picosatellites for communication, as well as a computer software infrastructure that manages the system operation in coordination with the operators, and performs wildfire situation diagnosis and prognosis.

The network of land sensors, carefully placed, monitors a wildland region for fire ignitions, raising an alarm when a hot spot is detected. A UAV then autonomously flies to the location, and thanks to its infrared camera assesses the presence of fire. The information acquired by the UAV is transmitted to the data processing software, where it is used to generate a fire map and a propagation forecast that are finally displayed to the operators. The satellite communications system allows the data transmission in remote locations where reliable coverage does not exist.

During an uncontrolled wildfire propagation, the FireRS system ensures the monitoring phase of the fire with the fleet of UAVs. Using previously gathered

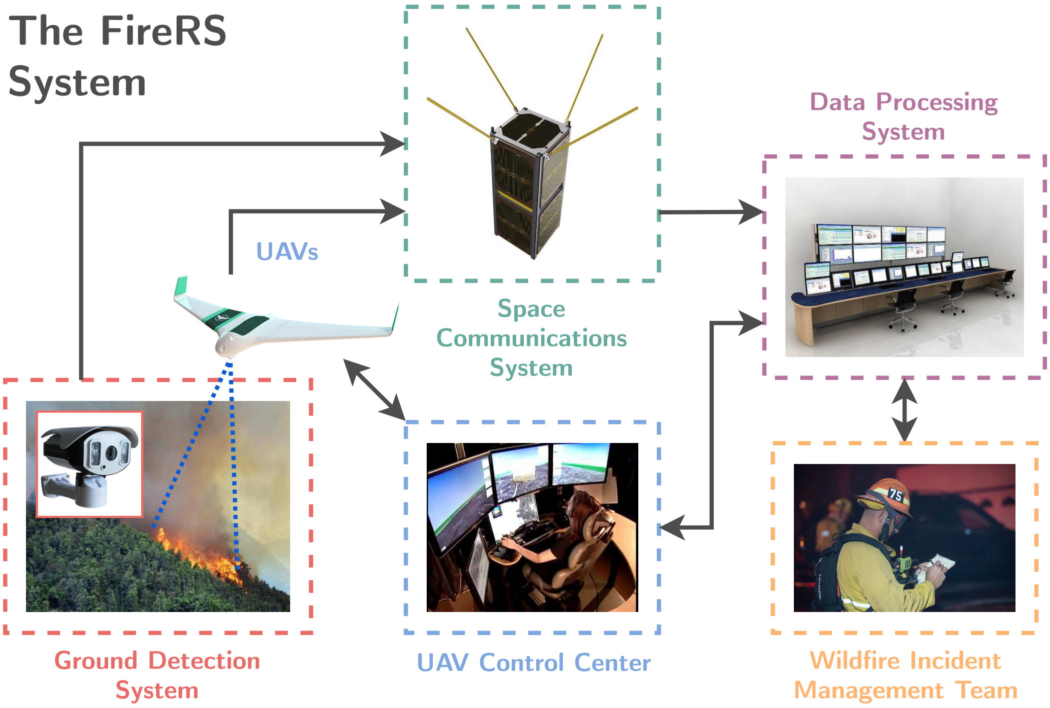


Figure 2.7: The FireRS system global architecture

data, the fire map, and the spread forecast, a monitoring plan that obeys operator demands is automatically generated to track the perimeter of the wildfire. Newly gathered information is timely sent to the data processing center to improve the fire maps – and hence the overall situation assessment.

2.5.3 Approach

The FireRS system works along a Perception–Decision–Action scheme.

The fleet of UAVs and the land sensors are the *eyes* of the system. Following an established plan, they gather information about the fire location used afterwards to produce fire maps and forecasts. UAVs are equipped with infrared cameras the enables them to create local fire maps. Land sensors are made of infrared and visible cameras, used to locate fire spots, and sensors that provide some meteorological information needed for the production of accurate wildfire propagation forecasts (wind speed and direction, and humidity). A situation assessment algorithm fuses the information gathered by the different UAVs and the lands sensors to create a unique fire map that encodes the past and current fire location belief. Likewise, the situation assessment algorithm uses a wildfire simulator software based on state-of-the-art models to produce a fire spread forecast that is merged into the global fire map.

The observation plan is generated by a search algorithm that finds the optimal paths for the fleet to observe the fire, on the basis of the current fire map and UAV motion restrictions. Next, the resulting paths are sent for execution in real or simulated UAVs. The FireRS system is designed to work in different levels of realism, from real UAVs in a real environment to simulated UAVs in a virtual environment, via by hybrid configurations – real UAVs in a virtual environment. Moreover, the perception, situation assessment, planning and execution subsystems of the FireRS are designed in a modular manner, allowing further developments on models and algorithms. For instance, the control software for the UAVs could be changed for another one, or the wildfire simulator could be changed for a more accurate model.

The properties of the FireRS system with respect to our analysis metrics are summarized in the last line of Table 2.1.

Part II

Models

CHAPTER 3

Wildfire models

Wildfires are uncontrolled vegetation fires that are difficult to model. Still, a lot of efforts have been put on developing models that allow to predict their propagation. This chapter introduces the reader into the foundations of the wildfire propagation process and the main factors that influence its spread, making an emphasis in studying the possibilities and limitations of existing models to predict wildfire propagation. The chapter concludes with the implementation of a simplified wildfire propagation simulation software that is at the core of the situation assessment and mission planning processes developed in this thesis.

3.1 Introduction

Fire is the result of the physical and chemical process of combustion, a rapid oxidation of flammable materials. In general terms, a fire is an exothermic chemical reaction that happens when a combustible material, the *fuel*, under the presence of oxygen, the *air*, is exposed to a source of *heat*, liberating carbon dioxide, water and more heat. Then, the thermal energy released initiates a chain reaction sustaining the combustion process. The interaction between the three agents, plus the effect of heat, is typically pictured as the fire tetrahedron (Figure 3.1) and removing any of these elements results on the extinction of the fire. This can be achieved by building a firebreak, by reducing the speed of the reaction through the use of fire retardants or just by spraying water to reduce temperature and increase fuel moisture.

A wildfire, also known as wildland fire or forest fire, is a vegetation fire that occurs in rural areas, not originated in inhabited places, yet having sometimes some impact into populated areas. A wildfire simply starts when an ignition source is approached to the vegetation fuel, either naturally (e.g. a lightning strike) or artificially, by human accident or arson. Subsequent fire spread depends on complex interactions between numerous fuel condition factors, topography and weather history.

The physical process of *heat transfer* happens under three distinct mechanisms: conduction, convection and radiation. In a wildfire, conduction, heat transfer by direct contact, is the least prevalent of the tree. On the contrary convection and radiation play a major role. Convection is the result of hot air being displaced to colder areas – which in the case of a wildfire can affect local weather when it becomes very large. Radiation is the emission of thermal energy, in the form of electromagnetic waves that matter emits just because of having some temperature above absolute zero. Wildfire flames can reach temperatures around $1000\,^{\circ}\mathrm{C}$, so a

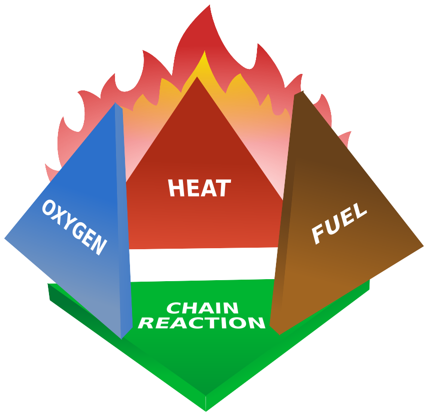


Figure 3.1: The fire tetrahedron. This geometric shape represents the four necessary components to produce a sustained fire. Removing any of them results on the exhaustion of the combustion reaction. (By *Gustavb*, published under public domain)

lot of energy is emitted in the form of infrared radiation, incidentally enabling the use of infrared cameras for its detection.

An important definition on wildfire propagation modeling studying is the concept of *fire behavior*: "a general term used to describe physical aspects of the combustion process such as speed and direction of fire spread" as described by [Keane 2015]. Other terms of the wildfire community jargon are *fire spread*, to denote the wildfire speed in a particular direction, and *growth*, to denote the evolution of its size.

3.1.1 Parts of a wildfire

A wildfire usually starts from a point ignition when heat is applied to combustible material. Then, the energy liberated by combustion is transferred to the surrounding unburnt vegetation raising its temperature until it sets on fire. Once the fuel has been consumed, the combustion operation stops. If fuel distribution is continuous and no measures are taken to suppress the fire, a wildfire is an ever expanding closed perimeter of flames, also known as the *fire front*, enclosing a region of burnt vegetation.

The shape of the wildfire perimeter depends on the time integral of fire spread in every direction, given by the many factors that determine wildfire propagation. In general terms, a fire on evenly distributed vegetation, in a flat terrain, and under the presence of constant wind follows roughly an elliptical shape with the larger axis aligned with the wind direction. This is illustrated in Figure 3.2 with a satellite image of a wind-driven wildfire where hot-spots have been highlighted. As pictured in Figure 3.3, the fastest moving portion of the wildfire perimeter is the *head* or *front fire* in the direction the wind is blowing to. The *back fire* is on the opposite side and the *flank fire* on the sides.

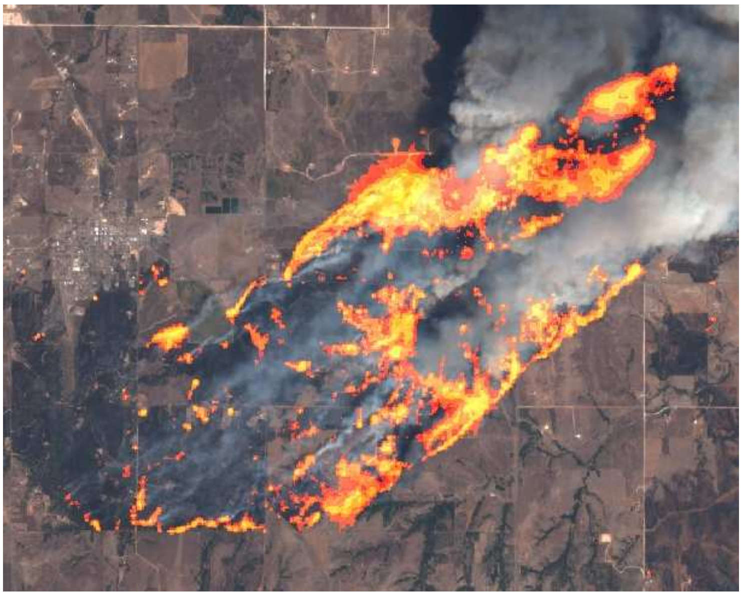


Figure 3.2: Satellite image of a wind driven wildfire. Hot-spots – areas with active fires or residual heat – are highlighted in bright red from the near-infrared and short wave infrared band emissions.

A wildfire in an uneven terrain or subject to changing environment conditions will not follow exactly an elliptical shape. In this case, *bays* and *islands* of unburnt terrain can be formed. *Spot fires*, new fires that ignite beyond the head fire, may occur under strong winds.

3.1.2 Environmental factors of wildfire propagation

This section describes the three natural elements that have an impact on wildfire behavior in descending level of importance: Fuel, weather and topography.

3.1.2.1 Fuel

Wildland fuels are the most important factor on wildfire behavior as without them no ignition would be possible. In a natural environment, fuels are found in what is called the fuelbed: "the complex array of biomass types for a given area" [Keane 2015]. In other words, every piece of combustible material that can be found in an ecological habitat. Trees are the most prominent elements in a typical forest, but many others exist, shaping the environment: live vegetation like saplings, shrubs, grass and roots, and dead vegetation such as duff, decaying plant litter, down logs and stumps.

The fuelbed is stratified into three vertical fuel layers [Keane 2015]: *canopy*, *surface* and *ground*. The canopy fuel layer is the top most stratum, encompassing tree trunks and crowns at least $2\,\mathrm{m}$ above the soil level. It is characterized by its canopy cover density, stand, base height and bulk density. Just below the canopy is

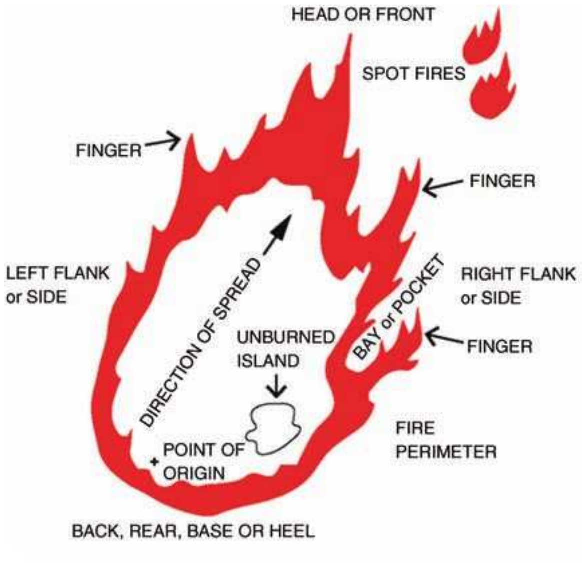


Figure 3.3: Parts of a wind-driven wildfire. *Excerpted from [Scott 2014].*

the surface fuel layer, where herbs, grass, shrubs and litter can be found. Finally, the ground fuel layer contains the duff that constitutes the forest soil over ground level.

The reason behind this fuelbed layer classification is wildfire behavior. Ground fires propagate by smoldering, a flame-less combustion, at slow speed (few centimeters per hour) while surface and canopy flames spread at a much faster rate (tens of meters per minute) [Scott 2014]. As a consequence, different wildfire propagation models are needed for each forest layer.

Every fuelbed layer in the wildland has particular physical and chemical properties that affect fire behavior depending on the kind vegetative elements it is composed of. Defined at a coarse level, these properties are: particle size, shape and density, fuel load, depth and distribution, and live/dead vegetation ratio. Furthermore, fire behavior greatly vary depending on the fuel moisture content as a result of the effects of weather phenomena. At the end, fuel moisture is the fundamental parameter the determines the possibility of a fire ignition or the continuity of an existing wildfire.

3.1.2.2 Weather

The effect of climate and weather is dual. On the one hand it has an indirect impact on wildfire behavior, determining wildland fuel composition and some of its properties, as discussed above. On the other hand, it has a direct impact on wildfire propagation primarily by the effect of wind, which is the most important factor defining wildfire shape and propagation direction.

Indirect effects At a global scale, climate has an important influence on world biomes which are the home of different wildland fuels. The presence of rain forests, taiga, savanna, tundra, shrub land, etc. depend manly on world climate and have clear different fuel composition. For instance, rain-forest is shaped by tall, dense canopies as a result of continuous hot and humid weather. Savannas are subject to seasonal rain, which results in varying humidity and live/dead vegetation ratio throughout the year.

Direct effects At a local scale, precipitation, air humidity, cloudiness and ambient temperature have an impact on short term and long term fuel moisture, and, as a consequence, a direct influence on wildfire behavior. An unstable atmospheric condition like a storm gives a cloudy sky and heavy rain, hence reducing the risk of a wildfire outbreak. However, strong winds result on a drier atmosphere. Frequently, a lightning strike is the cause of a wildfire ignition.

Wind deserves a special mention regarding wildfire behavior: it is indeed the main factor that dictates the propagation direction of the fire front. Furthermore, wind is also very variable, changing drastically depending on the time the day, the current temperature and cloud cover. Additionally, it contributes to wildfire spread through *spotting*: The transport of embers that provoke new ignitions beyond the current fire front (*spot fires*).

In extreme cases, very large wildfires have an incidence on local weather in the form of *firestorms* producing strong winds that live up the fire [Fromm 2006].

3.1.2.3 Topography

The third environmental factor having an impact on wildfire behavior is topography. Besides the known coarse influence of geographic accidents over world climate, and thus on natural environments and fuelbed composition, topography has a direct effect on wildfire propagation at a local scale.

Steep terrain causes faster fire propagation due to the geometry of the interaction between flames and the slope: As illustrated by Figure 3.4, the steepest the ramp is, the closest the flames are to more elevated fuels. Terrain geometry also plays a fundamental role in local surface wind formation with valleys, canyons and ravines favoring stronger winds.

To a lesser extent, altitude, terrain aspect – the orientation of the slope – and geographic accidents have an influence on fire behavior trough fuel distribution. The more elevated a place is, the colder it is and more humid it is resulting in reduced fire risk. South-facing mountain slopes receive more solar radiation that north-facing ones, and are typically hotter and drier. Finally, geographic accidents, man-made or natural, can act as barriers to wildfire spread (e.g. A firebreak, a highway, or a river)

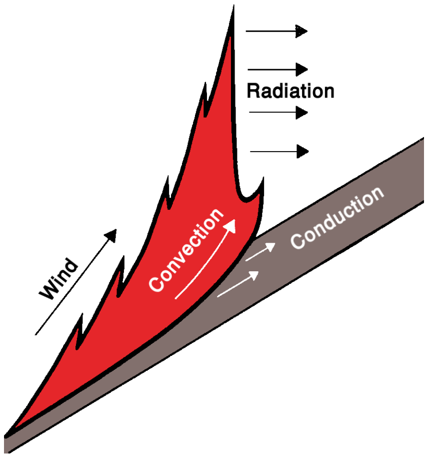


Figure 3.4: Illustration of the heat transfer mechanisms of an uphill wind-driven surface fire. *Adapted from [Rothermel 1972].*

3.2 Modeling wildfire behavior

Wildfire behavior modeling efforts began during the 1920s with the goal of deriving systematic relationships between wildfire spread and environmental variables. Hawley [Hawley 1926] and Gisborne [Gisborne 1927], from the USDA[1] Forest Service, were the pioneers on the subject. At that time, coarse propagation forecasts were provided by senior firefighters based on their experience but those suffered from the limitations of human discernment. Operational prediction tools were deemed necessary to improve the effectiveness and safety of fire suppression operations. Since then, numerous wildfire models have been created, mostly with important funding from US, Canadian and Australian research agencies, and more recently by the European Union. The complexity of wildfire behavior due to entanglement of multiple physical and chemical processes has not leaded yet to perfectly accurate models, even less to a universal one. Depending on the type of fire or on which particular aspect of wildfire behavior more accuracy is needed different modeling approaches are used. Reviewed at a large in [Sullivan 2009a, Sullivan 2009b], existing models belong essentially to one of two categories: physical models and empirical models.

Models of the two categories are useful in their own context. Physical models apply the scientific principles of combustion: chemical kinetics, heat transfer and fluid mechanics. They can provide accurate results when given precise information about the environment and for that reason they are very useful in wildland fire research. ForeFire [Filippi 2009] is an example of a physical model based simulator with a fire propagation model coupled to an atmospheric model. Because of that, ForeFire is able to exploit realistic wind profiles to simulate smoke plumes in addition to wildfire propagation. However, the computational cost of these complex

[1]United States Department of Agriculture

simulations is high: the authors of ForeFire state that a small-scale ($1\,\mathrm{km}^2$) simulation of 50 minutes takes 4 hours in a dual core Intel Xeon CPU at 3Ghz and 8 GB of RAM. Moreover, authors acknowledge that "realistic large-scale simulations [...] require a computer between 100 and 1000 times faster than the test platform". This means that the simulator cannot be used currently during firefighting with a regular computer.

On the other hand, empirical or statistical models, based on equations derived form observation are generally easier to use and less expensive computationally. They can be quite accurate at the expense of being specifically tailored to a particular circumstance and a lack of explicability. As a consequence, they cannot be extrapolated to new situations or handle some corner cases for which another specific model is required.

In practice, operational models used nowadays are quasi-empirical. That is, hybrid models that try to find the right compromise between the exactitude of physical models and the ease of use of empirical ones [Scott 2014]. The most widely wildfire spread model used is Rothermel's model, introduced by Richard C. Rothermel in 1972 [Rothermel 1972]. After seeing some corrections and improvements, this model is still at the heart of relevant wildfire simulation software like BehavePlus [Andrews 2014] and FARSITE [Finney 1998]. The reason for its success, in addition to being fast, resides on two factors. First, it is attractive for firefighters because it has been designed for operational use in mind: the inputs and the equations correspond to magnitudes typically measured, and its easy to understand. While physical models are definitely more accurate, they also depend on input information that is not readily available and not feasible to obtain during an emergency. Second, the propagation estimates are considered good enough for a standard surface fire in steady state conditions.

3.2.1 Limitations

Wildfires are by nature uncertain because of the multitude of factors they depend on. As a consequence, wildfire models are not exempt from accuracy errors and the possibility of serious prediction mistakes are a fact that limits their usage for long term wildfire forecast. Furthermore, error analysis of models is not trivial due to a high number of variables and to the strong non-linear structure of processes. Yet, the extent of the limitations of wildfire behavior models have already been studied in the literature. In particular, [Albini 1976] establishes three possible sources of error:

- Model applicability
- Model accuracy
- Data accuracy.

Model applicability errors appear when a model is used in a wrong situation. Fuel, as the main factor in wildfire propagation, also has an influence on modeling. Each of the three fuel layers, canopy, surface and ground, have very distinct ways of fire spread. Smoldering combustion is predominant in ground fires, making propagation slow and flame-less. Surface fires involve the combustion of a single layer of continuous fuel. Crown fires are, in short, aerial fires having a completely different interaction with the environment, especially regarding weather. Wildfire behavior models are tailored to one specific class of fire. For example Rothermel's model behavior is tuned for surface fires. Efforts have been devoted since the original publication date to also add crown fires into the model [Rothermel 1991, Andrews 2011], but the results are not as good as for the surface ones.

Wildfire model accuracy research is mainly based on prescribed fires. A prescribed fire is a real fire ignited in a controlled natural environment for the purpose of research, forest management or farming. Because of the restricted nature of wildfires for the purpose of research, it is rare to have models adapted to extreme situations. The study of infrequent wildfire events is less likely to occur because the situations that lead to extreme outcomes are not easy to reproduce in a safe environment.

Finally, wildfire models are not linear and depend on multiple parameters. While accurate data is available for some model inputs, other are essentially variable and uncertain. The fuelbed is typically considered uniform, continuous and homogeneous although it may not be the case. Wind can be easily measured but it can change at any time. Fuel moisture has to be estimated from the continuous evolution of past weather. Despite the existence of accurate models, wrong measurements may cause important modeling errors.

3.2.2 Wildfire situation assessment

Firefighters typically make use of wildfire maps and forecasts to design fire suppression plans and to ensure the safety of people nearby. Rapid procurement of information on a wildfire is essential to reduce the response time, which is crucial in wildland firefighting: the fastest a fire ignition is controlled, the least the suppression costs and damages are. Wildfire propagation predictions have to be as accurate as possible to get fire suppression plans as effective as possible, but, as discussed previously, forecast models have limitations that prevent them to be the definitive source of information that operators can rely upon. As forecasts can be uncertain, firefighters may run multiple simulations with different input conditions in order to evaluate all the main possible outcomes of a wildfire spread. Likewise, as new information about the wildfire or the environment is measured, plans have to be reconsidered with the most recent pieces of data in hand.

The impossibility of providing firefighters with systematic and perfectly accurate evaluations of the current and future state of a wildfire using only model propagation yields the need for alternate strategies to reduce uncertainty. In particular, a simple wildfire model can be used to provide a coarse estimate of the wildfire propagation, while information gathering means such as a fleet of UAVs can be used to refine the forecast when and where necessary. If this operation is done frequently, it is possible to produce a continuous and reliable assessment of the current wildfire situation and near future predictions.

3.3 Design of a wildfire simulator software

The implementation of a simulation software is an essential element for this thesis. In particular, it is used to provide short term wildfire growth predictions for the situation assessment and observation planning algorithms, and to generate synthetic fire scenarios. These particular usages are further discussed in the next chapters of this manuscript.

There are two key design requirements for this simulation software: it must allow to predict the main wildfire dynamics to provide reasonably accurate forecasts, while being fast to run for a standard wildfire size. Additionally, the simulator is going to be part of a bigger automated situation assessment and observation planning system, and will be used with other simulators. As such, it has to be designed as a software library with an application programming interface.

The wildfire simulator only needs to handle surface fires because they are the most important wildfire propagation dynamic, which dictates the overall fire spread. Including other behaviors like crown and ground fires is not considered, but they could be added if necessary in a particular context.

Some wildfire propagation software like BehavePlus [Andrews 2014] and FARSITE [Finney 1998] exist and are well known within the wildfire community. Unfortunately they cannot be used in the context of wildfire our work because the have been created as standalone programs with a graphical user interface and no application programming interface. Nevertheless, the proposed simulator resorts to the same well-tested wildfire models.

3.3.1 Wildfire model integration

The design of a complete wildfire simulation software requires the integration of multiple models. The heart of this simulator is Rothermel's propagation model that gives the velocity of spread in one direction. Thus, a growth model is needed to provide estimates of the extent and shape of the fire perimeter.

Rothermel's model predictions require three main pieces of data:

1. Fuelbed characteristics, provided by the fuel model,

2. Terrain slope, provided by a digital elevation maps, and

3. Local wind maps, estimated using a surface wind simulation software.

The result of the wildfire propagation is a wildfire map, which encodes the ignition time of every location in an area, like the one illustrated in Figure 3.5 along the necessary geographic input data maps.

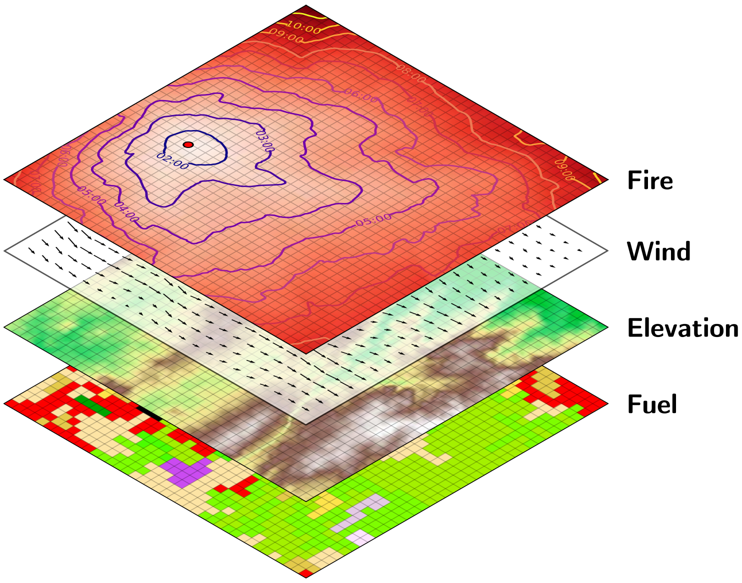


Figure 3.5: Illustration of a wildfire propagation map (top) based on the data provided by the bottom layers: wind, elevation and fuel (from top to bottom). Information is stored in raster form with cells holding values of corresponding magnitudes. The shades of red of the fire map represent the ignition time of each cell. In addition, contour lines have been drawn at regular temporal intervals to ease the interpretation of the map.

3.3.1.1 Forward propagation model

The purpose of the propagation model is to compute the steady-state rate of spread (RoS) of the front fire in units of speed. Rothermel's model equations are established from an energy balance between the energy released by the combustion and the necessary heat for ignition:

$$RoS = \frac{I_p}{\rho_b \varepsilon Q_{ig}} \tag{3.1}$$

Where:

- ρ_b is the *oven-dry bulk density* that determines the efficiency of heating as a function of the particle density,
- ε is the *effective heating number*, a dimensionless factor that depends on fuel particle surface-area-to-volume ratio,
- Q_{ig} is the *heat of pre-ignition*, the amount of energy per unit of mass required for ignition,
- I_p is the *propagating flux*, a component of the rate of spread describing how energy is transferred to the fuel, expressed as:

$$I_p = I_R \xi (1 + \phi_W + \phi_S) \tag{3.2}$$

which depends on the three main wildfire behavior factors:

- I_R, the *reaction intensity* in units of energy per area, represents the heat release rate from the vegetative matter in the fuels. It is a function of the fuel particle size, bulk density, moisture and chemical composition.
- ξ, the *propagating flux ratio* which relates the propagating flux to the reaction intensity.
- ϕ_W, the *wind factor*. A coefficient related to the wind speed. The faster the wind blows, the greater this factor is.
- ϕ_S, the *slope factor*. As for the wind factor, the steepest the terrain is, the more the coefficient grows.

3.3.1.2 Fuel model

The fuel related inputs for the propagation model depend on the physical and combustion properties of the burning material. Researchers have classified typical surface fuel types into fuel models, for which the propagation model inputs have been found experimentally. A technical report by the USDA Forest Service [Scott 2005] defines 40 *standard fire behavior fuel models*, belonging to 7 general fuel types: 1. Grass 2. Grass-shrub 3. Shrub 4. Timber-Understory, grass or shrubs mixed with litter from forest canopy 5. Timber litter 6. Activity fuel (slash) or debris from wind damage, 7. Non burnable. Figure 3.6 shows some terrains associated to each of these 7 models.

3.3.1.3 Shape model

The forward propagation model provides an estimate of the rate of spread of the head fire, but not for the back and flank fires. As a consequence, the surface and the shape of a wildfire cannot be estimated without additional models.

A wildland fire propagating over a flat terrain without wind spreads circularly around the ignition point. When a fire is driven by wind, it spreads faster in the direction of the wind, resulting in an elongation of the perimeter in this direction. It has been found, by the analysis of the final shape of a series of wildfires, that wind-driven wildfire perimeters have an elliptic shape. [Anderson 1983] introduces a fire shape model built which defines the shape of a wildfire as a double ellipse resembling the outline of an egg (Figure 3.7). His work also found that the size and elongation of the shape is only dependent upon wind velocity.

3.3.1.4 Fire growth model

The shape model gives an estimation of the wildfire perimeter shape but does not provide a way to simulate its evolution in time so an additional growth model

(a) High Load, Dry Climate Grass

(b) Low Load Dry Climate Shrub

(c) Agricultural field

(d) Small downed logs

(e) Very High Load, Dry Climate Timber-Shrub

Figure 3.6: Illustration of fuel types described in [Scott 2005].

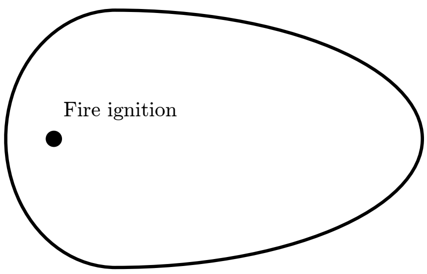


Figure 3.7: Illustration of the wildfire perimeter shape as proposed by [Anderson 1983].

is necessary. From all the models proposed in the literature [Sullivan 2009c], a couple of those have been considered for the task. One of the approaches considers the fire front as a closed polygon, with a finite number of vertices. The other, puts the perimeter in a raster grid of arbitrary resolution.

The vector approach considers every vertex of the perimeter as a new virtual fire ignition, with each of them propagating following local conditions. This technique is based on Huygen's wavelet principle, originally described for wave propagation, applied in the context of wildfire growth. This principle tells that every point of a wave front is also the source of small wavelets that interfere each other so the sum of the interfering wavelets forms a new front. This approach is followed the popular FARSITE wildfire simulator software [Finney 1998] and ForeFire [Filippi 2009].

The Huygen's wavelet principle model provides the advantages of a vector map representation, but it is difficult to implement, and the total number of vertices and the simulation time step have to be carefully chosen in order to keep the computation time reasonable.

With a raster model, fire is depicted as a grid of cells whose value is its burning state, burnt or not burnt. Fire grows by contagion from one cell to another with a speed conducted by the shape and forward propagation models. The raster model has been retained for our wildfire simulator because it is computationally less expensive than the vector method. Additionally, it is also more suitable for heterogeneous landscapes due to the way this data is stored. This technique has been already used in existing software like FireStation [Lopes 2002].

3.3.2 Input data

We have seen that wildfire propagation simulation depends on different environment information, coming from multiple sources, each with a varying degree of accuracy. Relief is known with great precision, wind is known at a coarse level but not locally and the choice of a good fuel model requires precise knowledge of the land. However, all those factors have to be managed in order to be correctly used.

3.3.2.1 Topographic relief

Topographic relief knowledge is needed to obtain the terrain slope and aspect for the fire propagation model. Aspect, the orientation of the slope, is necessary to compute the effective slope, which is the inclination of the terrain at the head fire direction. Relief is also used to get an accurate estimation of the wind speed and direction, as explained later.

Relief is generally depicted in the form of a raster named *Digital Elevation Map* or *DEM* a grid of pixels storing elevation values as illustrated by Figure 3.8. The DEM is geo-referenced and associated to a projected coordinate system (PCS) that maps pixel coordinates to a local flat approximation of the earth surface in a Cartesian frame.

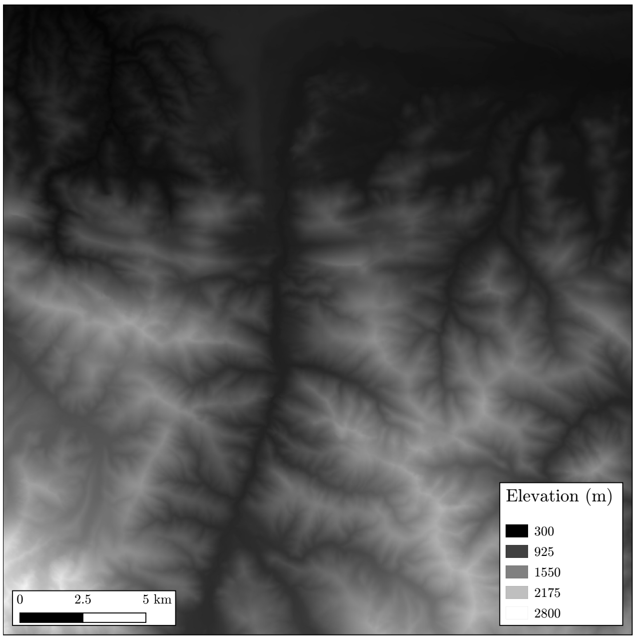


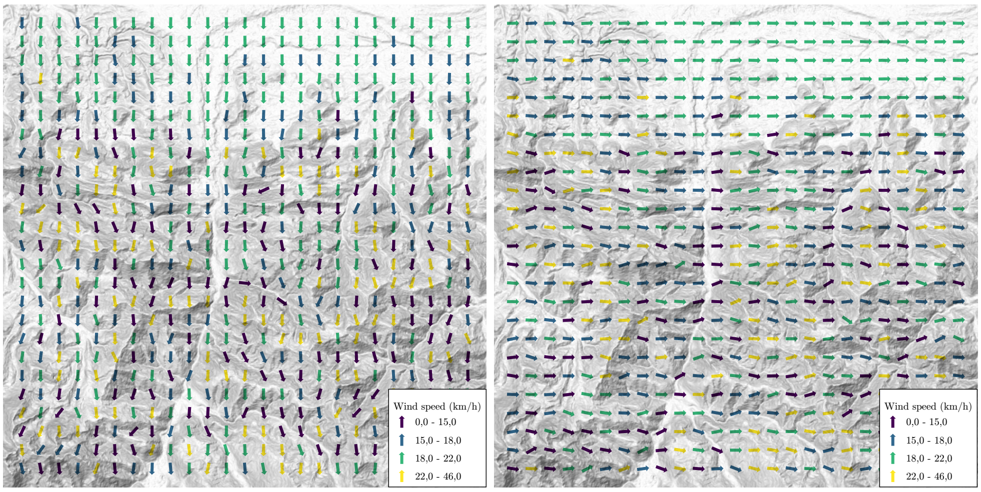
Figure 3.8: Digital elevation map of a $25\,\mathrm{km} \times 25\,\mathrm{km}$ area in the french Hautes-Pyrénées department, featuring a mountainous region in the center and south of the map, a valley, and plains in the north. The altitude range is between $300\,\mathrm{m}$ (black) and $2800\,\mathrm{m}$ (white). The map is encoded as a raster of 1000 rows by 1000 columns, with each cell measuring $25\,\mathrm{m} \times 25\,\mathrm{m}$.

There are numerous spatial reference systems and each country typically defines its own for their official maps. For instance, the french National Geographic Institute publishes maps of France in the RGF94/Lambert93 reference system, which is specifically tailored to represent mainland France accurately. The European Union publishes its maps in the ETRS89/LAEA reference system, encompassing all mainland Europe. Fortunately, thanks to modern Geographic Information Systems (GIS) it is possible to accurately convert maps from one reference system to another. If maps coming from multiple sources have to be used, they must be first converted to a common reference like the popular Universal Transverse Mercator (UTM), coupled to the World Geodetic System (WGS) also used by the GPS.

3.3.2.2 Surface wind

Accurate local data about wind speed and direction is essential to produce good wildfire spread forecasts. The wildfire forward propagation and shape models depend on a precise value of the surface wind speed, that is not generally available. Nevertheless, this value can be fairly well estimated provided by a DEM and the local weather is known.

Driven by the need for estimations of surface wind for wildland fire management, a group of scientists has introduced a couple of wind models sim-



(a) North (0°) (b) West (270°)

Figure 3.9: Surface wind maps obtained with WindNinja using the terrain depicted in Figure 3.8. The software has been configured to take in account time, temperature and cloud cover settings to produce the wind maps. The calculation assumes a mean wind of 20 km/h happening on October the 1st, 2019 at 12 am, with an ambient temperature of 20 °C under a clear sky.

ulating surface wind from a uniform wind field or scattered point measurements [Forthofer 2014]. The result is a model (associated to a computer program named WindNinja[2]) that can compute the surface wind field with varying levels of accuracy depending on the type of information provided. The least accurate setting only requires a DEM and a mean wind value, but one can specify the date and the time of the day, air temperature and cloud cover in order to get more accurate results. While precision can be arbitrarily chosen, exactness comes at the cost of longer computation times.

In mountainous wildland regions, where wildfire suppression is the most difficult, local wind speed and direction greatly diverges from mean measurements. Surface wind tends to blow faster when facing a mountain ridge and slower when behind. Canyons and deep valleys act as corridors changing the wind direction. Additionally, wind depends on the diurnal temperature cycle. With sun exposed slopes warming during daytime and cooling during night, there is an air mass flow caused by the temperature gradient.

3.3.2.3 Fuel

The use of the right fuel inputs is important to get good wildfire forecasts. A fuel map has to depict the fuel distribution in a region using the same fuel classes as in the model, so it can be used to get the adequate inputs for the forward propagation model.

[2] http://firelab.github.io/windninja/

The USDA provides a fuel map for the United States [Reeves 2009] created from the fusion of multiple sensor data coming from satellite imagery. No similar map is available in Europe, but the European Environment Agency (EEA) produces the CORINE Land Cover inventory through its Copernicus program [Büttner 2014]. The objective of the CORINE land cover inventory is to study the effects of human activity and assess the effect of environment policies in Europe. This database classifies the land of 39 European countries into five main categories, part of a 3-level hierarchical class system: 1. Artificial surfaces 2. Agricultural areas 3. Forest and semi-natural areas 4. Wetlands 5. Water bodies. Specific *forest and semi-natural areas* sub-classes, illustrated in Figure 3.10 as shades of green, can be related to corresponding fuel behaviour models, with non-natural areas and water bodies considered as non-burnable.

3.3.3 Building the wildfire map

The wildfire propagation simulation relies on building a fire propagation graph over a discrete environment, based on the principles described in [Finney 2002]. The representation of this environment is in the form of a wildfire map that encodes the ignition time of every cell: given the ignition time of a cell $ignition(x, y)$, the ignition time of the eight neighboring cells ($N_{(x,y)}$) is calculated following Equation 3.3.

$$\textit{ignition}(x, y) = \min_{(x_n, y_n) \in N_{(x,y)}} \{\textit{ignition}(x_n, y_n) + \textit{travel-time}((x_n, y_n), (x, y))\} \quad (3.3)$$

This method of wildfire propagation is similar to distance graphs in road networks, where each directed edge gives the *travel time* from one cell to its neighbors. In this context, the *travel time* comes from the propagation speed calculated using the fire growth model based Rothermel's rate of spread. The computation of the ignition time of all cells can be done in polynomial time using the classic Dijkstra shortest path algorithm [Dijkstra 1959]. The result is map like the one illustrated in Figure 3.11.

While fire spreads away from a particular location, fuel is consumed and the ignition disappears. Without additional fire modeling, as a simplification, it is considered that a cell ceases to be on fire when all the adjacent cells catch fire. Under such conditions, the end-of-ignition time $\textit{ignition}_{end}(x, y)$ is defined as Equation 3.4.

$$\textit{ignition}_{end}(x, y) = \max_{(x_n, y_n) \in N_{(x,y)}} \{\textit{ignition}(x_n, y_n)\} \quad (3.4)$$

Given the burning time span $[\textit{ignition}, \textit{ignition}_{end}]$ for each cell, the set of cells that make the wildfire perimeter at time t are those fulfilling the condition $t \in [\textit{ignition}, \textit{ignition}_{end}]$. In other terms, there is a set of cells forming a one-cell width closed contour polygon (an isochrone) for any given time t in the wildfire propagation graph.

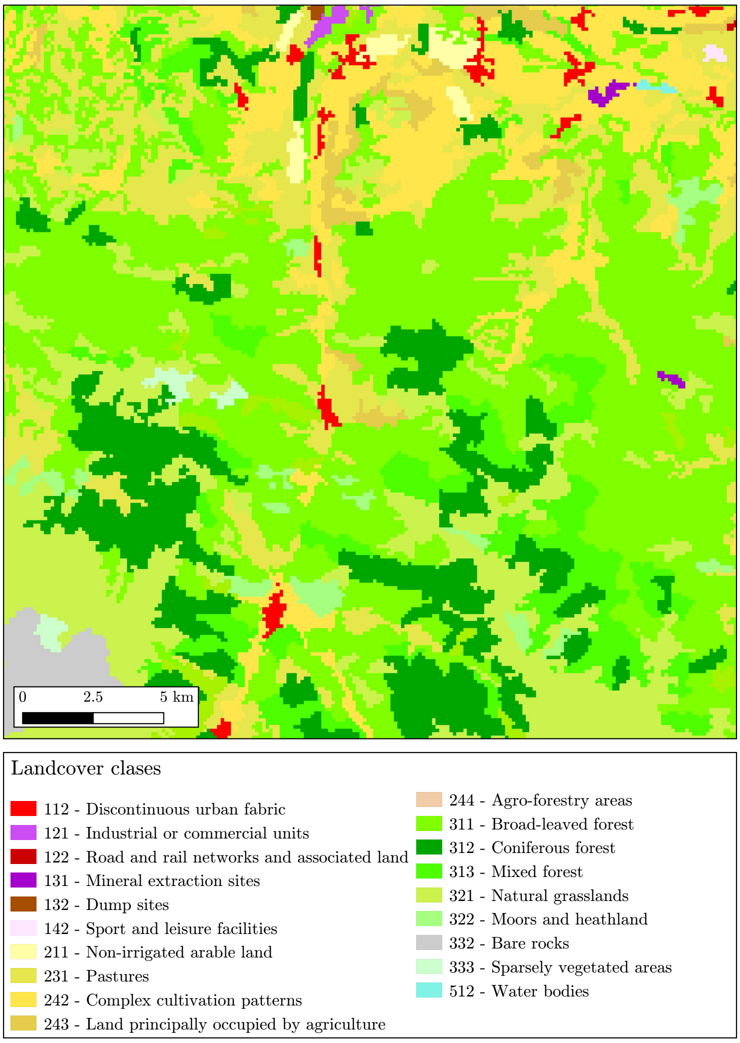


Figure 3.10: Land cover map of the region described in Figure 3.8 with colored pixels represent distinct land usage classes. Shades of green correspond to forest and other natural area classes that can be linked to fuel types for the purpose of wildfire behavior simulation. Red and purple colored cells represent artificial surfaces that are not handled by wildfire behavior models. The raster has been re-sampled from the original resolution of $100\,\mathrm{m}$ to match the resolution of the DEM.

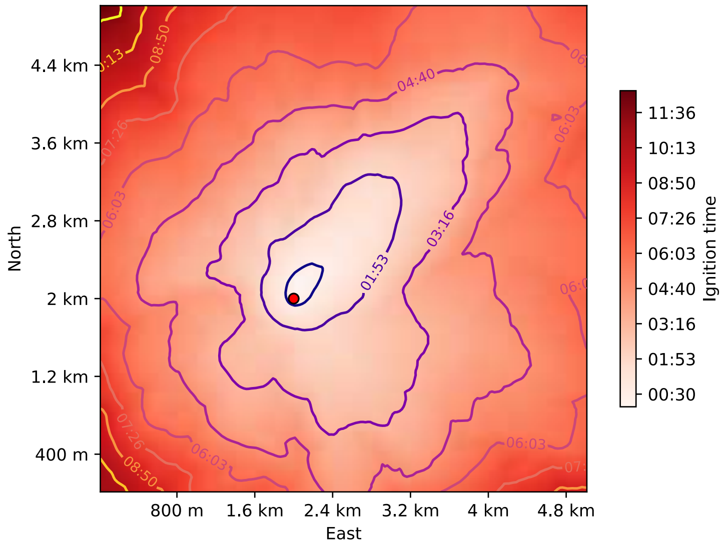


Figure 3.11: Wildfire propagation map in a 5 km by 5 km area. Shades of red denote the ignition time and fire fronts are depicted at regular time intervals.

3.3.4 Illustrations

Figure 3.9 illustrates the surface wind field of a mountainous area under different conditions. The map is encoded as a raster with the same resolution as the elevation raster used as input, which contains, for each cell, a local wind direction and speed. Due to the long computation time needed to produce local wind maps, resulting wind data is cached to the file system.

Figure 3.12, Figure 3.13, Figure 3.14 and Figure 3.15 illustrate additional wildfire propagation examples under different wind conditions and simultaneous wildfire ignitions.

3.4 Conclusion

This chapter has introduced the scientific principles of wildfire propagation, from the physical and chemical level to the main environment factors that affect its spread at the macroscopic scale: fuel, terrain and wind.

An effort has been made to synthesize the existing literature on wildfire propagation models in which two approaches coexist: physical models and empirical models. In practice, the most popular model used by firefighters combines both approaches to find a compromise between accuracy and ease of use. In this respect, the inherent limitations of wildfire modeling has been discussed, owning to propose the observation of selected areas with fleets of UAVs in a timely manner as a way to overcome propagation uncertainties.

The chapter is completed with the introduction of a wildfire simulation software based on several state-of-the-art wildfire models and realistic topographic, fuel and

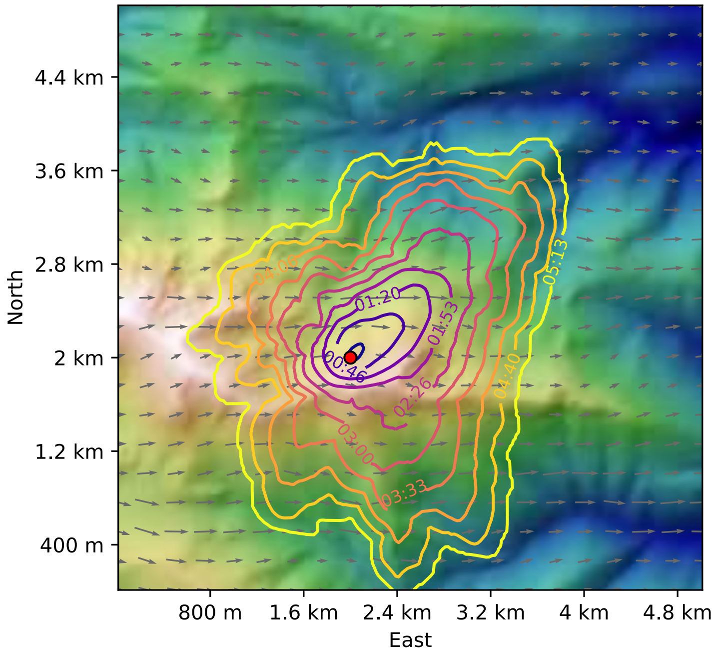


Figure 3.12: Illustration of a wildfire propagation from one ignition point. The relief is depicted in the background and the arrows represent the local wind flow.

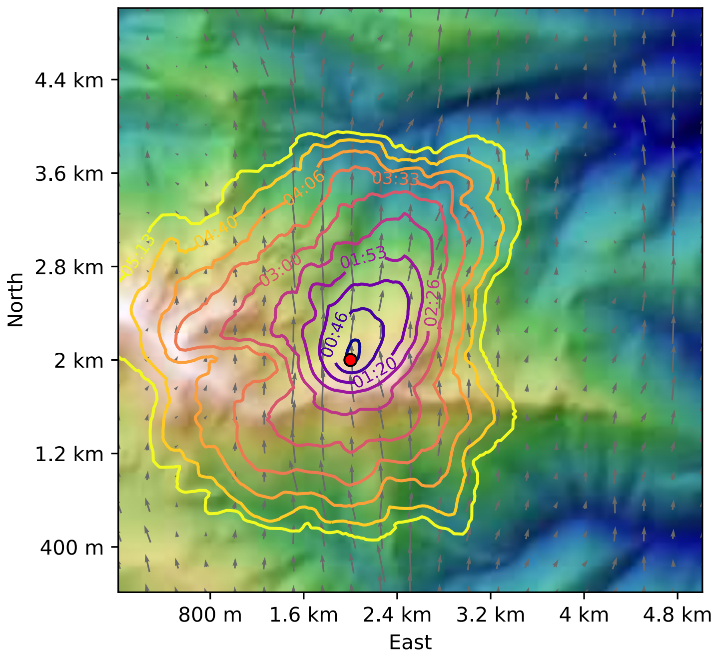


Figure 3.13: Illustration of the wildfire propagation of Figure 3.12 with different wind conditions.

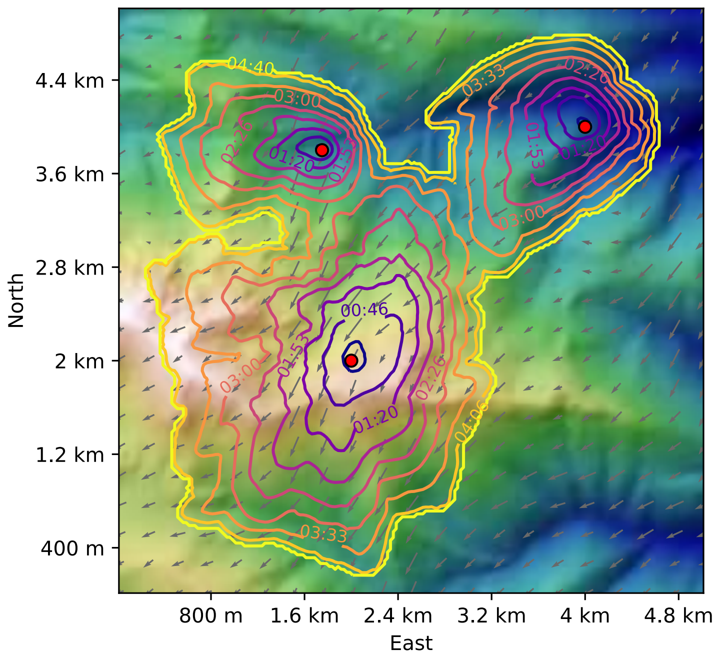


Figure 3.14: Illustration of the propagation of concomitant wildfires.

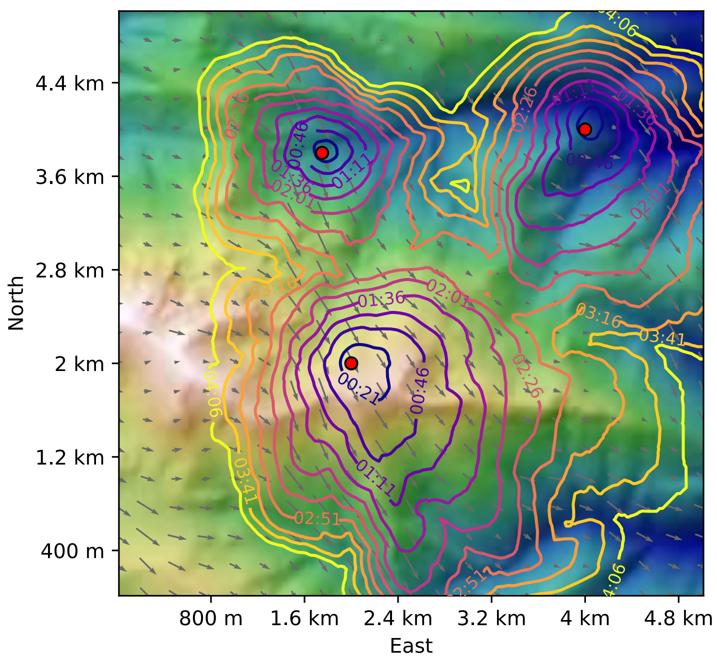


Figure 3.15: Illustration of the propagation of concomitant wildfires.

wind data. The simulator and the resulting wildfire maps are essential elements needed for the algorithms in Part III of this manuscript.

CHAPTER 4
UAV models

The goal of this chapter is to depict the two models exploited by the wildfire observation planning algorithm: A motion model to calculate fixed-wing UAV flight paths, and a perception model that assesses which pieces of land are observed by the UAVs and the utility of perceiving them. The motion model is necessary to ensure the feasibility of flight paths, even under the effect of wind, and to provide an estimate on travel time. The perception model provides a measurement on the interest of making an observation, so that the relevance of a particular trajectory can be assessed with respect to others.

4.1 UAV motion model

It is crucial for wildfire perimeter tracking to get the UAVs at the right place at the right time: if not, an early or late arrival to a desired spot will not increase system knowledge about the wildfire. Hence, it is deemed necessary to exploit a motion model during observation planning in order to predict the flight path and to estimate the time a UAV takes to travel between waypoints.

The computation of the travel time also serves as a way to assess the feasibility of the mission with respect to the UAV endurance. While the duration of a flight depends physically on the amount of energy stored in batteries, in practice, the allowed flight endurance is time-measured as constant thrust is applied to maintain a steady airspeed. This way, the operation of UAVs is simpler as one can take conservative flight duration figures that are systematically safe, independent of power consumption deviations.

UAV motion fully depends on the airframe type. While rotary-wing aircraft are able to hover and move in every direction, fixed-wing aerial vehicles have restricted motions. In particular, they have to keep always moving forward at a minimum airspeed (otherwise they would stall), and the minimum turn radius is bounded. The flight dynamics of small fixed-wing UAVs is quite complex due to intricate behavior of fluid mechanics. Nevertheless, in order to keep computational costs low, it is possible to take a simpler approach by only considering their main kinematic behavior, their state being fully described by their position, airspeed and angles of rotation (roll, pitch and yaw) as depicted in Figure 4.1.

Unlike bigger planes, the control of small UAVs is quite simple as they have few controllable flight surfaces (actuators). Airspeed is controllable under restricted margins, and attitude variations are only achieved by the means of the ailerons.

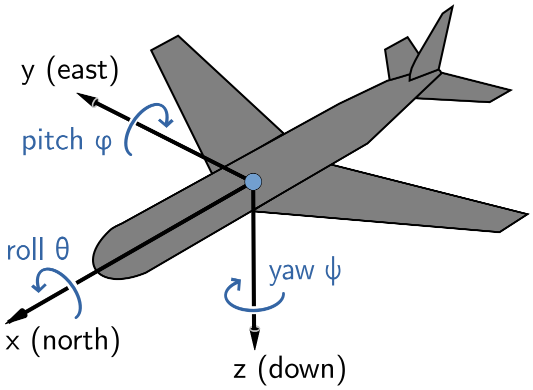


Figure 4.1: UAV reference axes and rotation angles.

Changes in altitude are accomplished by adjusting the pitch and changes of direction re obtained by the inclination of the vehicle towards the inside of the turn (banked turn). As a consequence automated flights of commercial small UAVs are composed of a sequence of basic flight primitives, restricted to waypoint navigation and loitering: every trajectory is always built upon a series of straight lines and turns.

The most important external factor that affects UAV motion is wind – as for wildfires. The primary consequence of the wind action is pushing the aircraft out of the planned trajectory as the effective flight speed varies with the aircraft orientation with respect to the wind direction.

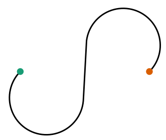
4.1.1 Computation of flat flight trajectories without wind

Let's consider a fixed-wing UAV, described by the kinematic model of Equation 4.1, that flies on a horizontal plane (x, y), at some constant altitude z, and at a constant airspeed V_a with a heading angle ψ. The change rate of the heading angle, which is the turn speed of the vehicle, is done by the means of a coordinated turn: this is making the aircraft roll to a set bank angle θ.

$$\begin{aligned} \dot{x} &= V_a \cdot cos(\psi) \\ \dot{y} &= V_a \cdot sin(\psi) \\ \dot{\psi} &= \frac{g}{V_a} \cdot tan(\theta) \end{aligned} \tag{4.1}$$

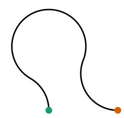
$$R = \frac{V_a^2}{g \cdot tan(\theta)} \tag{4.2}$$

The vehicle turn radius is derived from its airspeed and bank angle as depicted in Equation 4.2. The greater the bank angle is, the tighter the turn radius becomes. A reasonable simplification is to consider that the UAV can roll sufficiently fast, so the settling time is sufficiently short to be ignored. Furthermore, the value of θ is bounded by the physical limits of the aircraft (Equation 4.3).

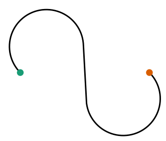


(a) LSR path.

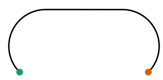
(b) LSL path



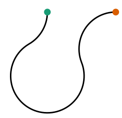
(c) LRL path



(d) RSL path



(e) RSR path



(f) RLR path

Figure 4.2: The six possible optimal standard Dubins path configurations linking two oriented points.

$$-\theta_{max} \leq \theta \leq \theta_{max} \tag{4.3}$$

It has been proved by [Dubins 1957] that the shortest path connecting two oriented points (x, y, ψ) for the vehicle defined in Equation 4.1 is built upon only 3 primitives: maximum curvature sections, where θ is set either to θ_{max} or to $-\theta_{max}$, denoted *L* and *R* for left turn and right turn respectively, and straight segments denoted *S*. This kind of path, named Dubins paths after its author, is a sequence of exactly three of those primitives. In particular, the work of Dubins shows the optimal path can only be found among one of the six following configurations: *LSR*, *LSL*, *LRL*, *RSL*, *RSR* or *RLR*. An illustration of the six configurations is shown in Figure 4.2 (note that the *RLR* and *LRL* paths only have to be considered when the start and destination points are close with respect to the UAV turn radius).

4.1.2 Computation of flat flight trajectories under constant wind

The presence of wind during flight affects the UAV as an additive disturbance, pushing the UAV out of his path in the wind direction. Adding this behavior to the regular UAV model requires to include the components of the wind vector (V_{wx}, V_{wy}) giving as a result Equation 4.4. While wind typically varies through space and time, locally it can be represented by its mean value as if it was steady.

$$\begin{aligned}\dot{x} &= V_a \cdot cos(\psi) + V_{wx} \\ \dot{y} &= V_a \cdot sin(\psi) + V_{wy} \\ \dot{\psi} &= u\end{aligned} \tag{4.4}$$

The impact of wind on the UAV produces a couple of differences with respect to the assumptions made with a basic UAV model. First, the aircraft ground speed $V = (\dot{x}, \dot{y})$ now differs from its airspeed V_a. While the airspeed remains constant, the effective ground speed depends on the UAV heading angle with respect to the wind direction. Second, the UAV flies with sideslip so the orientation of the aircraft tip is not the same as the direction it is flying to. In other words, the course of the UAV is not the same as the heading angle when flying towards a direction that is not aligned with the wind (non null slip angle).

Unlike Dubins paths, where banking left or right results in circular trajectories, the action of wind produces trochoidal paths as illustrated by Figure 4.3. Consequently, standard circular Dubins paths under the presence of wind are sub-optimal and, while UAV guidance controllers can compensate some wind disturbance in order to follow the original paths, it is more energy efficient to follow the optimal trajectories derived from the constant wind condition.

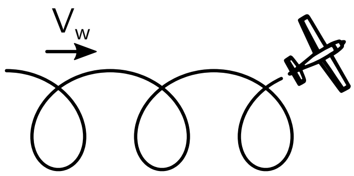
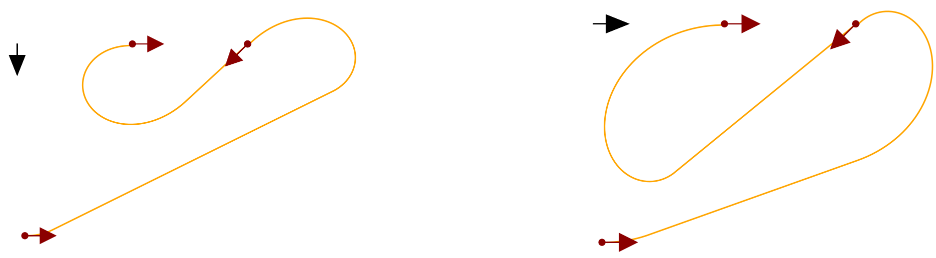


Figure 4.3: Trochoidal path performed by a UAV turning at a constant rate under the effect of a constant wind field.

The optimal path for the wind case still consists of intervals of maximum rate turns and straight lines, but the arc of circle sections are replaced by trochoidal sections whose shape depends on the wind direction (Figure 4.4). There exist a couple of publications in the literature that tackle the problem of finding time optimal paths under steady winds, that can be cast in two approaches. The authors of [Techy 2009] describe Dubins wind paths by parametric expressions, obtained analytically for the LSL and RSR cases and numerically for the others. Unlike regular Dubins paths, the parameters of some Dubins wind path configurations are found by solving a transcendental equation. This means that closed-form expressions of the solution do not exist.

A pure numerical approach is proposed by [McGee 2005]. Its strategy is to find the optimal path with wind by reformulating the problem as finding the no-wind path from a fixed position to a virtual moving destination that drifts in opposite direction to the wind vector. The goal of the redefined problem is to reach the virtual target at the right time with a regular Dubins path. When the planned no-wind path is transformed back by the action of wind, the disturbed path corresponds to the time optimal trajectory that reaches the original destination. This second



(a) Wind blowing toward the south (b) Wind blowing toward the east

Figure 4.4: Depiction of the effect of wind over Dubins paths. Depending on the wind speed and direction, the path linking the same waypoint sequence can drastically change in shape and length. Steering while facing wind results in a tighter turn radius due to the slower ground speed. Conversely, turning in the direction of wind leads to a wider turns.

approach better serves our purpose for a UAV motion model because the core of algorithm that finds the no-wind Dubins paths is reused.

4.2 Perception model

The interest of a wildfire monitoring mission is mostly supported on the actual possibility of detecting fire during a flight. So, in order to evaluate the suitability of an observation plan, it is necessary to have an educated guess of what UAVs are able to see. Based an estimation of where the wildfire perimeter is found at any time, the pose of an UAV, and the attributes of the on-board camera, a perception model is *(i)* able to provide an estimate of the portions of land that could be seen by the aircraft and the fraction of these expected to on fire, and *(ii)* able to assess the amount of information on the fire the perception will bring.

There are many perception models available of varying levels of realism but with an associated computational cost. Because the perception model is solicited very often during planning, the preferred one needs to simple and effective giving a coarse estimate of the observed terrain cells.

4.2.1 Assessing visible cells

The best way to predict which cell can be viewed from a given UAV pose is to consider the bare minimum behavior of a point-hole camera reduced to the footprint of the UAV camera over the ground. Additionally, if the camera is configured to always look down the model becomes the simplest – note that most, if not all, aerial mapping algorithmic solutions exclude the images not acquired in such conditions. By placing the UAV camera pointing nadir and by only considering the

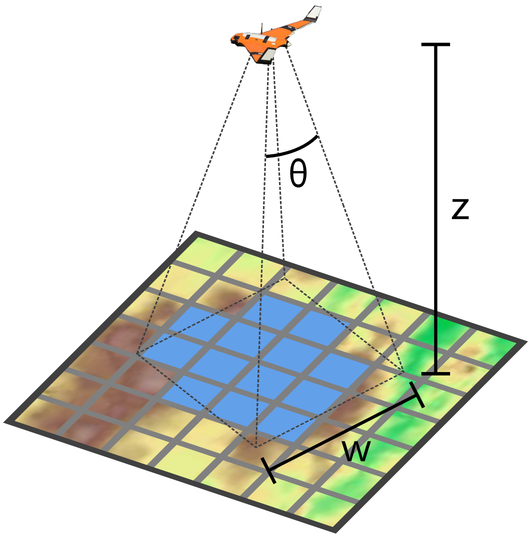


Figure 4.5: Illustration of the UAV perception model

straight segments of a flight trajectory, its footprint can be approximated to a rectangle. Figure 4.5 illustrates the camera placement configuration and the factors of the selected perception model described in Equation 4.5, where w is the length of one the field of view rectangle sides, z the flight altitude over the ground and θ the UAV camera field of view angle.

$$w = 2 \cdot z \cdot tan\left(\frac{\theta}{2}\right) \tag{4.5}$$

This particular camera arrangement is also beneficial during mission execution. As land is observed vertically, the landscape pictured by the camera corresponds to a contiguous piece of terrain where there is no need to consider occlusion issues, e.g. an area behind a hill. Moreover, the projection of an oblique view of the earth surface results on a map on uneven precision: The farthest elements pictured by the camera are depicted with less detail than those in the foreground. Again, a vertically-mounted camera, provides an advantage by making every pixel of the image the same size in earth dimensions.

The camera footprint obtained using the proposed perception model gives a list of *observed cells* that are inside the UAV camera field of view. The subset of these cells that also happen to be on fire are called *observed burning cells*. This classification is important because plan utility model considers observed burning cells primarily, as they are the only ones that convey some information about wildfire propagation in real life. Figure 4.6 illustrates a portion of a UAV flight trajectory depicting the observed cells and observed burning cells. These are only detected over straight segments of the path where the camera is vertically oriented.

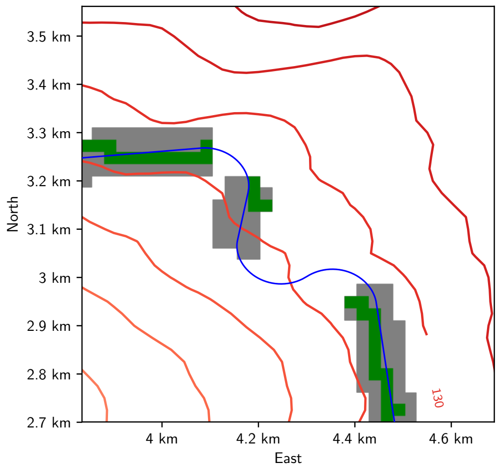


Figure 4.6: Illustration of the perception model outputs resulting from a UAV trajectory. Observed cells are depicted in gray, and observed burning cells, in green.

4.2.2 Assessing the utility of perceiving a given cell

A utility function is needed for any decision process to find the best solution regarding some feature of a problem that has to be solved. Utility is a numerical function that allows to assess the quality of a solution, rewarding the traits that are considered desirable and penalizing those that make a decision worse.

In the context of wildfire monitoring the fire front is the essential feature that has to be observed in order to be aware of its status. As discussed earlier in chapter 3, a wildfire is essentially a front propagation phenomenon with an active flame perimeter that expands away from the source of the ignition. Observations of untouched sites or already burnt locations do not give any clue concerning how near or far the fire is. Thus, active wildfire monitoring is in fact a fire perimeter tracking mission.

Deciding a utility measurement with respect to the extent and the quality of the knowledge acquired about a wildfire is far from trivial. Unlike performance ratings like flight time or distance, deciding which plan tracks a wildfire better accepts multiple interpretations, yet it is possible to find a set of general principles a utility function should follow. We consider two such principles:

- Clearly, a utility function for wildfire monitoring has to encourage leading the UAVs to the fire front. This supports the need for a wildfire spread model to provide a good estimate of where to search for this feature. As such, the primary factor to determine whether a planned observation is useful is to ensure if it is located over an active fire front cell.

- Next, we can expect some regularity locally around an observation of the fire perimeter. At medium and large scale, the fire front is regular and spreads slower than UAV flight speed. As a result, visiting two locations close to each other along the fire front do not give a lot more of additional information with respect to observing one of these two locations – not to mention it would hinder the observation of farther away locations. This also applies as the perimeter spreads out because short term propagation forecasts are not expected to diverge too much from reality, even if the underlying prediction model is not accurate. Repeatedly tracking the same spot has to be discouraged against broadly monitoring the wildfire perimeter.

Finally, one should not forget that the wildfire monitoring system is under the control of human operators that definitely "understand" the phenomenon and may task the system in a particular way. As such, the operator is part of the decision loop, providing a coarse guidance of the process, and in particular specifying areas to observe : this can be made through setting the utility of such areas.

Utility map. The manipulation of this notion of utility is made thanks to a utility map C: it is a map of the same raster structure and resolution of the fire map, for which each cell c encodes the utility of being perceived $U(c) \in \mathbb{R} \mid 0 \leq U_B(c) \leq 1$, 0 indicating the least utility and 1 the highest utility. This map is valid at a given time instant, and evolves each time the fire map is updated. There are various ways to build a utility map, depending on what one wants to encode in them. Below we present basic means to build a utility map:

- Initializing a utility map on the basis of the rate of spread. Considering that faster parts of the perimeter have to be observed much frequently than slower ones, one can define a utility map on the basis of a fire map using this definition:

$$U_B(c) = max\left(U_{min}, \frac{RoS(c) - RoS_{min}}{RoS_{max} - RoS_{min}}\right) \quad \forall c \in C \tag{4.6}$$

  in which U_{min} is defined so as to ensure that all burning cells at a given time instant are worth to be perceived.

- Encode users information. The operator can put a mask over the area he wishes the system to observe, clearing to zero the utility of all other portions of the map.

- Memory. When the fire map is updated, the utility map computed beforehand can be used to update the current utility map, thus introducing a memory of the utility as information are gathered – for instance, to prevent the observation of cells recently observed.

Exploitation of the utility map. The main purpose of the utility map is to compare the interest brought by observations made between different trajectories. The utility of a defined trajectory is simply composed of the sum of the utilities of the observed cells. When planning observation trajectories, a temporary utility map structure is computed, that encodes the expected utilities after the execution of the current plan. This is in particular not to count twice the utilities of perceiving cells during the evolution of the plan.

To account for the second principle of the utility model, the gain of utility provided by the observation of a given cell is complemented with the utility brought by the neighboring cells according to:

$$\Delta utility\left(o_f\right) = \sum_{c \in C} U_B\left(c\right) \cdot \frac{d(c, o_f) - r_{min}}{r_{max} - r_{min}} \quad \forall \{\, c \in C \mid d(c, o_f) < r_{max}\} \tag{4.7}$$

In the temporary utility map, the utility is updated accordingly.

4.3 Conclusion

This chapter has introduced the two essential models for observation planning: 1. A motion model, based upon the Dubins airplane adapted to steady wind conditions, to predict travel time under realistic conditions; 2. A simple perception model to estimate which portions of land are being observed by a trajectory and the interest of observing them.

Part III

Algorithms

CHAPTER 5

Situation assessment: Update of the wildfire status belief

Accurate information about the location of the fire perimeter and its future spread is essential for a wildfire monitoring system : first and obviously, it is the expected product of the system for the firefighters to plan the best countermeasures, second, and more related to our concern in this work, it is a basis on which the UAV planning algorithms can generate observation missions. But producing a comprehensive and up-to-date wildfire map is not trivial: due to the inability to rely on permanent and complete observations of the phenomenon, a dedicated situation assessment system is required to estimate the complete wildfire situation from limited data. This chapter proposes a simple situation assessment process to produce wildfire maps combining the local fire maps provided by the land sensors and the fleet of UAVs, and the forecasts created by the wildfire propagation model.

5.1 Introduction

A wildfire is a large scale and long-lasting event that cannot be observed in every single location at the same time, as this would require a so large number UAVs that it would not be feasible in practice. In a real situation, only a few vehicles are available to monitor the wildfire: even if observations are optimally planned, information will be lacking at various places. To build a complete representation of the current state of a wildfire, it is necessary to estimate the fire spread where data is not available.

The way the wildfire research community currently approaches this problem is through solving an inverse problem, that is finding the set of input parameters of a model whose output matches the real observations. However, as discussed in chapter 3, wildfire behavior is complex and numerical models, while being able of roughly predicting fire spread, are inherently uncertain due to unknown dynamics and large parameter sets that are difficult to measure. Recent publications [Rochoux 2013, Rios 2016] have tackled this issue by the means of techniques coming from climate and whether forecasting applications. *Data Assimilation*, and *Ensemble Kalman Filters* in particular, perform a controlled stochastic evaluation of many scenarios to optimize model outputs according to wildfire observations. Unfortunately, as of today, these tools are still matter of researches, and no out-of-the-box implementations usable in real time are available. As a result, this chapter

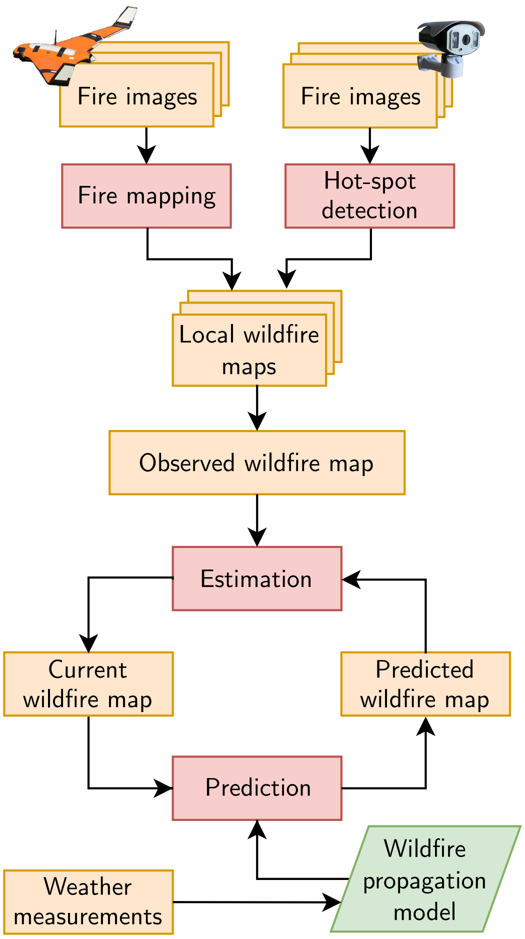


Figure 5.1: Diagram of the proposed situation assessment process.

proposes a simplistic approach that does not allow the optimization of the wildfire forward model, but is able to fuse observations with forecasts to estimate the wildfire perimeter. It does not propose a valid nor thorough model, not even a surrogate model, but it yields results that are qualitatively consistent, and sufficient for the need of defining a complete fire monitoring architecture.

The proposed situation assessment process is based on algorithms that produce an estimate of the wildfire state, current and future, from the fusion of perception data and wildfire forecasts. This process, summarized in Figure 5.1, is a continuous loop of:

1. *Estimation* of the current state by the means of fusing actual fire observations with previous forecasts. This algorithm takes the map of a predicted wildfire and, thanks to an image warping process, deforms it so that it matches real fire measurements,

2. *Prediction* of future wildfire spread based on the current estimate and the wildfire propagation model of chapter 3.

Fire observations come from two of sources, the fleet of UAVs and network of land fire sensors, in the form of *local wildfire maps* describing the detected wildfire position. In the case of observations made by the UAVs, local fire maps integrate the fire areas detected in the sequence of images taken during flight. Land fire sensors also produce the same kind of fire map on the basis of a hot-spot detection algorithm. Also, land sensors serve as weather stations, measuring the wind speed and direction that can be exploited by the wildfire propagation model.

The proposed situation assessment process is put together around a key piece of information: wildfire maps, that encode the knowledge about the wildfire propagation at a particular stage of the situation assessment process. There are three instances of fire maps that communicate information from one stage to another:

- The *observed wildfire map* aggregates the fire maps perceived by the UAVs and fire alarms. It depicts the actual knowledge about the wildfire spread at the current time since the beginning of the wildfire event. This is because wildfire maps encode the evolution of the location and time of the fire front, as described in detail in chapter 3.
- The *current wildfire map* depicts a global estimate of the complete current fire front and past propagation. The current wildfire map is the result of the wildfire estimation algorithm applied to an observed wildfire map.
- The *predicted wildfire map* describes the expected future evolution of the wildfire. Taking the current wildfire map as an input, the application of the wildfire propagation model gives the predicted wildfire map.

The following sections provide a detailed description of the suggested wildfire Situation Assessment process and describes the proposed set of algorithms to estimate the current wildfire map by fusing the predicted and observed fire maps.

5.2 Fire mapping from aerial infrared imagery

Building the map of a wildfire using a fleet of UAVs equipped with infrared cameras is a three step process:

1. The *detection* of fire in geo-referenced thermal infrared images acquired by the UAVs,
2. The *mapping* of the image pixels detected as burning over their real location on the earth surface,
3. The *update* the observed wildfire map according to the information gathered.

Land wildfire sensors are fixed infrared and visible cameras located at strategic elevated locations where most of their surroundings can be observed. Because the actual sensors are the same as those of the UAVs, the wildfire detection and

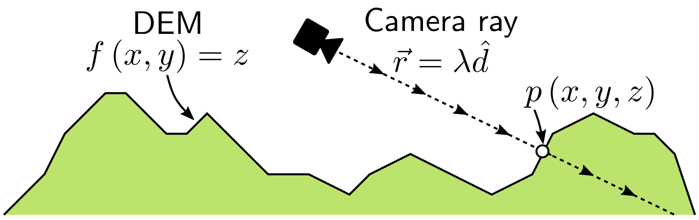


Figure 5.2: Mapping of a 2D point to a 3D earth location. It can be obtained by calculating the intersection between a ray $\vec{r}$ cast from the camera in the direction $\hat{d}$ of the pixel and the surface f defined by the Digital Elevation Map (DEM).

mapping processes described in this section are also applicable to land fire sensing. Nevertheless, smoke detection algorithms can be exploited for quicker detection that UAVs cannot benefit from.

5.2.1 Detection

Fires emits lots of energy in the form of infrared radiation, as well as the visible red color, due to their high temperature. Thermal infrared cameras sense the amount of energy in a range of infrared frequency bands and produce grayscale images where hot surfaces appear brighter than cold ones. Hence, this type of camera picture can be used to detect the presence of flames and hot-spots, detected as light pixels in the images, by applying a threshold to obtain a binary fire–no-fire array.

Fire detection is also possible with regular cameras by processing the red channel, but the reliability of this approach is reduced compared to the thermal infrared method: smoke is transparent in the infrared range but opaque in the visible hiding the flames from the camera. However, more complex algorithms, like the one depicted in [Martínez-de Dios 2005] can profit from visible imagery when available to fine-tune the threshold value in the thermal image.

5.2.2 Mapping

Images from the UAV thermal camera come with a series of metadata tagging the acquisition time and the position and orientation of the sensor at that time: it is possible with this information to estimate the actual location of image pixels over the earth surface, provided the terrain elevation model is known.

The process of taking an image is the central projection of a 3D space into a 2D surface where a position with X, Y and Z components gets reflected as a pixel of the image with two coordinates u and v. This removes depth information irreversibly, so if the inverse operation is desired – the corresponding 3D point of a 2D image – one has to find additional information conveying distance.

As illustrated by Figure 5.2, depth can be recovered with a digital elevation map (DEM) giving the elevation $f(x, y) = z$ for every location (x, y). Given the UAV camera model parameters, its position and its orientation, rays can be cast from the camera center of projection passing through every pixel in the viewing

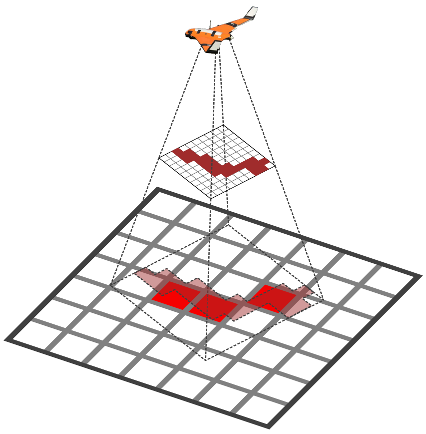


Figure 5.3: Mapping of the fire detected in a picture, taken from a UAV, to a coarser raster map. A cell is considered on fire if it is covered by a majority of pixels on fire.

direction [Collins 1998]. The point of intersection with the terrain surface p is the location of the image pixel in the real world. Repeating this procedure for every pixel of the binary infrared picture results on its projection over the DEM.

General-purpose ray tracing and intersection algorithms are typically costly, but as the terrain altitude is stored in a DEM, one can use a discrete geometric traversal algorithm [Bresenham 1965] to find the intersection point very fast.

In order to keep mapping simple, the camera is mounted on the UAV looking downwards so only images taken with the camera pointing nadir are processed. As discussed in [Lacroix 2002], this configuration prevents occlusion issues and precision disparities. Otherwise, a mountain can hide a valley, making the image projection non-contiguous or infinite if the horizon is visible.

The spatial resolution of the local observed wildfire map, which is the same as the DEM, does not match the layout of the projected image, neither in size nor in orientation. As the resolution of the wildfire map is coarser than the one of the image, multiple pixels will lay over one wildfire map cell as illustrated in Figure 5.3. Therefore, rasters are resampled following a majority policy: a wildfire map cell is tagged as on-fire if it is mostly covered by ignited pixels.

5.2.3 Fusion of local wildfire maps

The observed wildfire maps produced by the multiple perception agents depict complementary views of the same environment so to produce the global representation of the known wildfire propagation, an observation fusion policy must be considered: A cell identified as burning by a UAV or a land sensor is marked as such in the observed wildfire map with its acquisition time. If information is conflicting, when the same cell is observed burning at different times, the oldest detection time is kept.

This proposed fusion strategy assumes that observations are certain – mapping is completely accurate – but a more complex policy could be applied if significant inaccuracies are expected. For instance, if the mapping process has an error model associated to it, the probability of a cell being actually on fire can be encoded in an uncertainty grid, and only set the cell as burning in the observed wildfire map when the confidence is high enough.

5.3 Estimation of the current wildfire situation

Observations alone often do not deliver the complete vision of the wildfire situation, as with a limited number of UAVs and land sensors it is not possible to monitor every place at every time. So, with partial knowledge on how a wildfire spreads, it is necessary to estimate the wildfire map where measurements are not available. The goal of the estimation algorithm is to make an educated guess of the wildfire location through a process that combines actual observations with forecasts.

There exist data assimilation algorithms like the ones introduced in [Rochoux 2013] and [Rios 2016] that perform inverse modeling to improve the existing input parameter set of the fire propagation model. This way, previous forecasts can be corrected to match actual results, and with the improved model generated better predictions for the future. However, this procedure is far from trivial as Rothermel-like wildfire propagation models are highly non-linear and depend on numerous input parameters besides the ignition points. Performing such a kind of data assimilation technique is beyond the scope of this robotics-oriented thesis, as our focus is providing a framework for wildfire monitoring, independent of particular implementations of situation assessment algorithms.

A different approach to estimate the global wildfire map, that does not require to dive into the estimation of the wildfire model parameters, is to exploit a pure geometric strategy. In a typical scenario we can expect that the wildfire model and the input parameters are sufficiently good to model the main propagation behavior, but not perfectly. As long as propagation conditions are not extreme, wildfire spread geometry is regular and follows a circular or oval shape, with the fire perimeter growing outwards the ignition point. These conditions, translated into the ignition times depicted in wildfire maps, imply that the spread function is smooth and monotonically increasing from a minimum, which is the fire start. With this respect, the anticipated position of the fire perimeter is expected to be shifted away from the reality due to modeling errors and miscalculations of the input parameters, but the overall shape of the fire can be supposed to be roughly the same.

The proposed method to estimate the actual wildfire map is to calculate the displacement between the predicted wildfire and the observed wildfire maps, so the first can be bent in such a way it matches as best with the second, while keeping the regularity and smoothness of the fire spread.

Of course, this strategy only replaces missing information without improving the propagation model parameters. But, if the wildfire model never goes completely misaligned with respect to reality, frequent observation updates combined

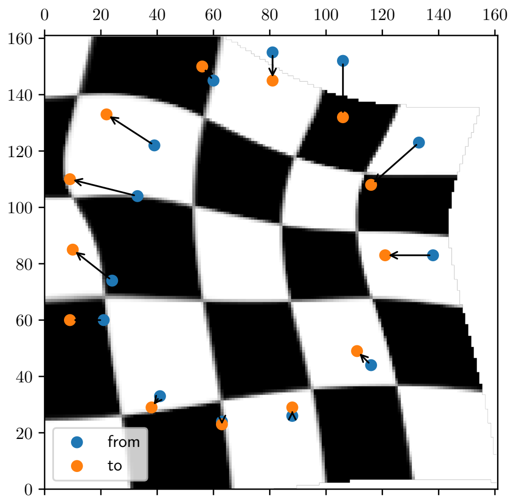


Figure 5.4: Deformation of a checkerboard pattern image using thin-plate image warping.

with regular runs of the estimation–prediction loop replace the need for very accurate wildfire models.

5.3.1 Fusion of observed and predicted wildfire maps using an image warping algorithm

Image warping is the name given to methods used to distort or deform an image from one shape to another. Typically, a set of control points in the original picture is displaced to a new position while the remaining pixels, whose displacement is not given explicitly, are also translated according to some formula that ensures the continuity of the warped image. An example of image deformation is displayed in Figure 5.4, where a checkerboard pattern image has been warped with respect to a handful of control point pairs.

This technique is very popular in image manipulation digital art, but has also proven useful in scientific image processing, in particular in the analysis of biomedical and archaeological imagery to study shape variation [Bookstein 1991][Webster 2010].

Such image warping algorithms can be used to correct the predicted wildfire spread shape assuming the forecast only has minor differences with respect to reality. The kind of fix we aim for is for short term minor differences in the input data, wind speed and direction in particular, that keep the overall shape of the fire the same but with different size and rotation.

Considering wildfire maps as a function of $\mathbb{R}^2 \to \mathbb{R}$, like an image, the objective is to define a displacement function $z : \mathbb{R}^2 \to \mathbb{R}^2$ that stretches the wildfire propagation map shape so it is coincident with the observed cells (x_o, y_o) as in Equation 5.1. The idea is to establish a mapping between the observed fire cells seen at time t, and a corresponding cell c_p in the predicted wildfire map with the same ignition time t so the function $\vec{z}(x, y) = (z_x, z_y)$ can be derived from this relationship.

$$(x_p, y_p) + \vec{z}(x_p, y_p) \rightarrow (x_o, y_o) \tag{5.1}$$

Finding c_p requires exploiting the gradient of the wildfire map that provides the direction of propagation from every cell. In other terms, this is also the propagation graph from the initial ignition cell because the fire is assumed to spread between neighboring cells. Starting from cell c_o, c_p is found by searching for a cell that has the closest ignition time to c_o along the propagation graph.

The displacement of the remaining cells in the wildfire map, those that are not coupled to an observation, has to be interpolated by a smooth function based on the known c_o and c_p displacement. Because nodes are not evenly distributed, a mesh-free interpolation algorithm is necessary. As proposed by [Bookstein 1989], radial basis function (RBF) interpolation with thin-plate splines is a suitable choice.

$$z^* \left(x, y\right) = \sum_{i=1}^{n} \lambda_i \, \phi \left(|| \left(x, y\right) - \left(x_p, y_p\right)_i || \right) \quad \left(x, y\right), \, \left(x_p, y_p\right)_i \in \mathbb{R}^2 \tag{5.2}$$

$$\phi(r) = -r^2 \ln r^2 \tag{5.3}$$

RBF interpolation (Equation 5.2) is defined as a weighted sum of radial basis functions $\phi(r)$ evaluated at the interpolation centers: in this case, a function of the thin-plate spline family has been chosen (Equation 5.3). Weights λ_i are calculated by solving a system of linear equations that results from the interpolation requirement so the relationship of Equation 5.1 is respected for z^* at every $(x_p, y_p)_i$ [Broomhead 1988].

5.3.2 Illustrations and analysis

This section illustrates the situation assessment warping procedure to estimate the current wildfire map and analyses the results of this process in selected scenarios. The purpose of this comparison is not to assess the accuracy of the algorithm, but to demonstrate its ability to reconstruct a complete wildfire perimeter from partial observations when the discordance between the predicted and real wildfires is small.

Five wildfire scenarios of one perimeter spreading across a $4\,\mathrm{km}$ by $4\,\mathrm{km}$ region of uneven terrain for four hours have been considered. In each scenario, there are differences between the wind speed and orientation used for the prediction and the values used for the observation, as listed in Table 5.1.

Figure 5.5, Figure 5.6 and Figure 5.7 show successful results for scenarios 1, 2 and 3 with small variations in the wind speed and direction after the observation of one third of the perimeter by three UAVs. Conversely, Figure 5.8 and Figure 5.9 corresponding to scenarios 4 and 5 illustrate problematic cases: scenario 4 considers

	Predicted wind		Real wind	
Scenario	Speed [km/h]	Direction [rad]	Speed [km/h]	Direction [rad]
1	6	0.75π	5	0.75π
2	5	0.66π	5	0.75π
3	6	0.66π	5	0.75π
4	6	0.66π	5	0.75π
5	6	0.25π	4	0.75π

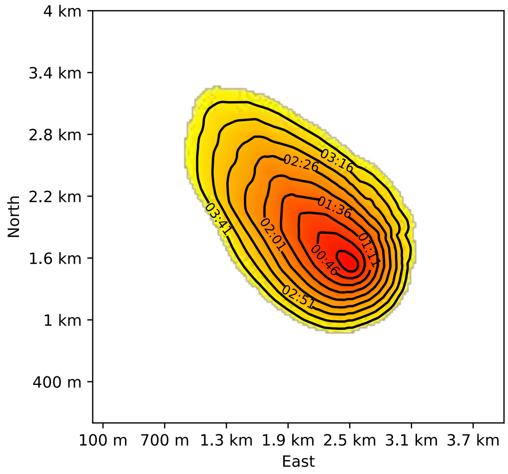
Table 5.1: Wind conditions of the situation assessment scenarios

the same environment conditions as scenario 3 and the same pattern of observed cells but at different locations. However, the resulting estimated wildfire map (Figure 5.8f) is worse than the one of scenario 3 (Figure 5.7f). This example shows that the distribution of the observations plays an important role in the accuracy of the warping algorithm.

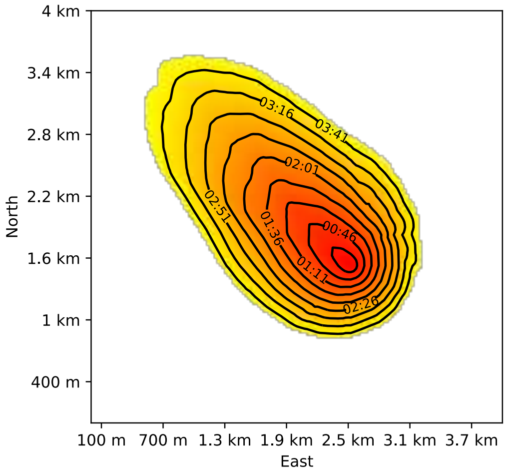
Finally, scenario 5 is an example of wildfire situation where the current situation cannot be estimated due to important differences between prediction and observation. Even if most of the perimeter has been observed, the proposed fusion approach does not give valid results as it differs too much from the real situation.

5.4 Conclusion

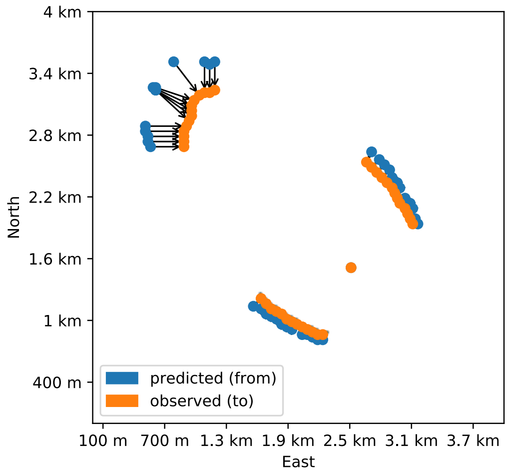
This chapter has described a wildfire situation assessment process to estimate the complete current wildfire propagation from partial observations of the fire perimeter. Simple fire detection and mapping algorithms are used to build an observed fire map from aerial and ground infrared footage. Next, an image warping algorithm is exploited to estimate the current wildfire map by modifying the expected wildfire spread so that its shape matches the observations. A short analysis of this algorithm has been provided, showing its ability to qualitatively reconstruct the real wildfire perimeter and its robustness against small discrepancies between the predicted and observed fire maps.



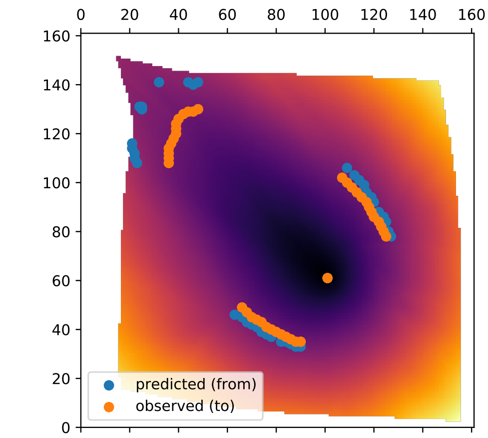
(a) Real wildfire



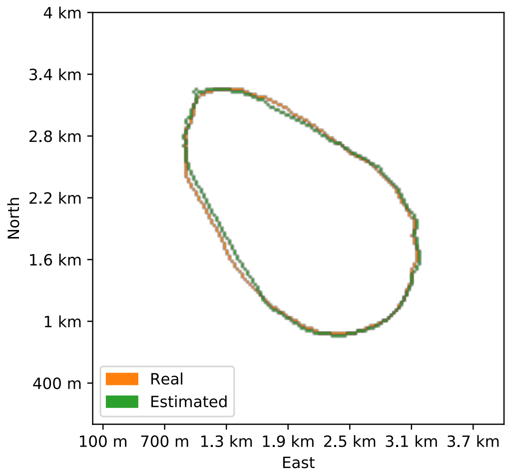
(b) Predicted wildfire



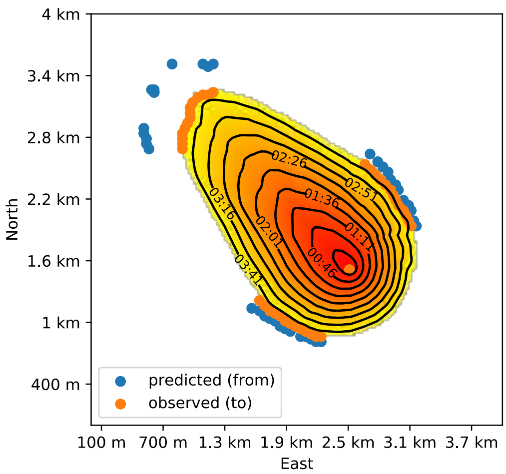
(c) Warping centers



(d) Deformation of the predicted wildfire map

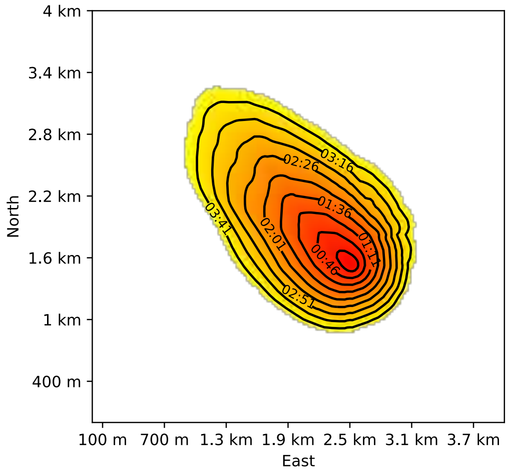


(e) Comparison between the real and estimated perimeters

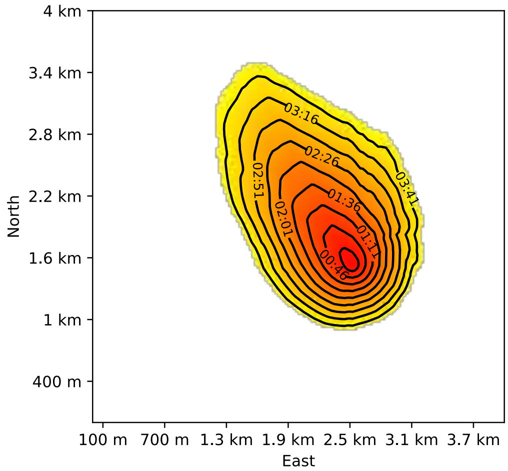


(f) Estimated wildfire map

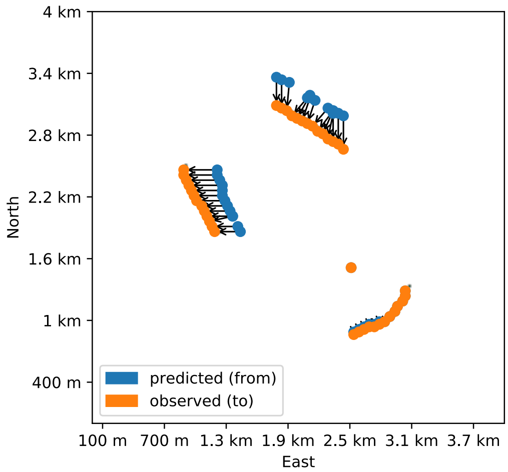
Figure 5.5: Situation assessment scenario 1



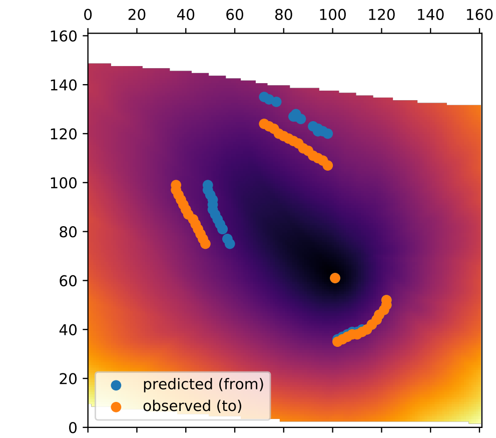
(a) Real wildfire



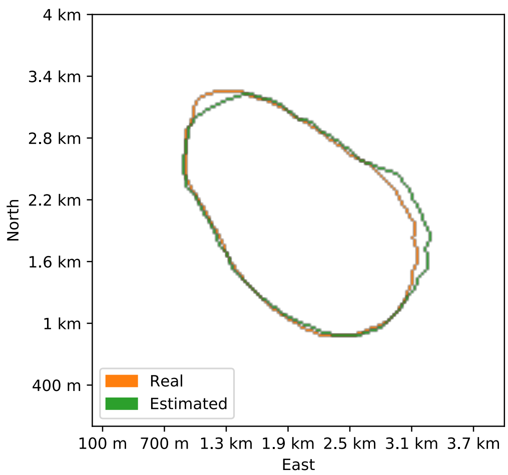
(b) Predicted wildfire



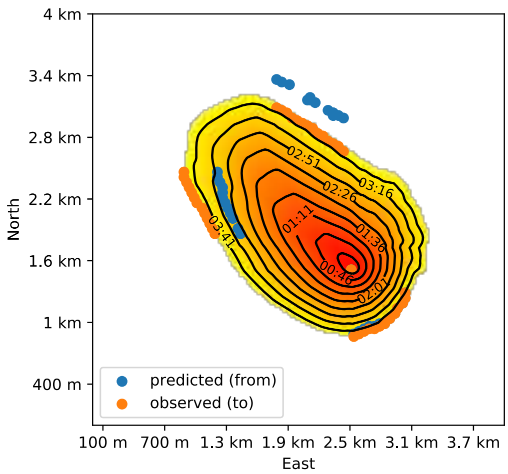
(c) Warping centers



(d) Deformation of the predicted wildfire map

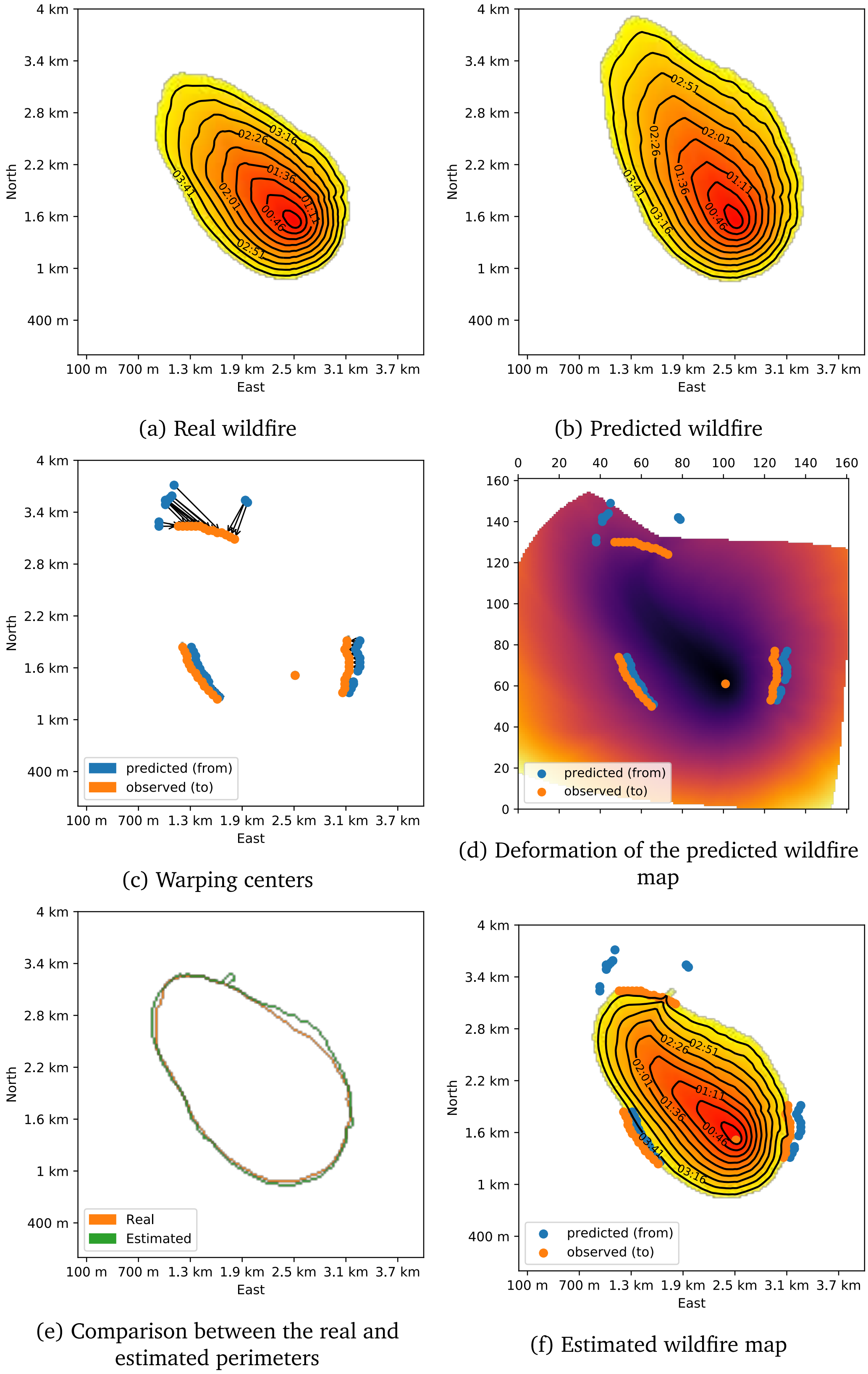


(e) Comparison between the real and estimated perimeters



(f) Estimated wildfire map

Figure 5.6: Situation assessment scenario 2



(a) Real wildfire

(b) Predicted wildfire

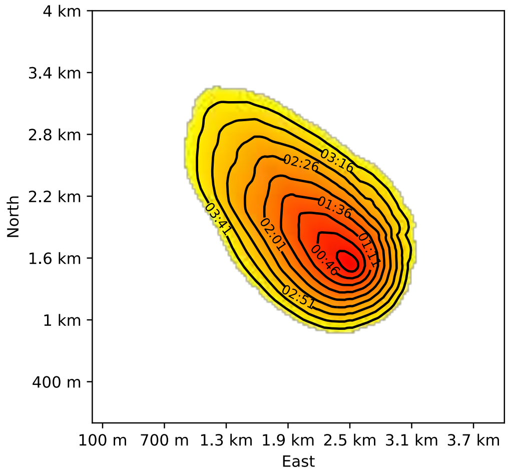
(c) Warping centers

(d) Deformation of the predicted wildfire map

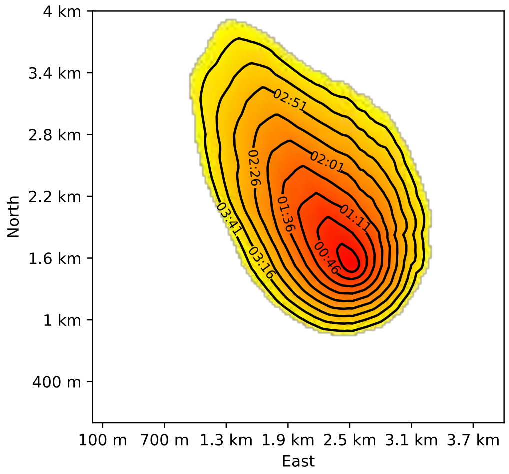
(e) Comparison between the real and estimated perimeters

(f) Estimated wildfire map

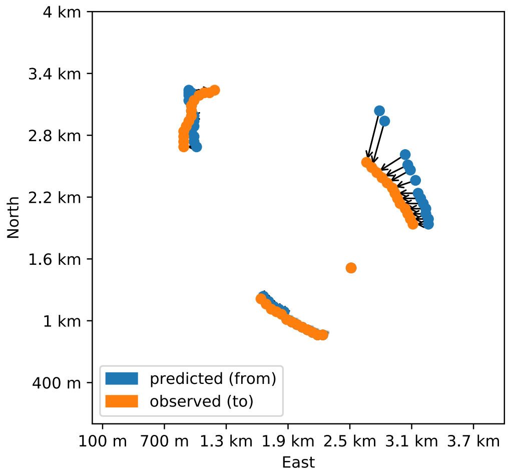
Figure 5.7: Situation assessment scenario 3



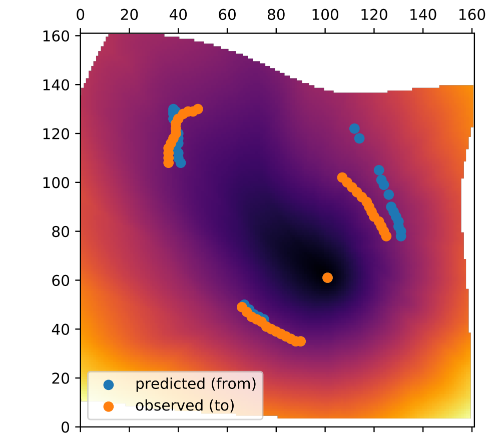
(a) Real wildfire



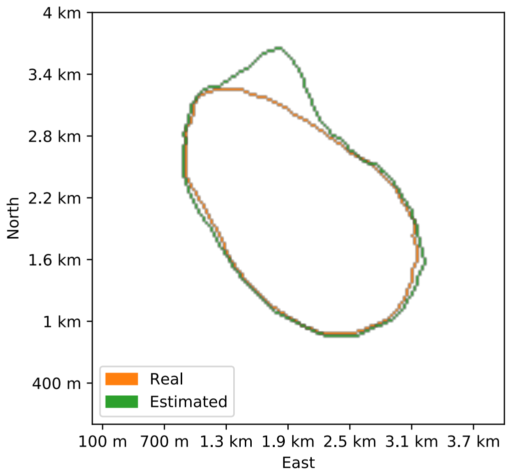
(b) Predicted wildfire



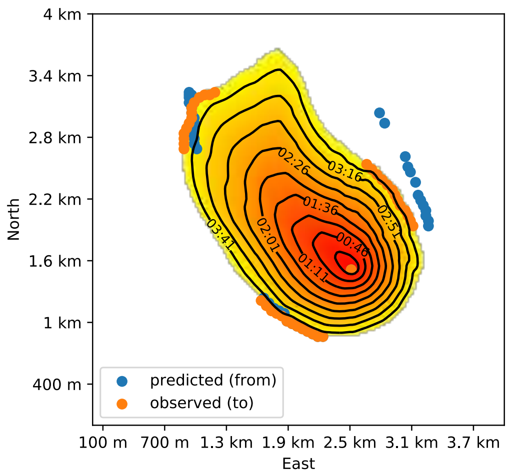
(c) Warping centers



(d) Deformation of the predicted wildfire map

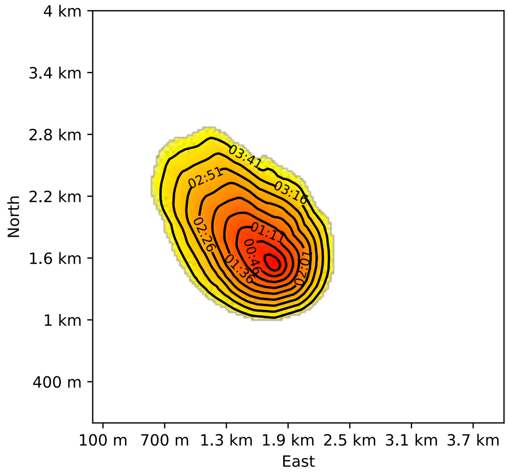


(e) Comparison between the real and estimated perimeters

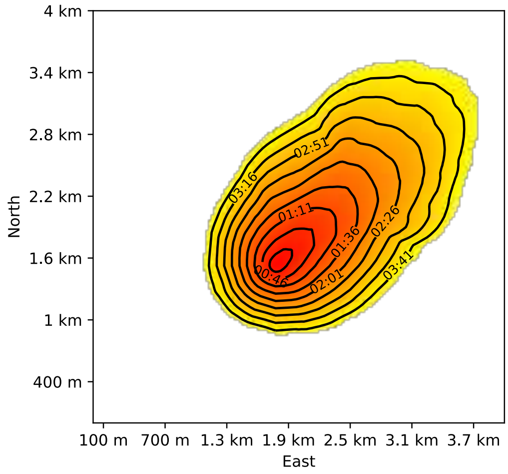


(f) Estimated wildfire map

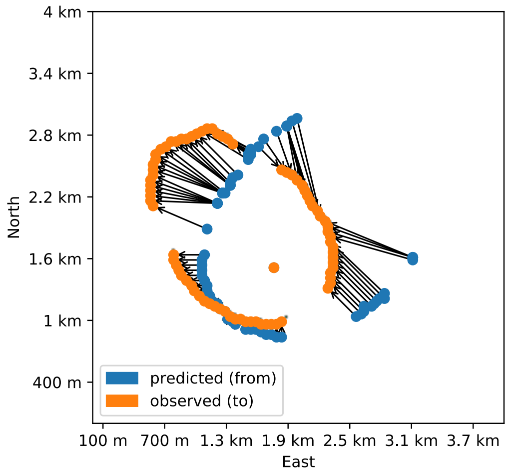
Figure 5.8: Situation assessment scenario 4



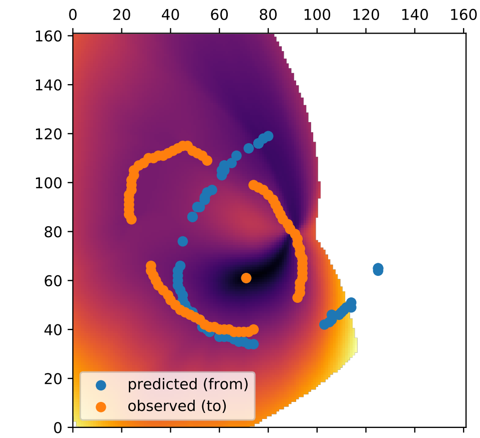
(a) Real wildfire



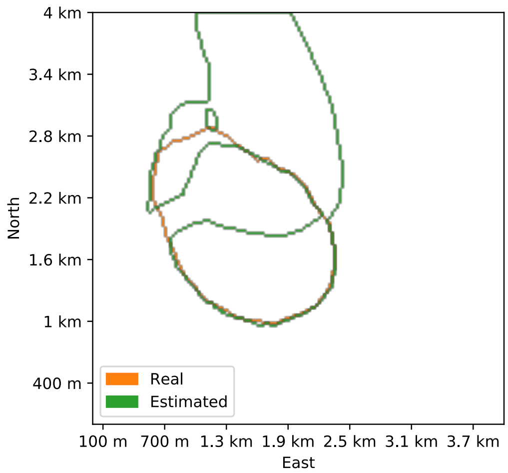
(b) Predicted wildfire



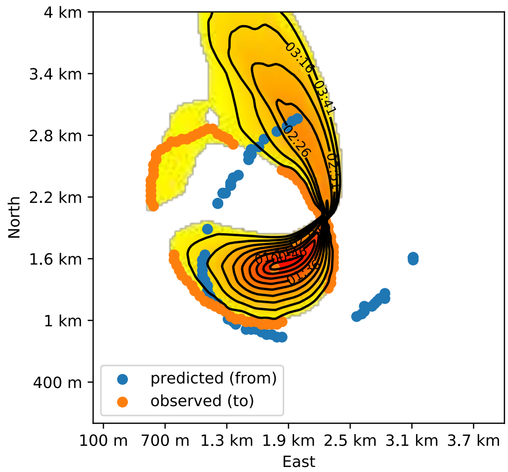
(c) Warping centers



(d) Deformation of the predicted wildfire map



(e) Comparison between the real and estimated perimeters



(f) Estimated wildfire map

Figure 5.9: Situation assessment scenario 5

CHAPTER 6

Planning algorithms for wildfire monitoring

Wildfire monitoring involve following a large perimeter which is constantly evolving. Thus, planning efficient surveillance missions for a fleet of UAVs require careful observation placement and sequencing with respect to the wildfire spread and UAV motion restrictions. Due to the many possible plan combinations, an exhaustive search of every alternative choice is not feasible and an heuristic approach must be followed. This chapter proposes a formalization of the wildfire observation problem based on the orienteering problem and introduces a planning algorithm derived from the Variable Neighbourhood Search metaheuristic, tailored to this specific problem.

6.1 Characteristics of the wildfire monitoring problem

Effective autonomous wildfire observation using a fleet of UAVs requires to plan the set of actions that the vehicles must undertake so as to maximize the amount of gathered information. The planning algorithm must efficiently exploit the UAV flying time to observe as much as possible of the fire while considering the various factors that conditions an observation mission. In particular, the number of UAVs is restricted, their capacity to move and observe the fire are limited, and the environment factors that impact the ability of UAVs to perform actions is constantly evolving. The following paragraphs introduce the various characteristics of the problem.

Wildfires are only observable at the fire front. The size of a typical wildfire is in the order of tens of km^2. Only the active fire perimeter conveys information about the fire propagation state, there is no interest in observing other burnt and unburnt fire areas[1]. Yet, the number of locations of the fire perimeter is huge, and so are all the possible trajectory combinations.

The wildfire perimeter is dynamic. Wildfires are dynamic and uncertain, and the fire front shape and size evolve with time following complex rules, as we have

[1]Observing burnt areas is also an essential task, to detect re-ignitions, but we focus on the monitoring of a live wildfire.

seen in chapter 3. Wildfire propagation models are essential for observation planning as they provide an educated guess on short term wildfire growth that can be used to guide observation actions. However, as discussed in chapter 3, propagation outcomes depend on many factors that models are hardly able to account for.

Time is crucial. The consequences for wildfire propagation to be only observable at the fire front and its dynamic nature constrains the planning process. The existence of a narrow time window to detect the active fire contour defines the cadence at which measurements should be made over time.

Observations must be prioritized. Monitoring one or multiple wildfires — multiple fronts are likely scenarios— over a large area with several of UAVs taking-off far away means that some cells will not get a chance to be observed. Information will be inevitably lacking so it is better to send UAVs in priority to locations where observations are more valuable and let the situation assessment algorithm introduced in chapter 5 estimate the remaining parts afterwards. For this purpose, the utility function the denotes the interest in observing a specific cell, introduced in chapter 4, plays a central role.

UAV motions are constrained. As we have seen in chapter 4, fixed-wing UAVs can only move forward and turn at a limited rate restricting the ability to reach a position in a closed area, which adds more complexity to the planning problem. Flight speed and time of flight constraints are a penalty on the freedom to observe the greatest cell count and the scope of these in space and time. Likewise, the effect of wind on displacement is not negligible.

All these characteristics of the wildfire observation problem impose many restrictions to an overall large set of possible solutions to explore, which prevents an exhaustive search for the optimal UAV fleet trajectories. Besides, this is a multi-objective optimization problem that depends on a balance between several arbitrary criteria and on numerous uncertain factors: finding an absolute optimal may not make much sense.

6.2 Related work

The wildfire observation problem resembles the Orienteering Problem (OP) [Golden 1987] of operations research, which is a variant of the Vehicle Routing Problem (VRP).

The Vehicle Routing family of problems, based on the Travel Salesman Problem, deals with the design of the optimal route to serve a set of clients with goods from one or multiple depots using a fleet of vehicles [Toth 2002]. The VRP is described as a graph where vertices are used to depict clients and home depots and whose arcs represent the roads that connect the different places. In the context of wildfire monitoring, terrain cells can be associated to the clients that instead of having goods delivered to them, provide information that has to be acquired.

The Orienteering Problem, inspired from the sport of orienteering, gives some score to each vertex that can be visited. The objective is to find a path visiting a subset of these vertices, such that the collected score is maximized without exceeding a limited time budget. Applied to wildfire monitoring, the Team Orienteering Problem [Chao 1996], which involves multiple agents, models the fact that UAV flight endurance is restricted and that only a fraction of the cells can be observed. Vertex scores, whose definition is linked to the utility defined in chapter 4, guides the observation planning algorithm to the preferred locations.

The OP and TOP have seen multiple extensions like the Orientering Problem with Time Windows (OPTW) [Kantor 1992], the Team Orienteering Problem with Time Windows (TOPTW) [Tricoire 2010], and the Generalized Orienteering Problem (GOP) that considers nonlinear objective functions [Wang 2008, Silberholz 2010]. The TOPTW models the opportunity to observe a particular cell only when it is burning and the GOP a non-trivial time-dependant definition of utility. However, none of the existing variants of the OP is adapted to continuous or very large space for the definition of the vertices.

6.2.1 Existing approaches solving the Orienteering Problem

Many variants of the OP and approaches to tackle them are presented in the comprehensive survey of [Vansteenwegen 2011]. As the OP is NP-hard, most successful approaches are based on known metaheuristics such as TABU search [Tang 2005, Archetti 2007], Iterated Local Search [Vansteenwegen 2009b], Genetic Algorithms [Wang 2008], Ant Colony Optimization [Montemanni 2009], and Variable Neighborhood Search [Vansteenwegen 2009a, Tricoire 2010]. Among the variety of choices, Variable Neighborhood Search stands out as one of the most effective approaches according to benchmarks [Vansteenwegen 2011] and has become the algorithm of our choice. However, none of the surveyed VNS variants fits exactly with the wildfire monitoring peculiarities. In particular, none of them was adapted to the continuous and very large space definition of the vertices. As a result, the VNS metaheuristic has to be tailored to our problem.

6.2.2 Aircraft motion models for Vehicle Routing Problems

Optimization problems such as the OP or the TSP rely on computation of the distance and time needed to travel from waypoint to another. Typically, this is the euclidean distance, but when a real situation is considered, an adequate motion model is needed. Various models can be used to describe the motion of a fixed-wing UAV with varying levels of realism depending on problem requirements. Given that the motion model is used extensively during the solution search process, a simple but effective kinematic model is necessary so the resulting plan performs according to reality without incurring in heavy computational costs.

The TSP and the OP have been extended to consider Dubins vehicles [Macharet 2014, Pěnička 2017] such as the Dubins airplane described in chapter 4. The former proposes heuristics for finding good orientations of waypoints while the latter relies on graph search to find the best orientations for a sequence of waypoints. As the time spent on trajectory computation is notable, effective path optimization heuristic approaches are needed considering the large number of position and orientation combinations possible. Regarding the Dubins airplane model with wind, some previous work can be found applied to the TSP [Anderson 2013, Luo 2018].

6.3 Problem formulation

The problem to solve is to find an observation mission plan to

> *Determine a subset of places considered on fire that can be visited by a fleet of UAVs, and in which order, so that the information gain about a wildfire is maximal and the allotted time budget is not exceeded.*

Several definitions about the way the observation plan is constructed are needed before we depict the algorithms that find solutions to this problem.

Definition 1 (Waypoint)**.** *A waypoint w is an intermediate point of the trajectory that a UAV has to reach. A waypoint is represented by a tuple (x, y, z, ψ) where x, y, correspond to East/North coordinates with respect to a reference frame, z is the flight altitude and ψ is the course angle.*

As small UAVs do not have actually so much freedom to change altitude during flight because of technical constraints, the z coordinate is considered constant at a nominal flight altitude. An incidental benefit of removing this dimension is a simpler planning process due to a reduction of the search space.

Definition 2 (Trajectory)**.** *A trajectory T is defined as a tuple (uav, t_0, W) where uav is a vehicle corresponding to one of the models depicted in chapter 4, t_0 is the start time and $W = \langle w_0, \ldots, w_n \rangle$ an ordered sequence of waypoints.*

The UAV model provides the travel time between consecutive waypoints, and combined with the trajectory start time t_0, makes possible to calculate the associated time of arrival $t(w)$ of every waypoint. In any case, to accept a waypoint into the trajectory it must be reachable by the vehicle uav. Specifically, three conditions must be respected for a trajectory to be valid:

1. The path between any pair of consecutive waypoints must be feasible by uav.

2. The last waypoint w_n must coincide with w_0 or with a safe landing spot defined by the user.

3. The time of arrival of w_n must be lower than the maximum flight endurance set for uav.

Definition 3 (Flight Window). *A flight window $F = (uav,\ T,\ d_{max},\ [t_{min}, t_{max}])$ represents the opportunity for uav to make a trajectory T and whose duration is at most d_{max}, the maximum flying time allowed by the UAV model. The flight window restricts the trajectory start and end times to the interval $[t_{min}, t_{max}]$.*

Flight windows reflect the UAV temporal allocation for the complete duration of a mission. This concept serves as a tool for the user to specify UAV fleet management requirements in the long term. For instance: a couple of UAVs can be assigned alternating flying intervals for the sake of extending the life of a monitoring mission.

Definition 4 (Plan). *A plan π is a set of trajectories $T = \{T_0, \ldots, T_m\}$, subject to their respective flight windows $F = \{F_0, \ldots, F_m\}$, and a utility U related to the expected information gain as a result of being executed.*

A plan π is valid only if every $Trajectory$ and F are valid.

Definition 5 (Utility). *The utility of a plan $U(\pi)$ is a figure of merit of π denoting the expected information gain on the wildfire. $U(\pi)$ is represented as a non-negative real number with decreasing values indicating expanding knowledge.*

This definition of utility corresponds to the one introduced in chapter 4, based on the observation model also depicted in the same chapter.

6.4 Variable Neighborhood Search planner

The observation planning we introduce in this section is based on the VNS metaheuristic [Hansen 2001], a local search approach that has been applied to numerous combinatorial optimization problems in Operations Research [Hansen 2010].

6.4.1 Basic VNS

VNS algorithms are built on a sequence of neighborhoods, where each neighborhood defines a local modification to a plan, typically aimed to improve some specific aspect of it. When applied to existing plans, a neighborhood generates close related plans, neighbors of the original plan. For instance, swapping the order at which two locations are visited when searching for the optimal sequence, or changing the orientation of a waypoint so travel time is reduced.

Algorithm 1 illustrates the way a basic VNS algorithms works: Given an initial trivial plan and a set of neighborhoods the procedure consists on repeating sequentially: 1. A perturbation phase, 2. A descent phase, 3. A change of neighborhood. The descent phase applies some local search exploiting all neighborhoods sequentially to incrementally improve the current plan until no more progress can be made. Then, the perturbation phase aims at escaping local optimums reached during the descent phase, by shaking the current best plan systematically or randomly. Finally, a decision is made on whether to continue the search with the current neighborhood or to move to the next one, based on whether a meaningful improvement has been made with the current one or not. The stop condition may be a maximum run time, a maximum number of iterations or a stabilisation of the improvement rate.

Algorithm 1 Basic Variable Neighborhood Search algorithm

```
function BasicVNS(π, ⟨N_1, ..., N_m⟩, stop-condition)
    while not stop-condition do
        i ← 1
        while i ≤ m do
            π′ ← shuffle(π, i)
            π″ ← local-search(π′, i)
            if π″ better than π then
                π ← π″                          ▷ Move to new plan
                i ← 1
            else
                i ← i + 1                       ▷ Switch neighborhood
            end if
        end while
    end while
    return π
end function
```

The key benefits of this metaheuristic reside on its generic and adaptable definition. The VNS algorithm can be tailored to a specific problem by changing how the descent and perturbation phases behave and the sequence at which neighborhoods are explored. The challenge of a VNS approach to solve a given problem resides in the formulation of the problem and in the definition of a good set of neighborhoods for solving it in reasonable time.

In particular, the advantage of using a VNS algorithm is that observation plans are built for the fleet of UAVs as a whole: the problem of allocating UAVs to areas to observe is implicitly solved with careful neighborhood design. Also, as a VNS algorithm works by applying small incremental improvements to a plan, it can be stopped at any time or restarted from an existing plan. The later is especially interesting, because plans can be repaired and improved over time as wildfire forecasts are updated.

Another benefit of a heuristic approach is that good solutions can be rapidly found. Because plans will be invalidated frequently due to evolving fire conditions and the ability of small UAVs to precisely follow a path are limited, planned monitoring missions do not need to be perfect.

6.4.2 Improved algorithm

The basic VNS algorithm takes a random plan from a neighborhood at a time and uses it in the local search to improve it until an optimum is reached. Then, if an improvement has been made over the best plan so far, the current plan becomes the current solution and the process continues with the same neighborhood or with the next one otherwise.

Unfortunately, this strategy is not suitable for wildfire observation planning because a descent phase consisting on finding a local optimum is not feasible. Improving the current solution means adding, removing and orienting waypoints, but the search space is very large. Furthermore, there is no deterministic optimization strategy to follow —no clear direction of descent— and an exhaustive search would be very time-consuming. Instead, our descent strategy relies on sampling to produce a representative set of small local optimizations within the current neighborhood and frequent neighborhood changes.

Our neighborhood definition differs from the usual one by the introduction of the utility function $u_{\mathcal{N}}$ which is local to $\mathcal{N}$:

Definition 6 (Neighborhood). *A neighborhood $\mathcal{N}$ defines for each valid plan π a set of neighbor plans $\mathcal{N}(\pi) \subseteq \Pi$ where Π is the set of valid plans.*

A neighborhood is associated with a utility function $U_{\mathcal{N}} : \pi \to \mathbb{R}$ giving the utility of a given plan in the context of this neighborhood.

The local $u_{\mathcal{N}}$ is a particular improvement measurement for a specific $\mathcal{N}$ and maybe unrelated to the plan utility function. For instance, a neighborhood aiming at optimizing trajectories could base its utility function only on the length of the plan. This neighborhood-dependent utility allows greater separation of concerns between different specialized neighborhoods, while the full problem remains mono-objective.

Our descent phase makes use of the function *gen-neighbor*:

Definition 7 (*gen-neighbor*). *Given a plan $\pi \in \Pi$ and a neighborhood $\mathcal{N}$, the function gen-neighbor$_{\mathcal{N}}(\pi)$ returns either (i) a new valid plan $\pi' \in \mathcal{N}(\pi)$ such that $u_{\mathcal{N}}(\pi') > u_{\mathcal{N}}(\pi)$, or (ii) nil if the neighborhood failed to generate an improved neighbor.*

It selects the best plan with respect to the neighborhood utility function among a set random samples in the neighborhood of the current plan.

In order to escape from local optima obtained in the descent phase a shuffling function is introduced:

Definition 8 (Shuffling). *A shuffling function $f(\pi, k) : \pi \times \mathbb{N} \to \pi$ produces a new plan by perturbing the plan π.*

The strategy applied by this function is described in Algorithm 2 and it consists in removing a sequence of waypoints from the current plan. The number of waypoints to be removed in each trajectory is randomly chosen between 0 and the maximum number of waypoints that can be removed, excluding the imposed start and end of the trajectory.

Algorithm 2 The shuffling algorithm pseudo-code to remove a random portion of every trajectory

```
function SHUFFLE(π)
    for all T ∈ π do
        R ← {w_1, ..., w_{n−1}} ∈ T            ▷ The set of removable waypoints in T
        s ← RAND(0, |R|)
        e ← RAND(s, |R|)
        for w_i ∈ {w_s, ..., w_e} do
            REMOVE(T, w_i)
        end for
    end for
    return π
end function
```

The algorithm Our VNS approach, depicted in Algorithm 3, takes as parameters an initial plan, a sequence of neighborhoods, a shuffling function and a maximum run time.

Given an initial —possibly empty— plan $\pi_{initial}$, the descent phase of VNS tries to generate plan improvements by systematically and sequentially trying all neighborhoods $\langle \mathcal{N}_1, \ldots, \mathcal{N}_m \rangle$ with *gen-neighbor* until a neighborhood $\mathcal{N}_i$ provides an improvement. If an enhancement according to the plan global utility function is provided, the current plan is updated and the process restarts from the first neighborhood $\mathcal{N}_1$. When no neighborhood is able to generate an improvement, the best plan found so far is perturbed by the shuffling function and the descent phase restarts from the first neighborhood $\mathcal{N}_1$. This process is repeated until the total *runtime* goes over the allowed budget CPU_{max}, at which point the best plan found is returned.

As the VNS approach is able to start from any valid plan, $\pi_{initial}$ can be set to a previously computed plan π_{prev}. In this case, the VNS algorithm is constrained to improve only future parts of π_{prev}. First, an initialization function translates the future waypoints to locations expected to be on fire, with waypoints that can not be translated removed from the plan. Then, the current plan is refined following the same procedure for initial plans.

6.4.3 Definition of the neighborhoods

We define two classes of neighborhoods that have proved useful in our setting:

- An insertion neighborhood

Algorithm 3 Pseudo-code of the proposed Variable Neighborhood Search (VNS) algorithm. VNS takes as parameters an initial plan $\pi_{initial}$, a sequence of neighborhoods $\langle \mathcal{N}_1, \ldots, \mathcal{N}_m \rangle$, a real CPU_{max} indicating the maximum planning time and a function *shuffle* that is applied to the best plan on a restart.

```
function VNS(π_initial, ⟨N_1, ..., N_m⟩, CPU_max, shuffle)
    initialization(π_initial)
    π_best ← π_initial
    num-restarts ← 0
    while runtime ≤ CPU_max do
        π ← shuffle(π_best, num-restarts)
        i ← 1                                  ▷ Select the first neighborhood
        while i ≤ m do
            π′ ← gen-neighbor_{N_i}(π)
            if π′ ≠ nil then
                π ← π′                         ▷ Update current plan
                if U(π) > U(π_best) then
                    π_best ← π
                end if
                i ← 1                          ▷ Switch back to the first neighborhood
            else
                i ← i + 1                      ▷ Switch to the next neighborhood
            end if
        end while
        num-restarts ← num-restarts + 1
    end while
    return π_best
end function
```

- A path optimization neighborhood

This choice is inspired from the observation planning principle of seeking to insert waypoints in trajectories to improve the utility of plan, while reducing the duration of those trajectories so more observations can be made.

A plan is relevant only for the predicted wildfire it has been computed for: this requires that trajectory waypoints lay over expected burning cells at the time of visit. During the execution of the VNS planner, one must ensure that the selected waypoints bring as much utility as possible. Hence, during the random selection of waypoints, a function that modifies the waypoints to match these criteria is required: This function *ProjectFF* (which stands for “project on fire front”) is presented below, before the depiction of the two neighborhoods.

6.4.3.1 *ProjectFF* function

Given the propagation graph of fire map, *ProjectFF* migrates waypoints along the spread direction to a suitable location where the time of arrival corresponds to the time interval within which the cell below is ignited. In other words, *ProjectFF* is a recursive function that takes a waypoint w_i of a trajectory and returns a waypoint w'_i such that $t(w'_i) \in [\textit{ignition}^{w'_i}, \textit{ignition}^{w'_i}_{end}]$, that is, a waypoint w'_i which is on the fire front when arriving at time t. Essentially, thanks to the *projectFF* function, any waypoint chosen randomly can be put in a viable position for the insertion trial. Algorithm 4 presents this function, and figure 6.1 illustrates its use in the context of a waypoint insertion.

Algorithm 4 Pseudo-code of the *projectFF* function. The algorithm relocates a given waypoint w with time t_w over a cell where the wildfire is active.

```
function PROJECTFF(fire-map, fire-map_end, w, uav, t_w)
    if t ∈ [fire-map[cell(w)], fire-map_end[cell(w)] ) then
        return w
    else
        α ← ∠∇fire-map[cell(w)]                          ▷ Direction of propagation
        d_x, d_y ← cos(α) · cell-width, sin(α) · cell-height     ▷ One fire-map cell
displacement
        if t_w > fire-map_end[cell(w)] then
            w' ← w + (d_x, d_y)              ▷ Move towards the propagation direction
            if ∄fire-map[cell(w')] ∨ fire-map[cell(w)] > fire-map[cell(w')] then
                return w                        ▷ Local maximum (Two colliding fronts)
            end if
        else
            w' ← w − (d_x, d_y)                                   ▷ Move backwards
            if ∄fire-map[cell(w')] ∨ fire-map[cell(w)] < fire-map[cell(w')] then
                return w                          ▷ Local minimum (An ignition source)
            end if
        end if
        return PROJECTFF(fire-map, fire-map_end, w', uav, t_w)
    end if
end function
```

This function is useful in various stages of the planning process:

- Waypoint insertion neighborhood:
    - To place a new waypoint in a valid location, and to correct the placement of other waypoints as a result adding or removing a waypoint in a trajectory.
    - When a waypoint is inserted into a closed trajectory, its length is extended and travel times to this location and these coming afterwards so are too. *ProjectFF* corrects their position with respect to the fire map in order to account for this delay. Essentially, thanks to this function, any waypoint chosen randomly can be moved to a valid position.

- Also, after an update of the wildfire situation, the observation plan can be refined instead of starting from scratch. In such situations, the *projectFF* function is used to quickly fix the original plan so refinement is possible.

6.4.3.2 Insertion neighborhood

A *waypoint insertion* neighborhood alters a plan by inserting a new waypoint w' in a trajectory T.

In order to fulfill the ultimate objective of a wildfire monitoring mission, that is observe the fire front at the right time, trajectory waypoints are forced to lay over the expected fire perimeter. This restriction imposes a couple of challenges for the waypoint insertion neighborhood: first, the choice of a waypoint location imposes a reduced arrival time range. Because the fire front is expanding at pace, an unwise spatial choice may force a UAV to wait or to travel at impossible speeds to reach a waypoint. Second, if a waypoint should be inserted in a trajectory, its arrival time must be scheduled to be between the previous and the next waypoint. Furthermore, inserting a waypoint typically results in a longer path, delaying the arrival time to subsequent waypoints. Therefore, the *ProjectFF* function is used extensively during the insertion process to ensure plan validity at all times.

Since the number of potentially valid waypoints is large, and finding a complete enumeration of them is unfeasible, new waypoints are obtained through sampling. As a result, the insertion procedure selects a location at random and tries to find the best insertion order in a trajectory and across trajectories. Here, the candidate waypoint goes by *ProjectFF* at every different trial so the UAV is able to reach it within the associated $[\textit{ignition}, \textit{ignition}_{end}]$ range of the underlying cell.

Given a random waypoint w and a trajectory T with waypoints $\langle w_0, \ldots, w_n \rangle$, we construct a neighbor for each $i \in [0, n-1]$ by *(i)*: inserting after w_i the w' resulting from the use of *projectFF* to place w in a valid location (feasible by the UAV and situated over an active fire perimeter when reached) and *(ii)*: for each $j \in [i+1, n-1]$, moving w_{j+1} to a suitable location with the help of *projectFF*. In a nutshell, this inserts a new waypoint in the trajectory and then updates all subsequent waypoints to make sure they are still over the fire front. This procedure is illustrated in Figure 6.1.

The utility of a neighbor plan is assessed by the global plan utility function, with ties broken by trajectory duration.

6.4.3.3 Path optimization neighborhood

In order to observe as much of the wildfire as possible in limited time, UAV flight path have to optimized, removing useless maneuvers that take precious flight time without giving back meaningful observations. Waypoint orientation has an important effect on Dubins path length and a bad adjustment that can greatly extend a trajectory with pointless turns. As illustrated in Figure 6.2, a path optimization

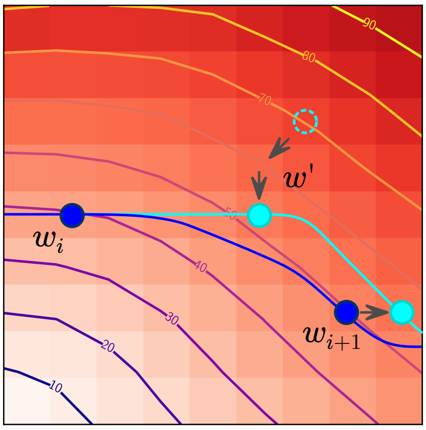


Figure 6.1: The waypoint insertion process. A random chosen waypoint w' (dashed, light blue) is inserted in a trajectory between w_i and w_{i+1} (dark blue). w' is re-projected into a previous isochrone (light blue) whose time corresponds to the time needed to reach it from w_i. Finally, due to the increment in travel time between w_i and w_{i+1}, w_{i+1} is moved to a later isochrone.

neighborhood is necessary to reduce flight time, so more waypoints could be added within the limited flight duration.

A local path optimization neighborhood applies a stochastic or deterministic rotation to a randomly chosen waypoint in the plan with the objective of reducing the duration of a trajectory. Systematic path length optimization of all waypoints through an iterative optimization method is not feasible because the problem is non-linear and has too many variables. Repeated computation of Dubins paths is expensive and reducing the length of a piece of trajectory by the means of rotating the origin and destination waypoints affect the previous and next portions the trajectory.

The path optimization neighborhood works by, picking a random modifiable waypoint w_i —different from the take-off and landing spots w_0 and w_n— from a trajectory $T_i \in T$. Then, an orientation changer generator, selected at randomly or arbitrarily, proposes a new orientation angle for this waypoint and the travel time from and to this waypoint is computed again. If the candidate reduces the length of the trajectory, then the move is accepted. Otherwise, the operation is repeated until a move that improves trajectory length is found or a maximum number of trials is reached.

Three orientation change generators have been found useful for this problem:

- *Random rotation*,
- *Flip*,
- and *Mean angle*

The first just applies a random rotation to the waypoint and the second flips the direction of the waypoint. The third generator is inspired from [Macharet 2014]

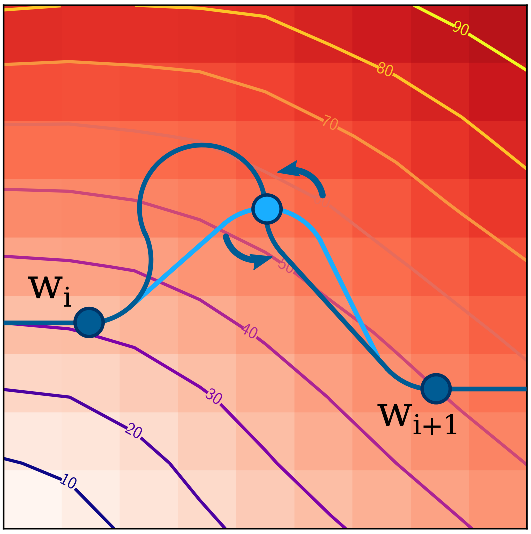


Figure 6.2: Illustration of the local path optimization process. The dark blue curve represents the original trajectory and the light blue curve the optimized one. The improved path contains a longer straight line sections so more portions of the wildfire can be observed.

and smooths the trajectory by giving a waypoint w_i the mean angle between w_{i+1} and w_i. The two last heuristics are particularly useful to fix badly placed waypoints that may be oriented against the overall trajectory direction forcing UAVs to perform pointless sharp turns. Figure 6.2 clearly shows how this situation can be solved.

While the main purpose of waypoint rotation is not to improve the global plan utility value, well oriented waypoints typically yield improvements to the observation utility as the optimized path contains larger straight flight sections.

6.5 Illustrations and analysis

This section is devoted to provide some insight on the observation planner actual implementation and to illustrate its application over typical wildfire scenarios.

6.5.1 VNS implementation details

A successful adaptation of the VNS approach to a particular problem requires an overall well-designed neighborhood generation strategy because particular neighborhood implementations have a significant impact on the quality of the resulting plans. The search for the best configuration would require an exhaustive analysis of the complete parameter space to find the ones that are better fitted to solve a particular problem. Still, the possible wildfire observation scenarios are diverse: the number of wildfires, their location and their size is variable and the UAVs may be taking off from different places. Given this broad number of circumstances, the best approach is to perform a general study over a selected number of typical situations and VNS configurations.

Previous sections introduced a couple of neighborhood structures that are useful for solving the wildfire observation problem. However, specific details about their implementation were not provided. In particular, the order at which the neighborhoods are called by the VNS planner, the total planning time and the sampling parameters. This subsection summarizes the results of [Bit-Monnot 2018] where several neighborhood configurations were evaluated on a previous version of the VNS planner introduced in this chapter. While improvements have been made since, especially on the definition of observation, the findings about the configurations are still valid.

The best sequence of neighborhood structures so far is composed of one instance of the path optimization neighborhood $\mathcal{N}_{dub}$ and three instances of the insertion neighborhood $\mathcal{N}_{ins}$:

$$\langle \mathcal{N}_{dub}, \mathcal{N}_{ins}^{all-best}, \mathcal{N}_{ins}^{1-best}, \mathcal{N}_{ins}^{rand} \rangle$$

Both the $\mathcal{N}_{dub}$ and $\mathcal{N}_{ins}$ classes of neighborhood rely upon sampling in order to propose improvements to the current plan. There is a trade off on the number of samples as the more cases are assessed the better the improvements are but, less plan enhancements are produced in the same time period. $\mathcal{N}_{dub}$ is configured to try 100 orientation-change samples for the illustrative examples in this section. The $\mathcal{N}_{ins}^{all-best}$, $\mathcal{N}_{ins}^{1-best}$ and $\mathcal{N}_{ins}^{rand}$ are set to 50, 200 and 200 waypoint-insertion samples respectively.

The difference between the three variations of $\mathcal{N}_{ins}$ resides in the specific strategy random waypoints are inserted into plan trajectories to be proposed as improvements. The proposed configuration aims at combining the strengths of various approaches:

1. $\mathcal{N}_{ins}^{all-best}$ systematically tries all possible insertion locations of every trajectory of the plan. As a result, this neighborhood favors quality at the expense of diversity.

2. $\mathcal{N}_{ins}^{1-best}$ picks one trajectory at random and finds the best place to insert the waypoint .

3. $\mathcal{N}_{ins}^{rand}$ takes the random waypoint and tries to insert it in a random position of a random trajectory.

$\mathcal{N}_{ins}^{all-best}$ allows to quickly build an initial solution by promoting high quality of neighbor plans. However, once $\mathcal{N}_{ins}^{all-best}$ fails to generate improved solutions, the planner falls back to the more diverse $\mathcal{N}_{ins}^{1-best}$ and $\mathcal{N}_{ins}^{rand}$ neighborhoods.

The effect of the $\mathcal{N}_{dub}$, located first in the neighborhood sequence is the reduction of trajectory duration so more waypoints can be inserted afterwards. A secondary benefit of $\mathcal{N}_{dub}$ is the production of trajectories that are smooth and subjectively feel more natural to a human operator.

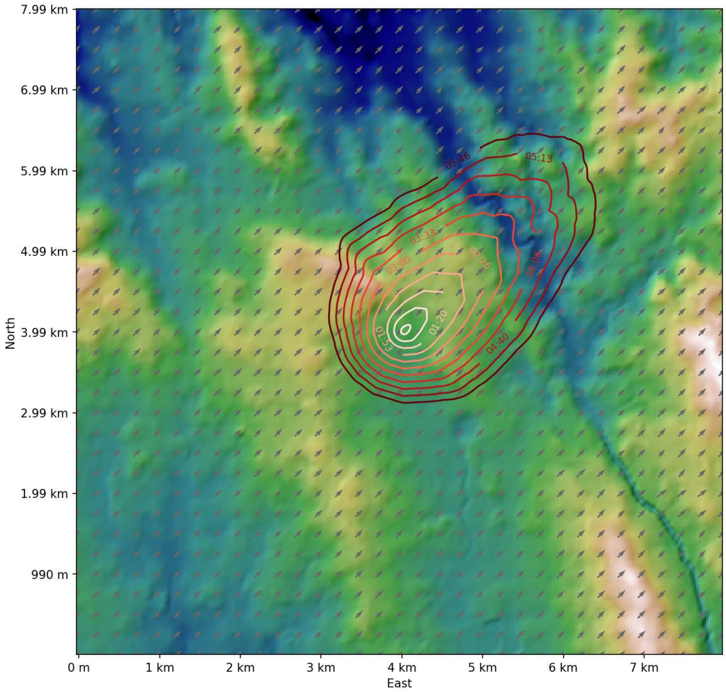


Figure 6.3: Illustration of the planning test situation 1: One wildfire

6.5.2 Demonstration of the observation planner over selected wildfire scenarios

This subsection illustrates the behavior of the variable neighborhood search planner over three distinct wildfire situations that reflect real life scenarios and demonstrate single and multiple UAV plans to monitor them. Figure 6.3 is a typical situation where an ongoing fire has to be mapped and the UAVs have sufficient endurance to observe the complete perimeter. Figure 6.4 considers a bigger fire and reduced autonomy. Figure 6.5 shows the case of multiple active wildfires of different sizes.

The observation plans have been generated using the VNS configuration previously introduced, during 30 seconds in an Intel Core i7-7820HQ CPU running at $2.90\,\mathrm{GHz}$. UAVs in situation 1 and 3 have 30 minutes of flight endurance and 15 minutes in situation 2. Figure 6.6 and Figure 6.7 demonstrate the priority to observe the front fire with respect to the backfire as trajectories contain more waypoints over the former than the later. When UAVs do not have enough endurance to cover the complete perimeter (Figure 6.8), the efforts are distributed (Figure 6.9). In situation 3, UAVs have sufficient endurance and the planner finds the best way to switch between independent perimeters (Figure 6.10). Although the trajectories of in Figure 6.11 look concomitant, UAV passage times are different: adding more aircrafts does not provide more information about the total perimeter, but more frequent updates as UAVs are scattered.

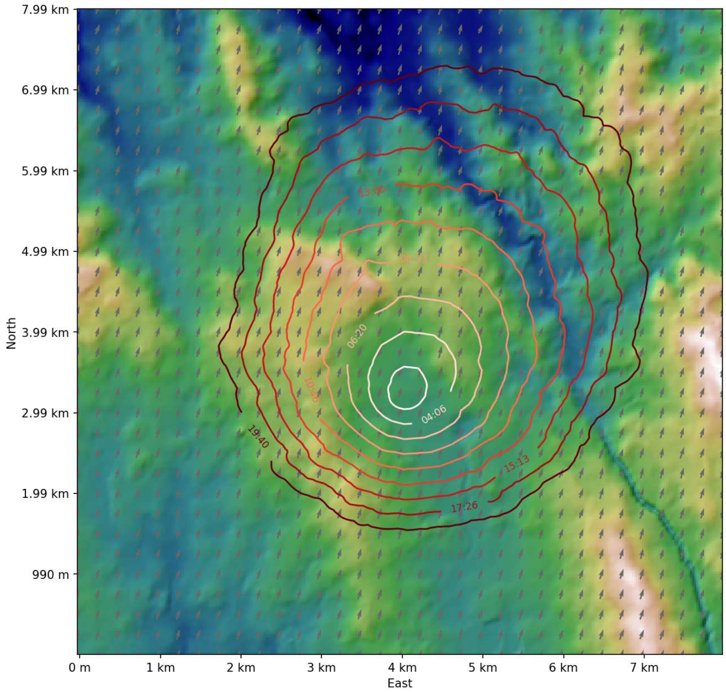


Figure 6.4: Illustration of the planning test situation 2: One large wildfire

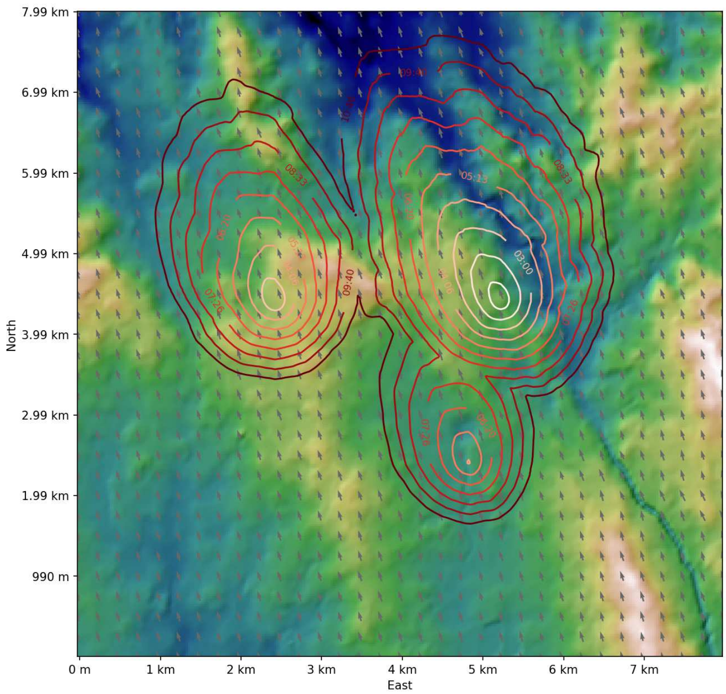


Figure 6.5: Illustration of the planning test situation 3: Three wildfires

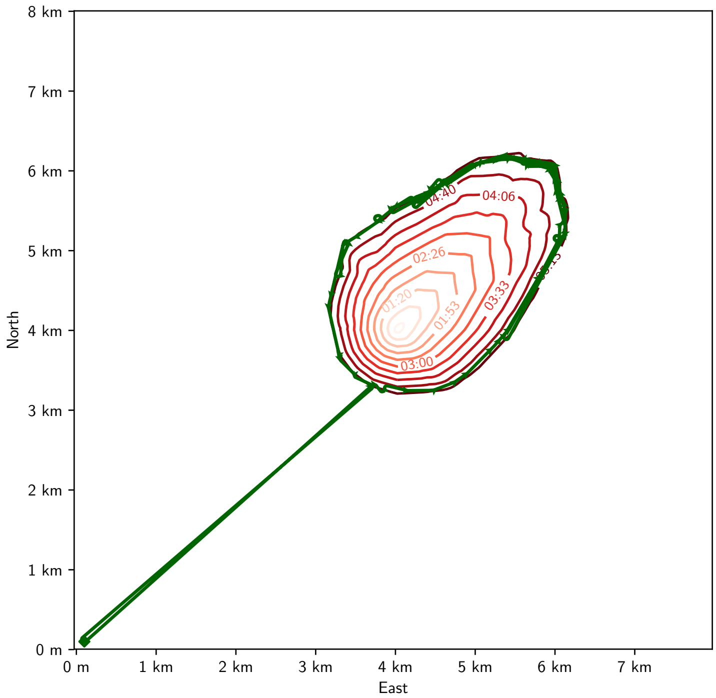


Figure 6.6: Illustration of plan to observe the wildfire of situation 1 with one UAV

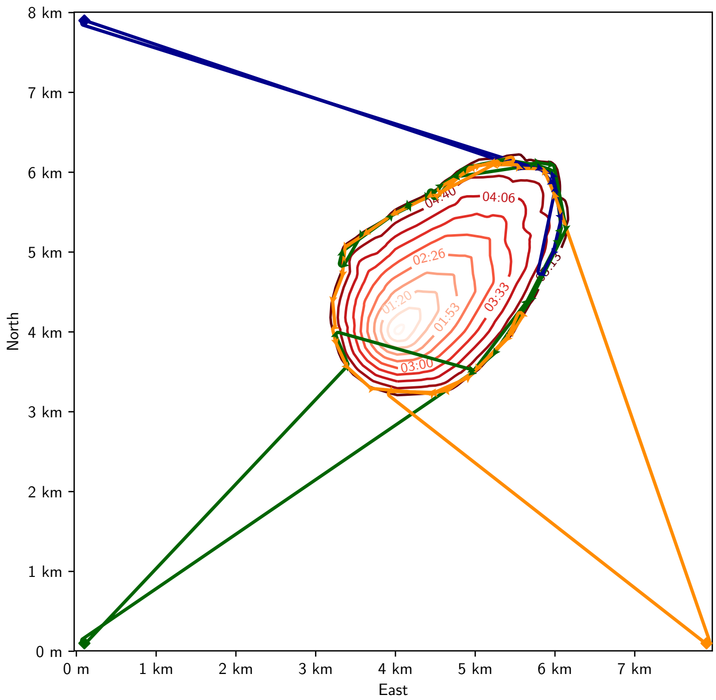


Figure 6.7: Illustration of plan to observe the wildfire of situation 1 with three UAVs

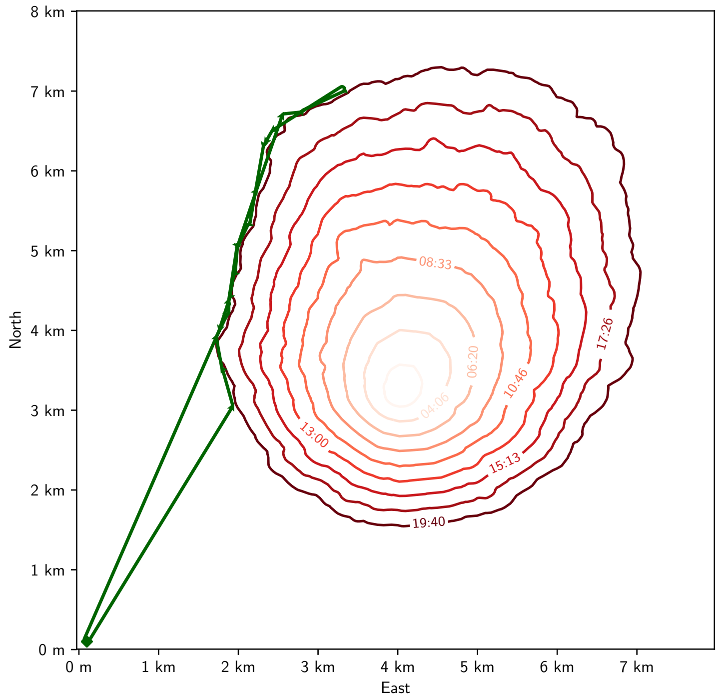


Figure 6.8: Illustration of plan to observe the wildfire of situation 2 with one UAV

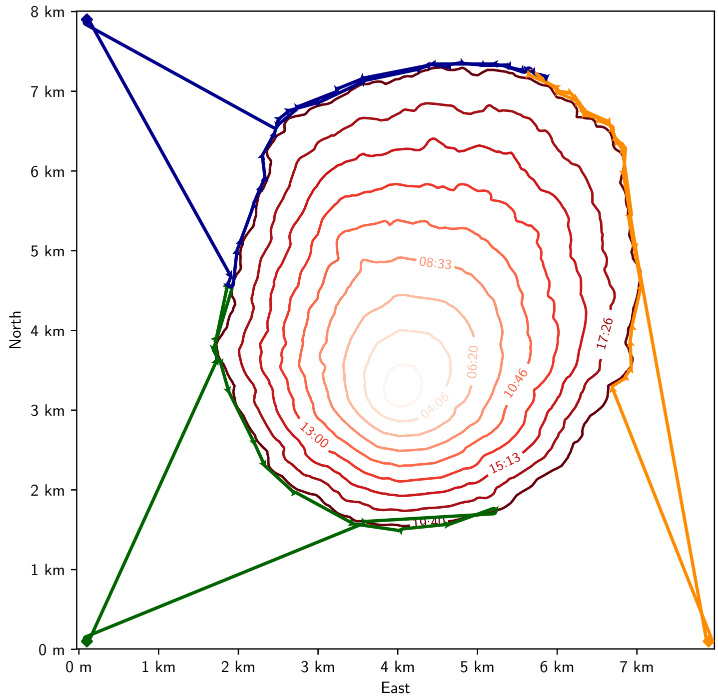


Figure 6.9: Illustration of plan to observe the wildfire of situation 2 with three UAVs

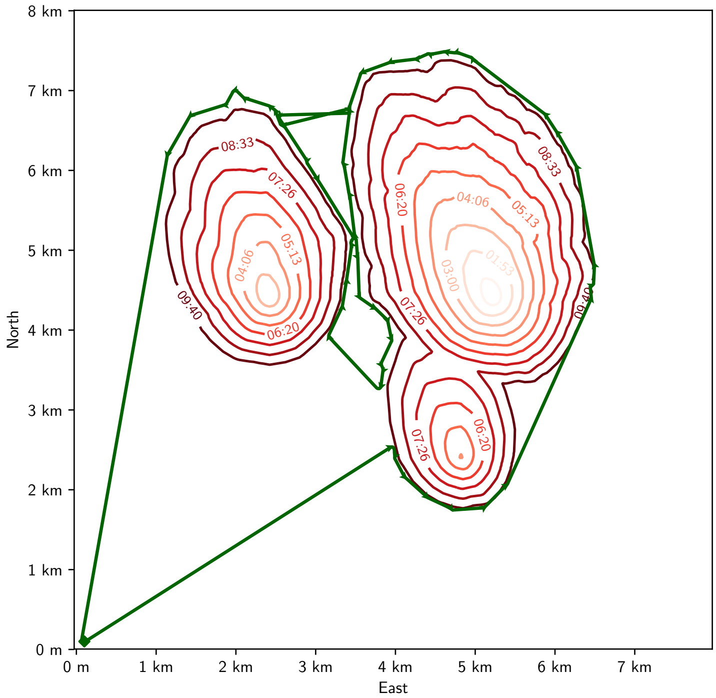


Figure 6.10: Illustration of plan to observe the wildfires of situation 3 with one UAV

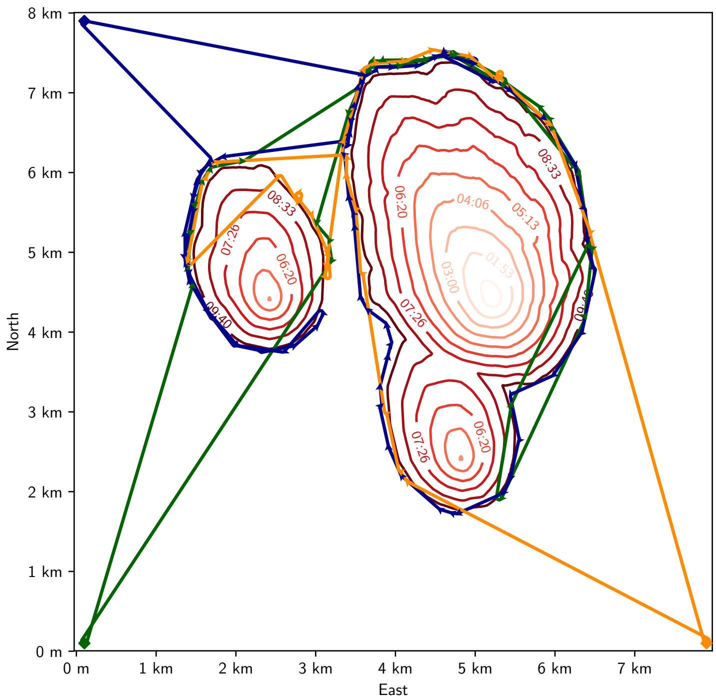


Figure 6.11: Illustration of plan to observe the wildfires of situation 3 with three UAVs

Illustration of the effect of wind on observation trajectories Figure 6.12 and Figure 6.13 reproduce situations where strong winds are present and have great influence in the resulting trajectory. In both cases, different flight patterns can be seen depending on whether the UAV is flying into or against the wind: when traveling with tailwind, the added ground speed allows covering more areas in less time. Conversely, the extra speed results in wider turn radius that make tight trajectories more difficult and, at the end, require longer paths.

The twisting paths flying north in Figure 6.12 elongate trajectory duration, but make longer straight segments available. Due to the effect of wind, in Figure 6.13 the turn radius is wider during the outbound part of the trajectory than in the inbound section in the north.

6.6 Conclusion

This chapter has introduced a definition to the wildfire observation problem, adapted from the Orienteering Problem, and a planning algorithm, inspired by the Variable Neighborhood Search metaheuristic, to produce realistic plans for fleets of UAVs to acquire the greatest amount of data about one or multiple wildfire perimeters.

Wildfire spread has specific traits that make its surveillance unique: the perimeter is the most valuable portion of wildfire topography to assess its spread shape and speed, which also happens to be dynamic. Due to its large size and observation time dependency, it is impossible to obtain a complete picture of the perimeter in space and time simultaneously. As a result, the wildfire observation problem has been defined as a variation of the Team Orienteering Problem with Time Windows and Dubins airplanes, including a utility value that guides the planning process.

The proposed planning algorithm, based on the generic VNS metaheurstic, evaluates local modifications to an initial plan, defined as a set of sequences of oriented waypoints. Enhancement proposals are generated by two classes of neighborhood structures, *insertion* and *path optimization*, adapted to this unique problem by making extensive use of sampling. The resulting plans are fitted to the observation of diverse wildfire situations with one or multiple UAVs.

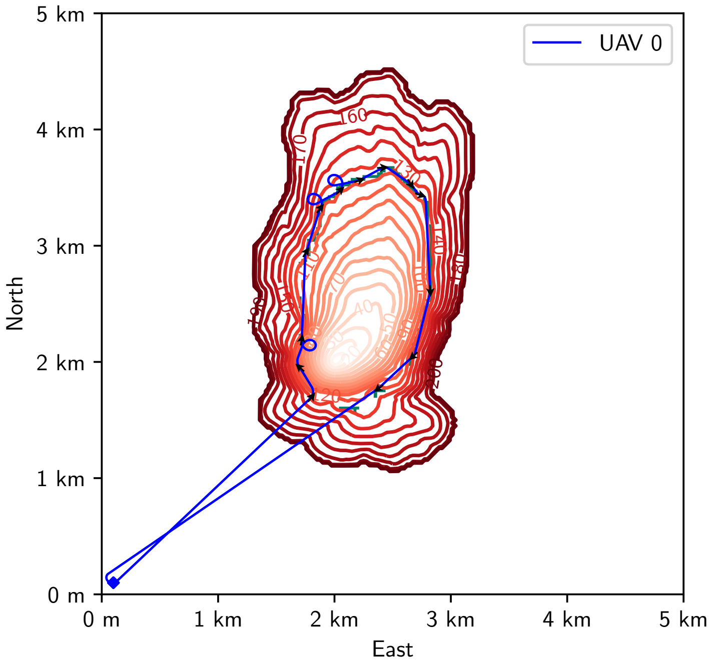


Figure 6.12: Illustration of an observation plan with strong wind blowing from the south.

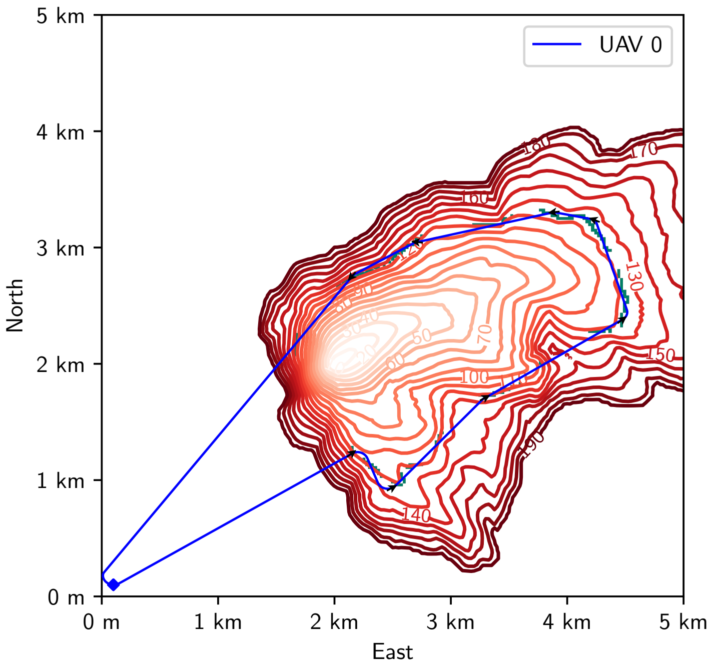


Figure 6.13: Illustration of an observation plan with strong wind blowing from the west. Note the turn radius is wider in the outbound part of the trip (south) than in the return section (north) due to the effect of wind.

Part IV

Integration

CHAPTER 7

Wildfire monitoring system integration

The integration of the wildfire monitoring algorithms and models to make them work together under real world conditions requires a software architecture to orchestrate the overall system behavior. This chapter introduces the FireRS SAOP wildfire monitoring architecture and the design of a mixed-reality simulation framework, a combination of real and synthetic components that ease the testing and development of the system. Additionally, the results of several test campaigns of the FireRS SAOP system are presented.

7.1 System architecture

> *Designing a robot architecture is much more of an art than a science. The goal of an architecture is to make programming a robot easier, safer, and more flexible. [Siciliano 2008, p. 202]*

The integration of the various models and algorithms introduced in previous chapters in order to build a complete wildfire monitoring platform requires addressing the issue of how components are combined in order to work together seamlessly as a *system*.

Modern robotic systems are built upon a multitude of software and hardware entities that are themselves made up of other components. Sensors, actuators, and control software among others can be identified as independent sub-systems. However, in the context of their interaction with other components, they constitute modules or building blocks that are part of a bigger system. The challenge of designing robot software architectures resides essentially in the identification of which are the relevant sub-systems the platform is made up of, their purpose, and how their interactions is carried out in a way that makes sense to the particular application. With this respect, an architecture built upon modules has many advantages with respect to monolithic structures: it fosters component reuse, so a particular design can benefit from previous work or from new pieces of software and hardware when system specifications evolve.

Software architectures typically require the integration of heterogeneous pieces of work that come from different manufacturers and run on distinct operating systems, processor architectures and network interfaces. A middleware provides a common component model and a homogeneous data exchange protocol freeing the

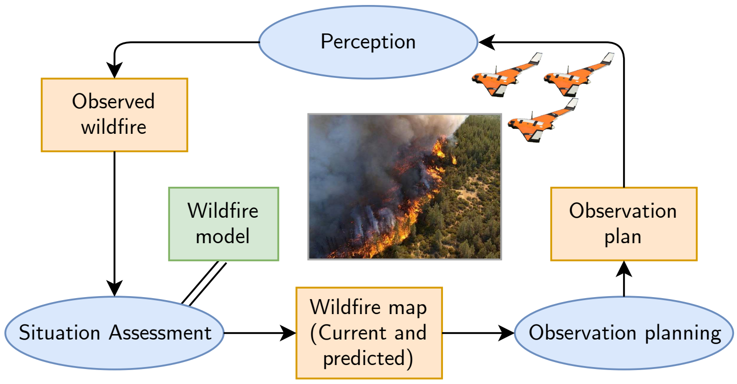


Figure 7.1: FireRS SAOP wildfire monitoring concept architecture

user from platform-dependent operations by obeying to a standard protocol. Robot oriented middleware also provide tools for inspection and storage of exchanged data and the management of all module execution life-cycles. For instance, being able to restart some component in case of error without taking down the complete system.

The wildfire monitoring system introduced in this thesis, known as FireRS SAOP, is based on the design of an architecture denoted *SuperSAOP* from *Supervision of Situation Assessment and Observation Planing*. Its goal is to orchestrate the processes of planning and executing observation missions, the procurement and analysis of wildfire-related information, and providing final users with a comprehensive understanding of the wildfire situation. This software architecture is mainly built over the Robot Operating System (ROS) middleware, that serves as the common foundation of the integration of the algorithms and models depicted in previous chapters, an external command and control architecture for the fleet of UAVs and a dedicated communications satellite.

7.1.1 Overview of the FireRS SAOP system architecture

The FireRS SAOP system architecture following the sense-plan-act paradigm: it operates as a closed loop of perception, decision and action activities. As illustrated in Figure 7.1, the observations made by a fleet of UAVs are used to produce a global understanding of the wildfire situation that serves afterwards to plan the best course of actions for the UAVs to observe the wildfire.

The overall architecture of SAOP, depicted in Figure 7.2, is built around the situation assessment (perception) and observation planning (decision) components, whose functions correspond to the algorithms introduced in chapter 5 and chapter 6 respectively, and the execution platform (action), that is the UAVs and the ground control station. There is also a supervision component that orchestrates the overall system operation with the concurrency of human agents.

The operation of FireRS SAOP is initialized with a wildfire alarm that holds the location and the detection time of a new fire outbreak. Next, the situation assess-

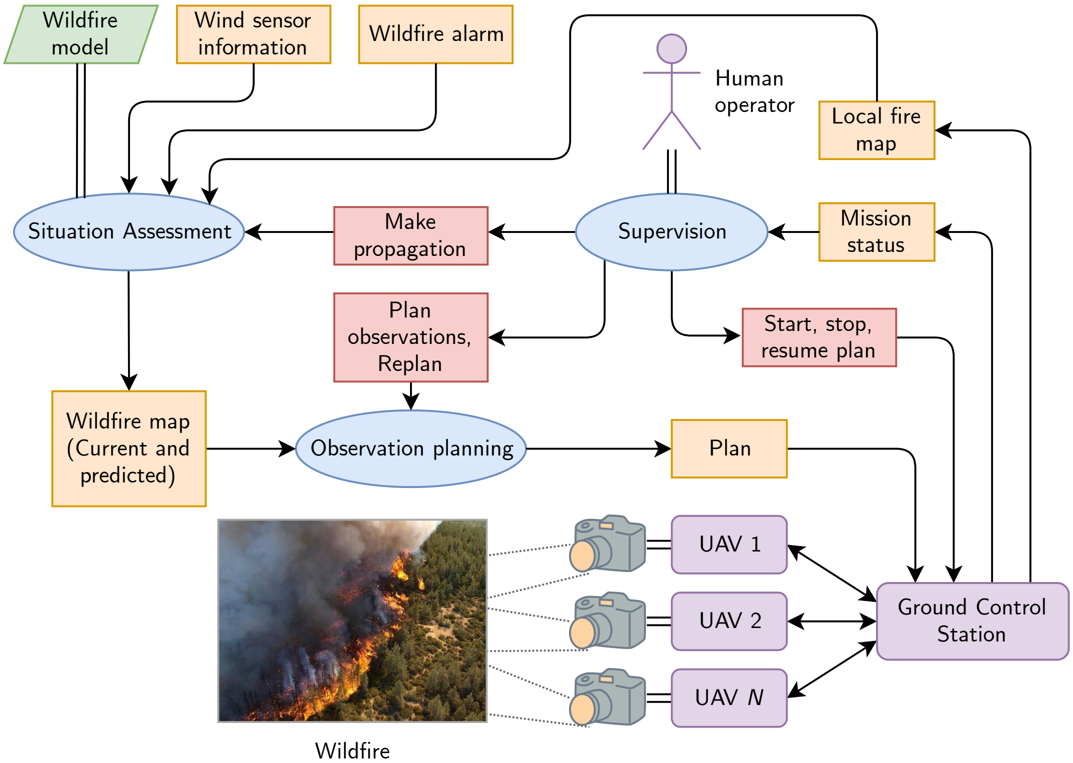


Figure 7.2: FireRS SAOP system architecture. Orange boxes represent data messages, red boxes are commands, and blue ovals are the main processes.

ment subsystem produces a wildfire forecast using the wildfire model depicted in chapter 3 and the wind information coming from the sensors. In later stages, local observed fire maps are combined to update the prediction. Continuing, the wildfire assessment, current and predicted, is transmitted to the observation planer which produces a monitoring plan. Then, UAVs are to and continuously generate updated maps of the regions they are overflying (under the control of the UAV ground control station). The supervision module, oversees system execution and handles the human-machine interaction: a graphical user interface shows the wildfire maps and the mission status to the human operator that is able to controls other components by issuing commands manually or let the supervisor do it autonomously.

7.1.2 UAV control software

The UAV platform is managed by the LSTS toolchain [Pinto 2013], a software architecture developed by the University of Porto for the operation of heterogeneous fleets of unnamed aerial and marine vehicles:

- *Dune* is the embedded software that drives the UAV guidance, navigation, control, network and sensors. The wildfire mapping algorithm also runs as an integrated module.
- *Neptus* is the command and control graphical interface for the operation of Dune-enabled vehicles. Neptus supports the different phases of a typical mission life cycle: planning, execution, and mission analysis. Figure 7.3 shows the Neptus graphical interface illustrating the monitoring of UAV flight mission.

The LSTS software suite uses its own message-oriented communication protocol named IMC (Inter-process Module Communication) tailored to the operation of the University of Porto autonomous vehicle platform. IMC is a robust communication protocol tested in a wide range of scenarios with heterogeneous fleets of robots and designed to support various physical network infrastructures.

7.1.3 Communication infrastructure

Inter-process communication between FireRS SAOP components is mainly based on the Robot Operating System (ROS) middleware with interfaces to other special-purpose communication protocols for the fleet of UAVs and the satellite.

ROS is a collection of software specially designed for robotic system architectures that handles communication between heterogeneous components, —algorithms, sensors, actuators— using a common message-oriented publish-subscribe model. The ROS middleware component model is designed around a graph of *nodes*, independent processes, that communicate by exchanging messages through named channels known as *topics*. Nodes interact with topics in a publisher-subscriber manner. ROS messages are a collection of simple fields of primitive types or other message types and typically header with a timestamp.

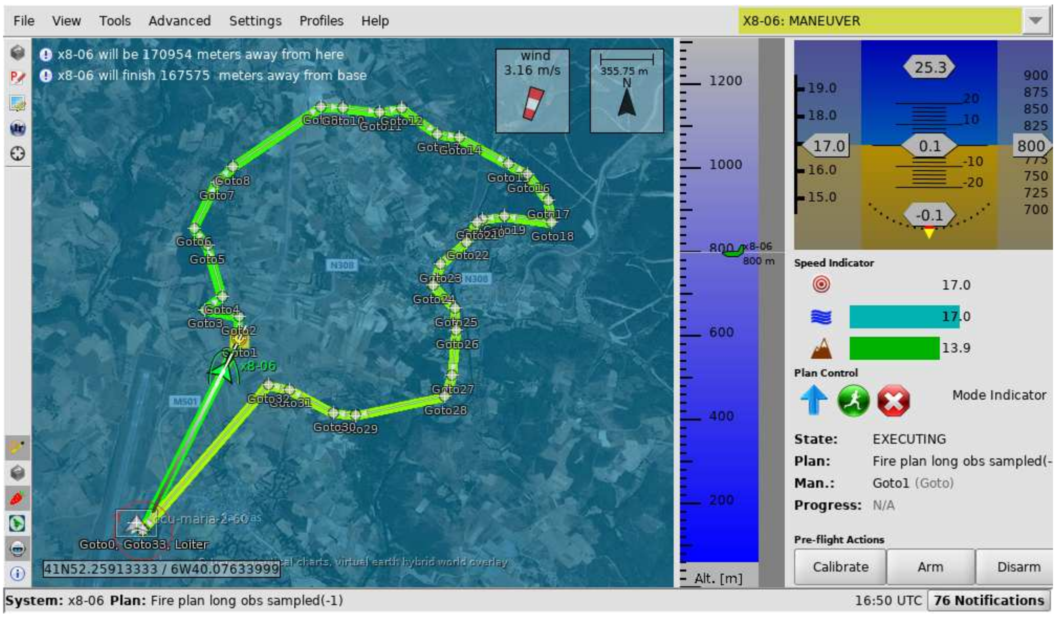


Figure 7.3: Illustration of the Neptus command and control software while operating autonomously a UAV

The FireRS SAOP communication model, which is based on the ROS message model, is built upon on two kinds of messages: data and commands, depicted in Figure 7.2 in orange and red respectively. Data messages carry data like wildfire maps, observation plans and UAV state reports from a producer node to other nodes that share an interest on the particular piece of information. Command messages carry instructions and their parameters to the different components in order to influence their behavior. This gives the supervisor, manual or automated, the ability to operate the complete system at its discretion.

7.1.3.1 Interaction with the UAV control software

The integration of the LSTS toolchain and the IMC protocol into the FireRS system, illustrated by Figure 7.4, is achieved with a bridge module that links the two networks and performs the marshaling between selected IMC messages and ROS messages involving UAV mission control and wildfire mapping.

This bridge module has two parts: A plugin in the Neptus software that listens to selected IMC messages and a ROS node that converts those IMC messages into their corresponding ROS ones. The Neptus plugin sends IMC messages to the ROS node as TCP packets.

7.1.3.2 Interaction with the satellite communication system

The satellite link is used to receive wildfire alarms from the ground sensors that are typically located at remote locations in the wildland. Because the data throughput

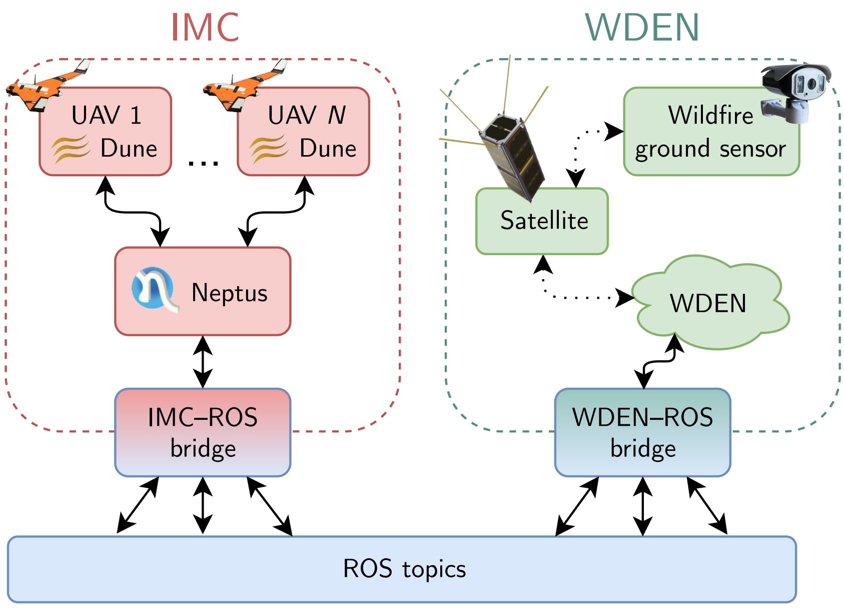


Figure 7.4: FireRS SAOP integration of the fleet of UAVs and satellite external network infrastructure

of the communications satellite is limited, only small pieces of information can be transmitted at once. In particular, the up link data transmission rate is 2400 bits per second and 1200 bits per second for the downlink. Also, network access is divided into small half-duplex time slots [Nercellas 2019].

The WDEN (Wildfire Data Exchange Network) is a cloud-based service that encapsulates the complexity of the satellite communication link with a simple TCP/IP interface. Like the IMC–ROS bridge node, a WDEN–ROS node connects to the WDEN to receive the wildfire alarm messages and then publishes them into the corresponding ROS topics. Figure 7.4 illustrates the key elements of the WDEN.

7.1.3.3 Overall dataflow

Figure 7.5 presents the ROS architecture of the whole system, with all the data and control flow topics.

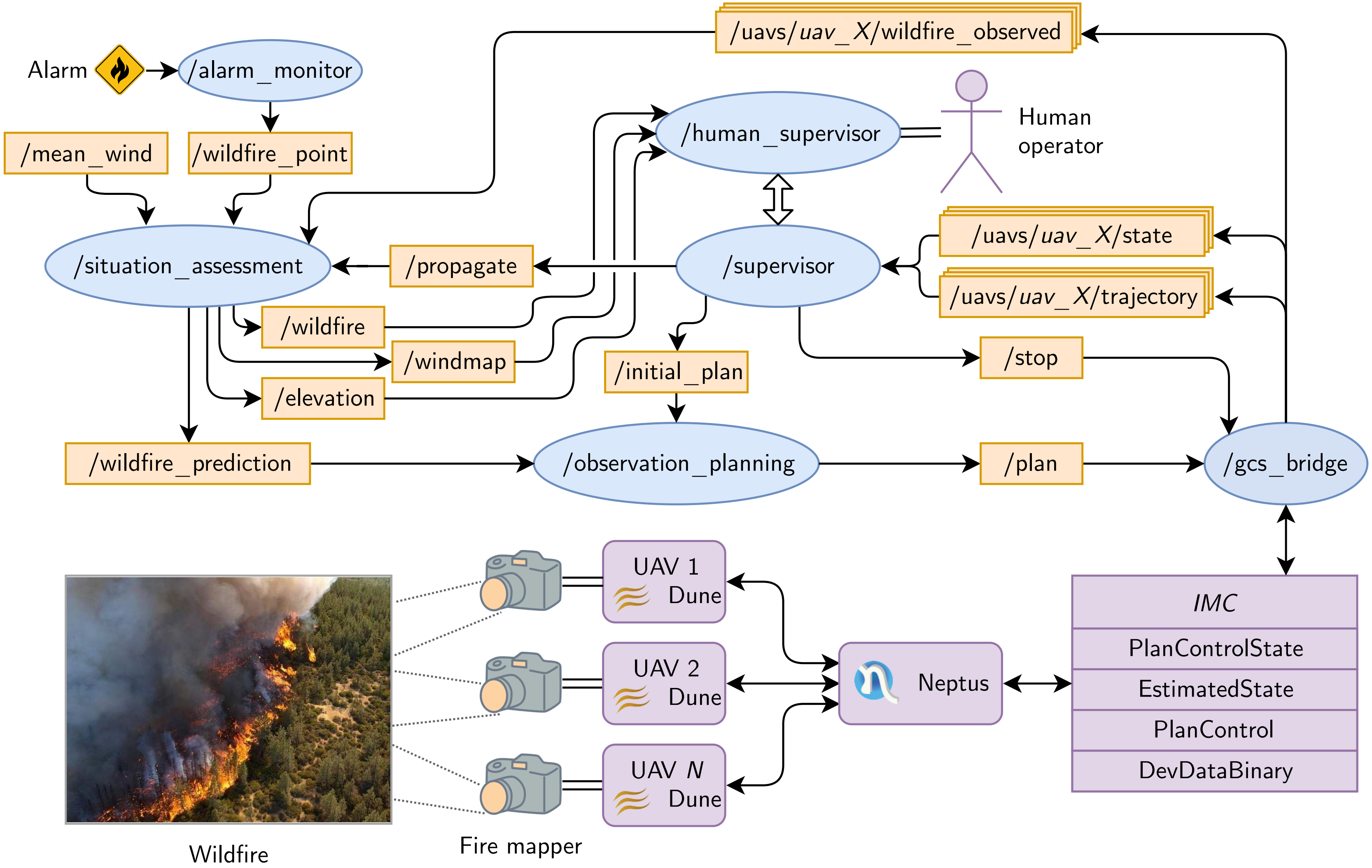


Figure 7.5: FireRS SAOP implementation details. Similar color and shape conventions as in Figure 7.2 apply: orange boxes are ROS topics and blue ovals, ROS nodes. Ground control station components are depicted in purple.

7.1.4 Subsystems

This subsection details the integration of the various FireRS components, grouped in subsystems according to the functionality they provide:

- Wildfire mapping and situation assessment
- Observation planning
- Mission execution
- Supervision

7.1.4.1 Wildfire mapping and situation assessment

The wildfire mapping and situation assessment sub-system functionality is distributed across multiple locations of the FireRS SAOP system. Wildfire perception and mapping algorithms are carried out inside the UAV embedded computer and the prediction of wildfire spread and fusion with the observed wildfire maps is done on ground. The reason for this separation is twofold: performance and autonomy.

The transmission of the infrared images to the ground station through an irregular network link may result in information losses. Contrarily, the execution of the fire mapping algorithm in the UAV embedded computer, while demanding, only requires the delivery of wildfire maps that are updated at a reduced rate compared to the production of infrared images.

Providing the UAVs with the ability to map wildfires is a first step into increasing the overall autonomy score of the system in the terms described in chapter 2. Future wildfire monitoring UAVs may have increased active perception capabilities to plan their own trajectory on the basis of their observations.

7.1.4.2 Observation planning and mission execution

The observation planning algorithm of the FireRS SAOP system corresponds to the one depicted in chapter 6, but the resulting plan needs to be adapted to the UAV mission format.

The IMC mission execution model is focused into the actual operation of the UAVs, so its specification do not match exactly SAOP plans whose definition is more abstract. The generic definition of FireRS SAOP plans requires to be translated from its ROS message form to the closest IMC message format, which in fact does not share the same layout. Within the IMC message model, a plan is associated to a sequence of maneuvers for one vehicle while for SAOP a plan encompasses the set of trajectories for a fleet of UAVs.

In nutshell, an IMC Plan, a *PlanSpecification*, is composed of an id, a list of maneuvers, and a list of transitions. Maneuvers relevant to UAV operation include Goto, Scheduled Goto, Take-off, Land and Loiter, but there are many others tailored to different scenarios. Transitions are messages that describe the index of the next maneuver and additional actions that are triggered with the transition. This is used to enable the fire mapping module when approaching the wildfire and disabling it when returning to the landing site.

Additionally, the security procedure requires that the UAV operator validates the plan before its launch.

Translation between ROS and IMC messages As depicted in Figure 7.4, a *bridge* component is in charge of interconnecting the IMC and ROS networks by acting simultaneously as an IMC server and ROS node. This communication block handles message marshaling from one format to another, accounting for the design differences between the UAV control software and FireRS SAOP.

The translation of plans from SAOP to IMC definitions has two particularities: First, due to the aforementioned model differences, FireRS SAOP plans are decomposed into trajectories so plan progression has to be followed independently for every UAV. The IMC *PlanSpecification* identification field is used to track the association between a SAOP plan trajectories and IMC plans.

Second, because FireRS SAOP plans rely heavily on timing, waypoints are translated into *scheduled goto* maneuvers so the UAV guidance software controls the airspeed in order to respect the desired time of arrival. In fact, even if SAOP plans are designed to be realistic with respect to UAV motion, errors in the arrival time can delay the last segments of the planned trajectory by a noticeable amount of time without regulation of the airspeed. A limitation of the IMC *goto* maneuvers and the UAV navigation software is the unfortunate lack of support for arbitrary waypoint orientations. As a consequence, UAVs are restricted to a line-of-sight style navigation approach between waypoints. A typical strategy to overcome this limitation is to sample the original sequence of Dubins paths, but for this application, due to the relative long distance between waypoints, the benefits do not make up for the added complexity.

Because robust autonomous take-off and landing is still a rare feature of current small UAVs, the operator typically handles take-off manually until the UAV goes up to a given altitude and stays in *loiter* mode. As a result, the first and last waypoint of SAOP trajectories, those describing the start and end points, are handled differently: They are replaced with loiter maneuvers, so UAV control can be transferred from and to manual operation safely.

Plan progression and UAV state reports are regularly transmitted back to SAOP, and published as ROS messages in independent topics for each UAV in the fleet.

7.1.4.3 Supervision

In a fully autonomous system, a complete *Supervision* component should be in charge of the control and synchronization of all the system processes. However, we opted for a less ambitious scheme, yet more realistic in the application context of FireRS: the supervision component is mainly the interface with the system operator. Its role is twofold: first, it handles the data disquisition and presentation,

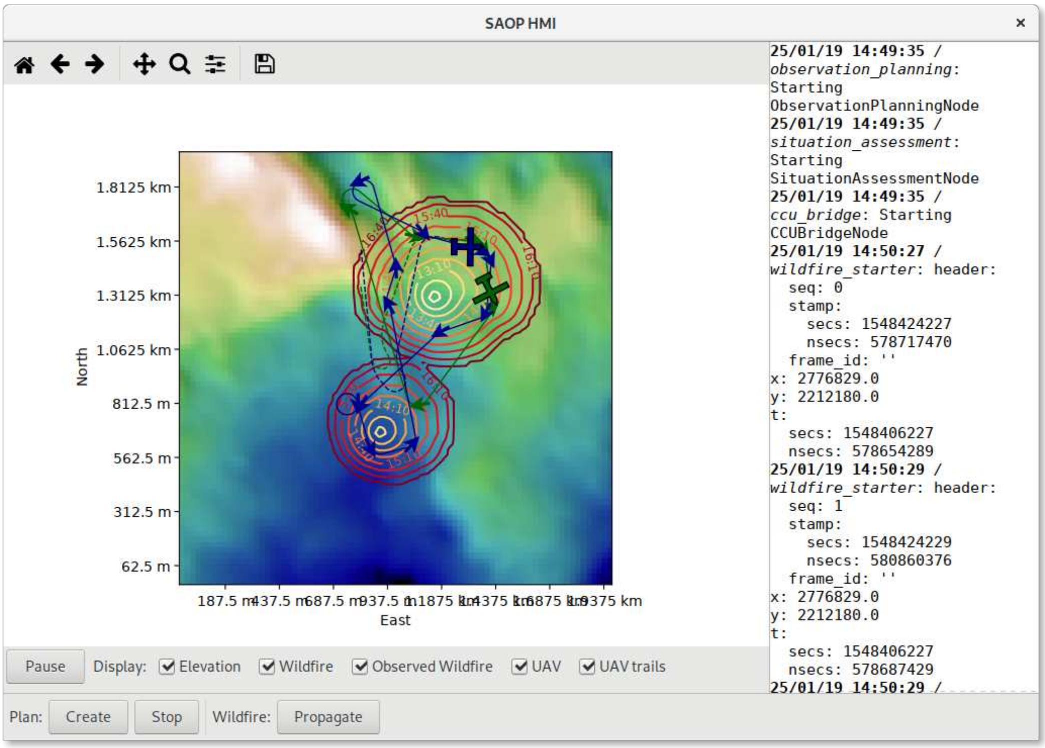


Figure 7.6: Screenshot of the FireRS SAOP human-machine interface

by monitoring the relevant information that are exchanged between subsystems and makes the necessary transformations, so they can be displayed in a graphical human-machine interface. Second, it coordinates the behavior of FireRS SAOP components by issuing actions commands. This endeavor is shared between the operator and the supervisor: using the graphical HMI, the operator can task the system by issuing commands (such as the specification of an area to observe). Once the system is given goals, the supervisor can be configured to keep the system running the predefined monitoring loop.

The graphical HMI, shown in Figure 7.6, displays monitoring mission information in the center of the screen as a map that depicts the terrain elevation, the wildfire map —a combination of assessment and prediction— and the observed wildfire cells. The observation plan is shown on the foreground along the location, orientation and executed trajectory of the fleet of UAVs. High level commands *Create plan*, *Stop plan* and *Predict wildfire*, found at the bottom of the screen, allow the manual control of the system. Acting on the *Start plan* button opens a window offering some planning configuration options such as selecting which UAVs should participate in the mission or deciding the time to start the mission. On the right side, the user can monitor the details of the messages being exchanged between the modules for debugging purposes.

7.2 Mixed-reality simulation of wildfire monitoring with fleets of UAVs

Testing and experimenting with the FireRS SAOP wildfire monitoring system in real life is a major challenge due to the nature of wildland fires and the logistics of fixed-wing UAV operation.

Wildfires are by definition uncontrolled fires spreading over rural areas, so the possibility of performing tests depend on the “chances” of an outbreak happening at a particular time. A planned fire in a controlled environment could be used as an alternative, but due to the magnitude of the event, one cannot expect to repeat the experiment twice with the same conditions: once an area is burnt, the test can not be replicated until vegetation grows again. In this context, computer simulation is the only feasible way to obtain arbitrary repeatable wildfire test scenarios.

Indeed, UAV operation is not as extraordinary as a spreading wildfire, but it is still tedious, time-consuming, and theretofore costly. Hence, simulated unmanned aircraft platforms play a crucial role in system design and early testing, but synthetic UAVs can not capture alone all the integration problems and errors that typically arise in complex real world situations. The solution comes from hybrid or mixed-reality (MR) testbeds, were a mixture of real and simulated components work together seamlessly and are arbitrarily interchangeable.

Mixed-reality (MR) is a broad term that designates the technologies of the *virtuality continuum* [Milgram 1994] that merge real and virtual elements. This concept includes the widely known Augmented Reality (AR), were representations of the real world are "augmented" with computer generated graphics, and Augmented Virtuality (AV), with synthetic worlds being affected by real-life events. In general, mixed-reality covers the middle ground virtual and real worlds are mixed.

Two important concepts regarding the development process of embedded devices, and autonomous vehicles in particular, are Software-In-the-Loop (SIL) and Hardware-In-the-Loop (HIL) simulations. The former runs the real control software in a simulated platform, and the latter is executed over real hardware. For instance, a UAV autopilot SIL simulation runs the algorithms in a test PC while the HIL simulation uses the actual embedded card.

According to the literature regarding remote sensing multi-UAV systems, well-designed integration and modular architectures provide substantial help during the transition from pure virtual to complete real products. With this respect, [Rollo 2015] defines three test stages: *Purely virtual*, *mixed-reality*, and *fully deployed hardware*. The first stage is suitable for early development when frequent reproducible tests are required. Next, real components are gradually introduced in mixed-reality scenarios. At this stage UAVs are typically simulated as aerial vehicles are the most expensive and cumbersome to operate pieces of hardware. Finally, the fully deployed hardware stage correspond to a real application setting and is reserved to the final design phases. [Selecky 2017] introduces a second-to-last *fully* *deployed hardware field test* phase were UAV platforms are operated, in their final configuration, outside an actual production environment.

An example of a mixed-reality simulation using fleets of UAVs is [López Peña 2016]. This paper sets up a combined real-synthetic remote sensing scenario of a simulated polluted smoke plume emerging from a factory chimney, modeled after realistic conditions. Real UAV and environment elements are added progressively during product design: first, one simulated UAV is used inside a simulated environment to adjust flight controls. Then, a test with a real aircraft is performed near the factory to gather real data to validate the smoke plume model. Finally, a fleet of combined real and simulated UAVs is set to collectively monitor the environment. The real UAV measurements are used to update the model of the phenomenon feeding the simulated aircraft sensors. A similar approach is devised for the FireRS SAOP wildfire monitoring architecture.

7.2.1 Implementation of a mixed-reality wildfire mapping framework

Considering the aforementioned progressive simulation scenarios, the FireRS SAOP architecture is designed to operate mixed-reality simulations of the fleet of UAVs and the wildfire propagation. The framework can be exploited in flexible combinations of simulated and real UAVs, and real and synthetic fires. In particular, the tested arrangements include: A pure virtual environment, and a fleet of mixed real and simulated UAVs with a synthetic wildfire. Further associations of real and simulated UAVs over real wildfires could be used without major architectural modifications once the possibility to fly over actual fires will be granted.

The mixed-reality wildfire mapping architecture, depicted in Figure 7.7 exploits the Morse robotics simulator to set up a virtual infrared perception environment to observe an artificial wildfire, generated using the model introduced in chapter 3. The synthetic images, produced according to each UAV position and orientation are sent to the fire mapping module embedded into the aircraft so updated local observed wildfire maps can be produced. Real UAVs get infrared images from a real camera and the produced maps are disseminated in SAOP using the same interface as the simulated ones.

7.2.1.1 UAV simulation profiles

The challenge of having fleets of real and virtual fleets of UAVs simultaneously is mostly solved by the LSTS toolchain that encompasses the Neptus, Dune and IMC software. Fortunately, this architecture is built around various execution profiles that account for different level of realism: *Simulation*, *SIL*, *HIL* and *Hardware*. The SIL configuration relies on Ardupilot[1] SIL simulation, based on the realistic JSBSim flight dynamics model [Berndt 2004] and, conversely, the simulation configuration uses a basic 6-degree of freedom model. The flexible modular architecture of Dune

[1] https://ardupilot.org

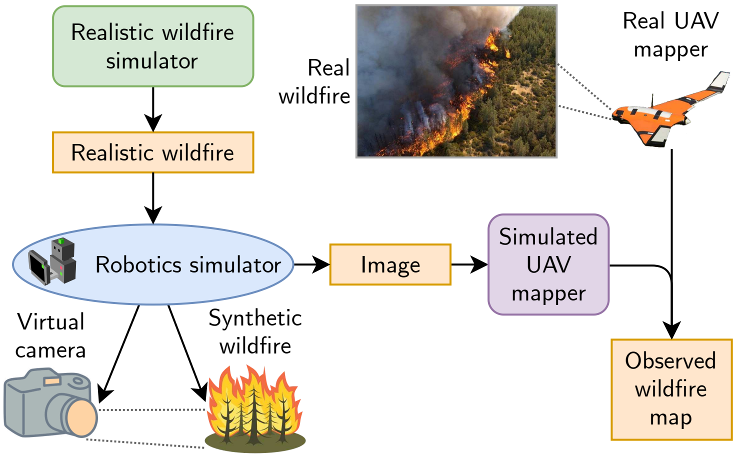


Figure 7.7: Mixed-reality wildfire mapping architecture

and Neptus yields the possibility of virtual and real autonomous vehicles to coexist transparently to SAOP and the operator.

7.2.1.2 Wildfire environment and perception simulation

Morse is a robotic simulation engine developed on top of the Blender 3D creation suite that has been used to simulate complex robotic scenarios [Echeverria 2012] and integrate realistic simulation engines [Degroote 2016]. A Morse simulation scene definition has two parts: first, the environment is designed in blender like any other 3D environment. Then, a custom initialization script describes the robots, with their sensors and actuators, involved in the simulation. Other software can interact with the simulated scene through common robotics middleware like ROS or via a standard TCP communication protocol.

The synthetic FireRS SAOP wildfire environment is created prior to the simulation with the DEM of the incumbent area, which is imported into Blender using a special GIS plugin[2]. The result is illustrated in Figure 7.8.

During simulation, the real wildfire is displayed as the terrain texture, and updated frequently in real time. As depicted in Figure 7.9, the wildfire propagation texture is a black and white image with light pixels denoting terrain cells currently on fire. The choice of this color palette is backed by the nature of infrared cameras, that produce gray-scale images as they are designed to sense the intensity of the radiation emitted. Unfortunately, unlike the robots defined in the initialization script, the environment is static. As a result, Morse had to be modified to add a command enabling arbitrary changes to the texture of a non-robotic entity.

The simulated wildfire mapping software, that runs as a Dune module in the UAV, sets the pose of the virtual robot to match the location and orientation of

[2] https://github.com/domlysz/BlenderGIS

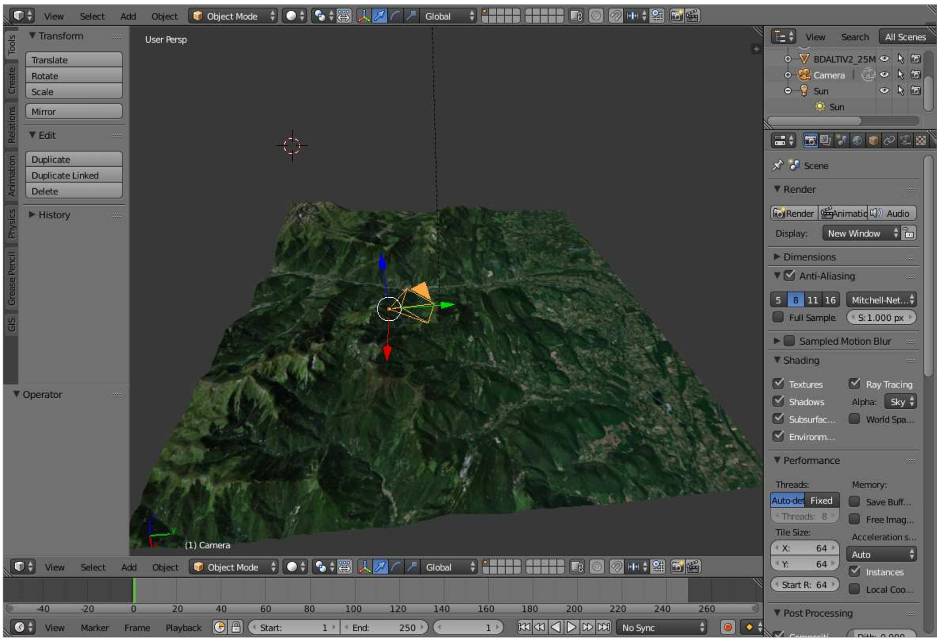


Figure 7.8: Illustration of Blender showing a Morse environment of a real rural area digital elevation map.

the aircraft —simulated or real— and immediately asks Morse for an image. The content of the picture is then used as the source for the mapping algorithm.

7.3 Field tests and results

This section reports about the demonstration campaigns of the SAOP architecture performed in collaboration with the FireRS project partners and stakeholders. The purpose of the demonstration sessions was to validate all the capabilities of the complete FireRS SAOP system over three independent functional scenarios of increasing complexity.

7.3.1 Test scenarios

The three test scenarios are:

INIT Detection of a wildfire and communication of the alarm to SAOP.

RUN Confirmation of the fire alarm by an UAV.

MONITOR Continuous monitoring of the wildfire perimeter.

The following subsections describe in detail the operation of the aforementioned test scenarios.

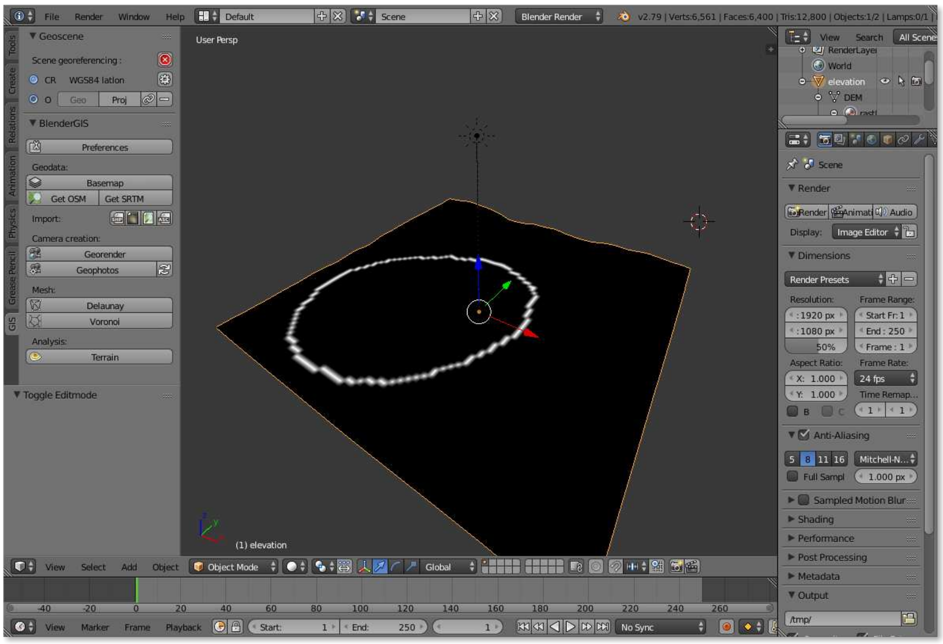


Figure 7.9: Illustration of Blender showing a synthetic wildfire. Terrain surface is in black and the active wildfire perimeter in white to imitate fire infrared emissions. During the simulation in Morse, the white perimeter is updated in real time.

7.3.1.1 INIT scenario

The objective of the INIT scenario is to showcase the triggering of an alarm after detecting a wildfire followed by the generation of an observation plan expected to verify the alarm. This scenario validates the work of the land sensor, the dissemination of an alarm through satellite communication and the generation of an observation mission. Figure 7.10 depicts the elements of the FireRS SAOP system involved in the operation of this scenario.

During the INIT test scenario the job of SAOP is:

1. The reception and decoding of the alarm message.
2. Predicting wildfire spread from realistic geographic and weather information.
3. Generating a flight plan for a UAV in order to verify the alarm and confirm the existence of a wildfire.

7.3.1.2 RUN scenario

The RUN scenario is a continuation of the INIT scenario with an emphasis on UAV action: the execution of the initial observation plan and mapping of the suspected area. During the scenario RUN, depicted in Figure 7.11, SAOP performs:

1. The reception and decoding of the alarm message

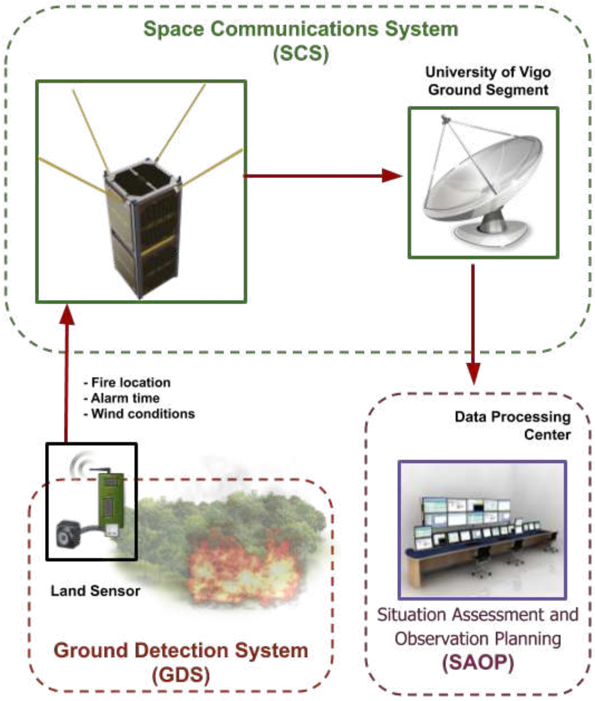


Figure 7.10: Diagram of the components involved in the INIT scenario

2. A wildfire propagation forecast using real geographic data and realistic weather information.
3. The creation of a flight plan for an UAV to verify the alarm and confirm the existence of a wildfire.
4. Then encoding and delivery of the flight plan to the UAV command and control software.
5. While the UAV is executing the observation plan, the reception and display of the initial wildfire map created by the UAV from the detected fire.

7.3.1.3 MONITOR scenario

The MONITOR scenario consists in periodic wildfire observation tours to continuously update the fire map so propagation forecasts can be also updated. For this task, a UAV follows an observation plan that lies over the expected wildfire perimeter in order to confirm its location.

Given an initial perimeter, from a previous MONITOR or RUN scenarios, SAOP performs the following operations:

1. Prediction of the wildfire propagation from a previous wildfire perimeter and updated geographic and weather data.

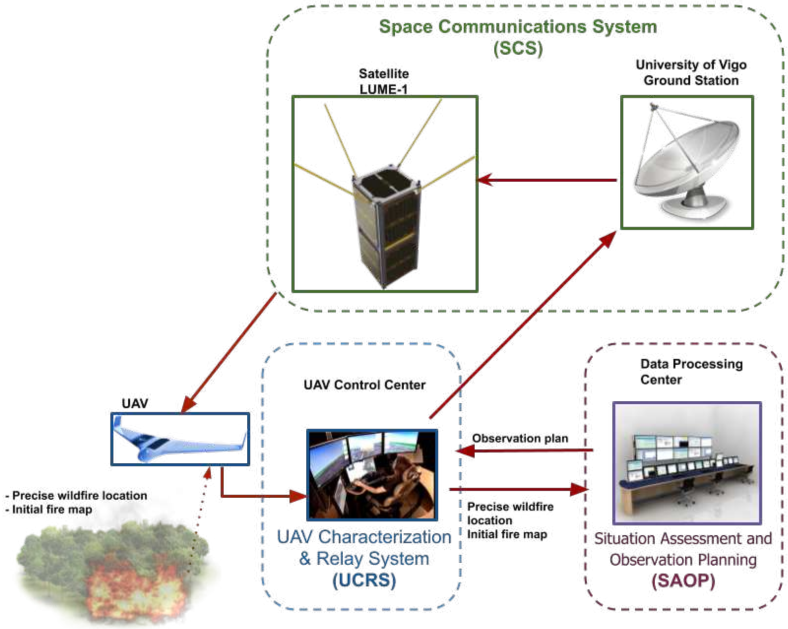


Figure 7.11: Diagram of the components involved in the RUN scenario

2. Generation of an observation plan that follows the expected wildfire perimeter.

3. Encoding and delivery the flight plan to the UAV.

4. During the ongoing observation plan, the continuous production of fire maps immediately presented to the operator.

Figure 7.12 illustrates the components involved in the MONITOR scenario.

7.3.2 Demonstration campaigns

The scenarios presented in the previous section were tested during two demonstration campaigns held in Spain and Portugal in April and May 2019 respectively. Spanish demonstration campaign featured the satellite communication and wildfire land sensor platforms, and their interaction with the SAOP architecture with a real wildfire and a simulated one. The Portuguese campaign focused on wildfire observation planning and action with a real UAV flying in a mixed real-synthetic environment.

7.3.2.1 Vigo demonstration campaign

The first campaign was held with the purpose of demonstrating the INIT scenario with a synthetic fire alarm and real fire situation over two locations near Vigo (Fig-

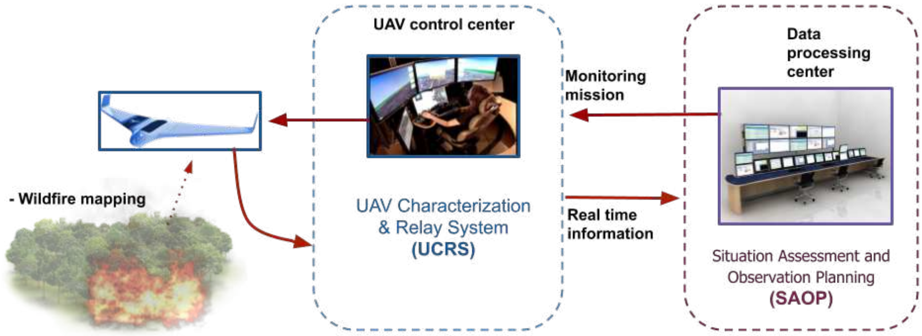


Figure 7.12: Diagram of the components involved in the MONITOR scenario

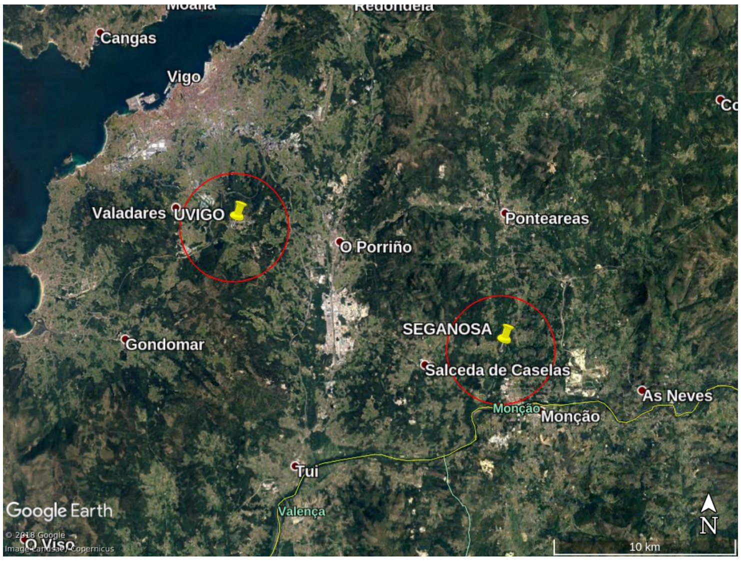


Figure 7.13: Demonstration sites in Spain: The campus of the university of Vigo (left yellow marker) and SEGANOSA fire emergency training center (Right marker)

ure 7.13). Due to the lack of the legal authorization to fly autonomous UAVs, only the demonstration of the INIT scenario was possible.

The location of the first demonstration, held the 23th April 2019, was the campus of the University of Vigo (Spain) with a synthetic alarm. The focus of this session was to demonstrate the behavior of the satellite communication infrastructure within the INIT scenario.

The location of the second demonstration, held the 24th April 2019, was the fire emergency training center SEGANOSA in the municipality of *Salvaterra de Miño* (Spain) with a controlled fire. This session aim was to test the wildfire land detection sensor, based on infrared imagery, and the alarm dissemination. Figure 7.14 illustrates an infrared still image from the ground fire detection sensor video stream during the test. Figure 7.15 depicts the software response to the event: On the left,

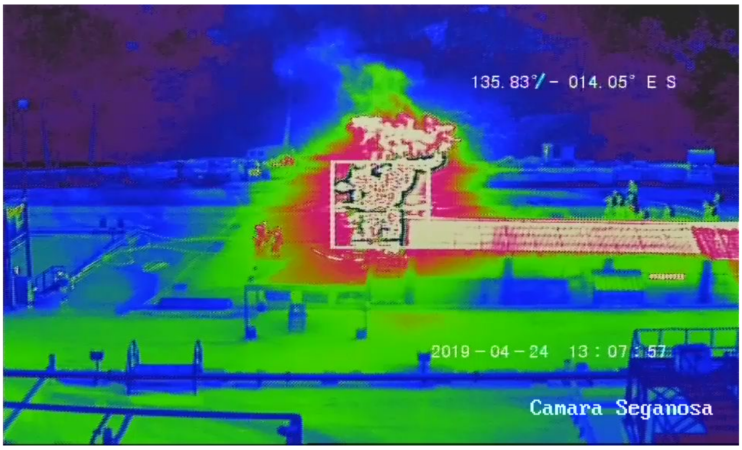


Figure 7.14: Still image of a real fire from the infrared video feed provided by the wildfire ground sensor located in the fire emergency training center

the interface of the satellite ground station shows the location of the alarm over a map, and on the right the predicted wildfire is displayed on the SAOP HMI.

7.3.2.2 Porto demonstration campaign

A second demonstration campaign was held in the location depicted in Figure 7.16, near Porto (Portugal), on May 23th, 2019. The goal was to test the operation of SAOP during the RUN and MONITOR scenarios: confirmation of a fire alarm, and mapping of a wildfire perimeter. In both stages, an UAV operated by the University of Porto has been used.

Because of the impossibility of having a real fire in this location, the realistic hybrid simulation framework introduced in section 7.2 to produce a synthetic wildfire. Due to communication issues with the network on-site between the UAVs and SuperSAOP, the mapping module ran off-board rather than on the embedded UAV computer. Interestingly, this problem confirms the utility of modular dynamic architectures so the system can work despite unexpected situations.

A live demonstration in 3 phases showcasing the scenarios RUN and MONITOR was performed for audience composed of FireRS project stakeholders and University of Porto representatives. The fist stage consisted executing the scenario RUN with a UAV previously put manually into a flying loitering state. The second and third stages demonstrated the scenario MONITOR over the front of the simulated wildfire 30 min and 1 h after the ignition. Figure 7.17 depicts the synthetic alarm along the expected wildfire forecast; and Figure 7.19 and Figure 7.18, the MONITOR scenario during the execution of the second and third stages respectively.

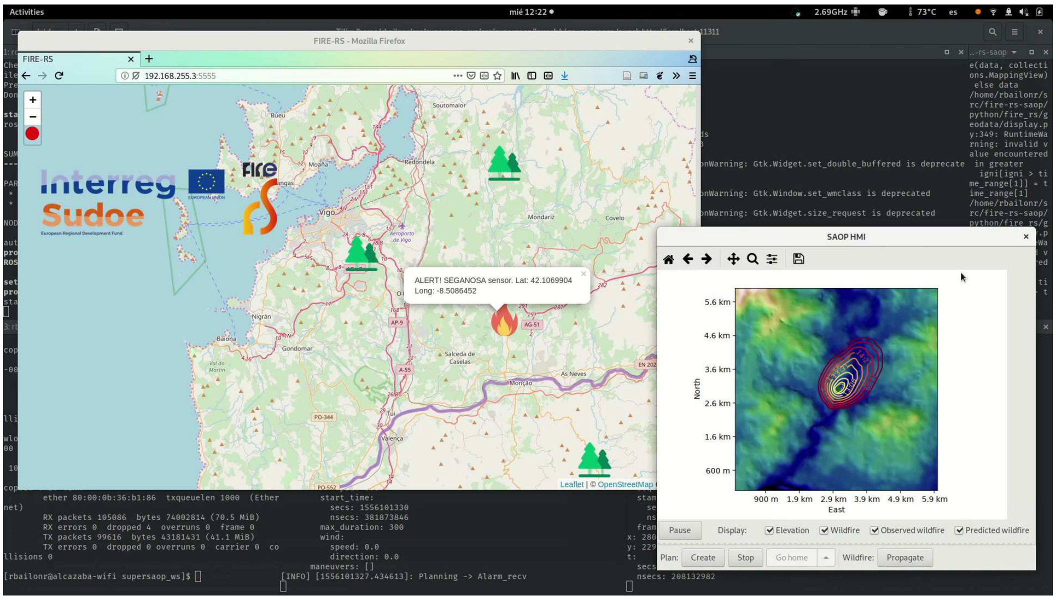
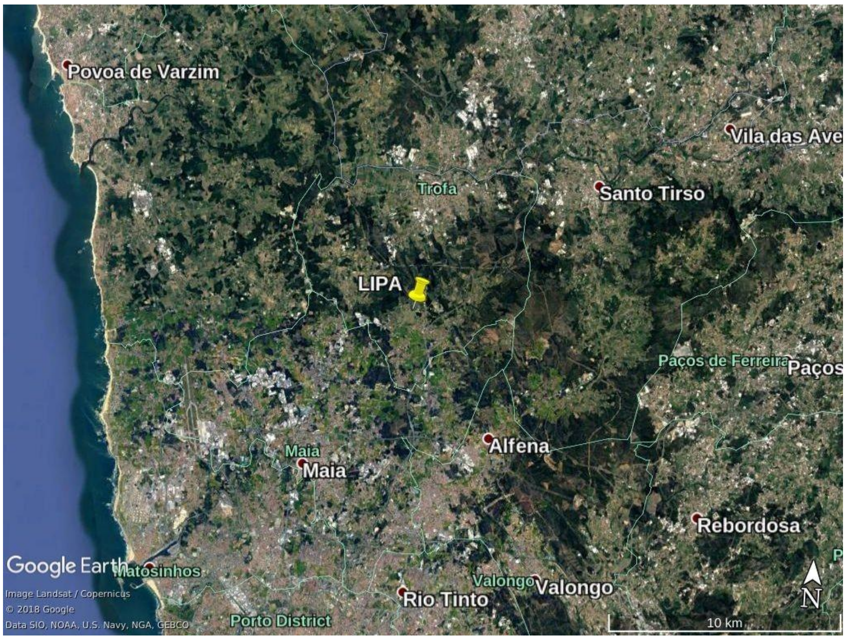


Figure 7.15: Screenshot of the satellite communication system interface displaying a wildfire alarm (left) and the SAOP HMI showing the corresponding predicted wildfire map.

7.4 Conclusion

This chapter has described several aspects of the FireRS SAOP system integration, achieved thanks to an architecture based on the ROS middleware that aggregates the wildfire situation assessment and observation planning algorithms with a UAV control platform and a dedicated communications satellite. A supervision subsystem handles the interactions with an operator, and the FireRS SAOP architecture includes a simulation framework that allows testing the system in a mixed-reality environment with virtual and physical UAVs and a synthetic wildfire. System performance has been tested in demonstration campaigns in Spain and Portugal where different levels of realism were put in action: a full virtual setup and a hybrid configuration of real UAVs and a virtual wildfire. Those tests featured the Fire RS SAOP architecture under selected scenarios of initial wildfire detection and perimeter monitoring.



(a) Overview of the flight site within the Porto metropolitan area

(b) Detailed map of the flight site. The 500 m radius red circle represents the approximate area where the UAVs are allowed to operate. The *flame* tag marks the location of the simulated fire ignition.

Figure 7.16: Details of the demonstration flying field near the city of Porto (Portugal)

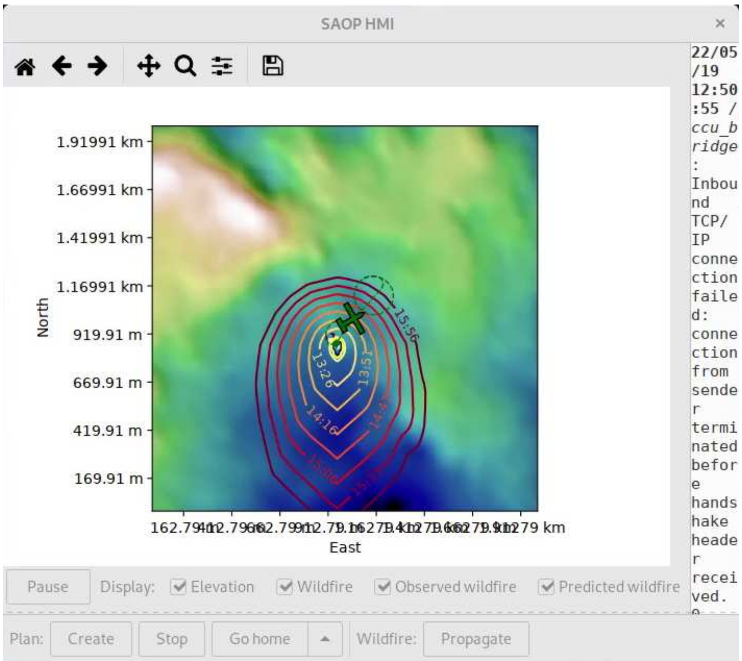


Figure 7.17: Scenario RUN situation during the Porto demonstration flight.

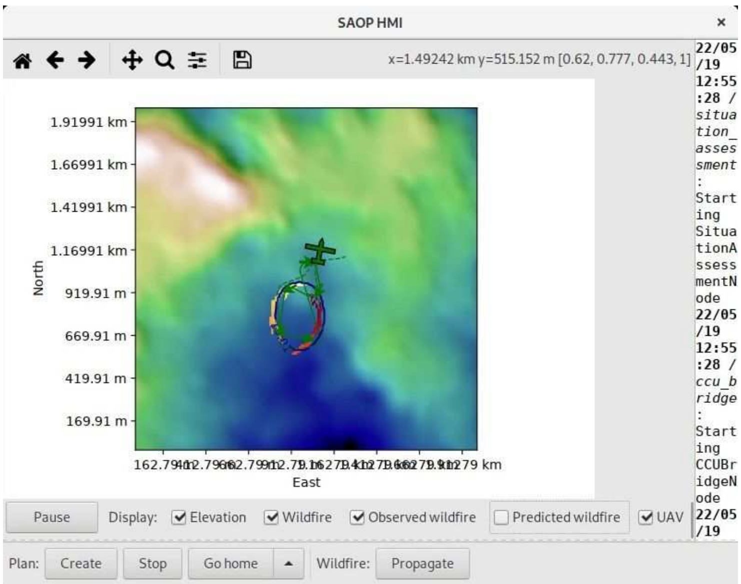


Figure 7.18: Screenshot of the SAOP HMI after running the MONITOR scenario in Portugal. The picture shows the UAV flight plan, executed path, the detected wildfire contour and the observed fire.

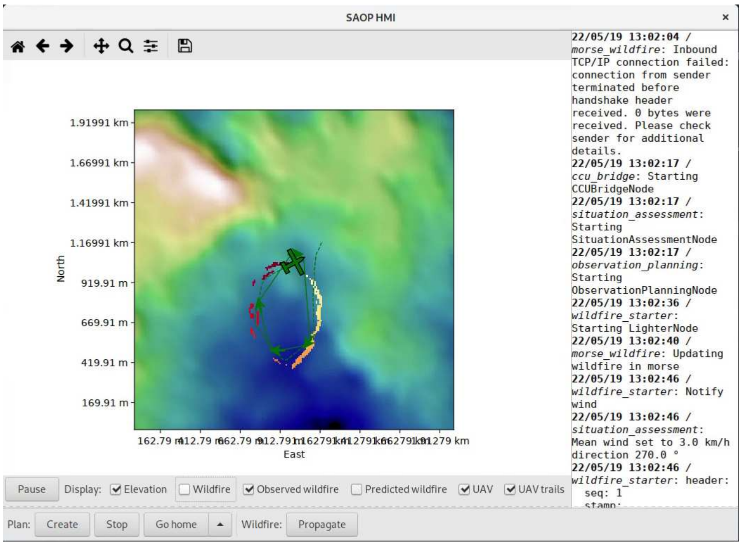


Figure 7.19: Observed wildfire map of the same fire depicted in Figure 7.18 30 minutes later

CHAPTER 8
Conclusion

8.1 Summary

This thesis has depicted the design of a wildfire monitoring system using fleets of unmanned aerial vehicles with the goal to provide wildland firefighters with complete up-to-date maps of an ongoing wildfire event.

First, chapter 2 presented a review of the literature about the use of autonomous unmanned aerial systems for wildfire remote sensing and introduced a taxonomy to classify these systems according to three criteria: awareness, decision and collaboration. The analysis revealed a path of overall increase of multiple UAV autonomous systems for wildfire remote sensing.

Chapters 3 and 4 depicted the necessary wildfire and UAV models respectively for the prediction of future fire spread and precise coordination of the drone fleet.

Chapter 5 described a wildfire situation assessment algorithm to estimate the current wildfire spread from partial observations of the fire perimeter. The algorithm, derived from an image warping algorithm, combines the map of observed fire locations, gathered by the fleet of UAVs and a network on ground sensors, and the predicted wildfire map to reconstruct the complete wildfire perimeter.

Chapter 6 introduced the definition of the wildfire observation problem, inspired by the Orienteering Problem, and a planning algorithm for fleets of UAVs, derived from the Variable Neighborhood Search metaheuristic.

Finally, chapter 7 depicted system integration. A software architecture using the ROS middleware orchestrates the operation of the wildfire situation assessment and observation planning algorithms described in previous chapters and the action of the UAVs. Also, this chapter introduced a tailor-made mixed-reality simulation infrastructure able to integrate virtual and physical drones within a synthetic fire environment. The system was tested with real UAVs thanks to this simulation framework.

8.2 Discussion and future work

This section provides a critical analysis of the work presented in this thesis with a focus on potential improvements to some aspects of the monitoring system in order to add more features or improve their performance. These improvements mainly pertain to the following topics:

- Tackling the third dimension, which mainly impacts the motion model[1],
- Better mapping and data assimilation process. This is in particular required to define a more efficient model to assess the utility of observations,
- Improvement to the planning scheme,
- Towards a more autonomous system, which mainly requires the definition of a more complex supervisor,
- and finally, but this pertains more to engineering activities, defining a more complete and more realistic simulation infrastructure.

The following section gives hints on how these improvements could be handled.

8.2.1 Computation of 3D flight trajectories

The wildfire monitoring concept introduced in this thesis deliberately omits the third dimension (the altitude) for the UAV motion model and the observation plans. This is a reasonable assumption given that current fixed-wing small and micro UAVs are not very capable of performing dynamic trajectories involving altitude changes. Typical remote sensing plans stay over a horizontal plane at a nominal height and in a few cases only, UAV alter flight altitude between discrete levels. This is due to the nature of fixed-wing aircrafts that are not as able to climb up like rotary-wing UAVs do and the limitations on the autonomous flight controllers that can not make them perform advanced maneuvers safely. Nonetheless, it is reasonable to expect than in the next years hardware and software technologies will improve to leverage UAV motion and providing more autonomy. Improved control algorithms will be implemented, and more capable aircraft designs will be widely available, e.g. convertible VTOL[2] UAVs.

Still, we cannot always assume that the UAVs can fly high enough so that terrain relief is not obstacle, particularly in mountainous regions, which are the cases where a system such as the FireRS system would be very relevant. The benefits of 3D wildfire monitoring plans mainly pertain to the flexibility for the fire observations: when an overview of the wildfire situation is required, UAVs can climb up to extend the vision range (with low mapping resolution), and conversely, vehicles can fly at lower altitudes to produce more precise localized maps, which could yield the estimation of relevant information such as flames height.

The extension of the Dubins paths to consider 3D trajectories is not cumbersome, but the generation of three-dimensional monitoring plans is far more complex than 2D ones because the motion planning search space is larger and the inaccessible flight zones impose more constraints.

Dubins 3D trajectories in a collision-free environment are of three types with respect to the altitude change between the origin and destination waypoints. As

[1] and the UAV flight control of course, but this is less related to our concern.
[2] Vertical Take Off and Landing

illustrated by Figure 8.1, if the end point elevation is reachable using the UAV climb rate without exceeding its limits, the result is a *low altitude* trajectory with the shape of a 2D Dubins path and constant climb speed. Otherwise, whether the altitude gain rate of the UAV is not sufficient, the Dubins trajectory must be elongated so the UAV is able to reach the desired altitude. Interestingly, in this case, every trajectory between the two points is a time-optimal path provided by the airplane climb speed is kept to the maximum at all times [Chitsaz 2007]. Paths of the *high altitude* class are usually extended by adding a helical altitude-gain section at the beginning. *Medium altitude* trajectories only add a circular portion section because they require an intermediate elongation between the low and high altitude types.

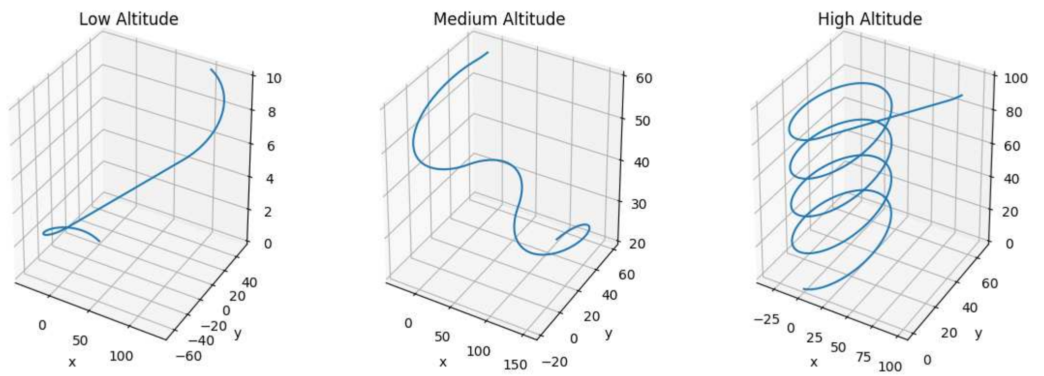


Figure 8.1: Illustration of the three types of Dubins 3D paths. The choice is derived from the difference in altitude between the origin and destination points.

If topographic obstacles are considered, every altitude in a continuous z-axis contains different patches of inaccessible terrain, defined as the locations were the effective flight altitude is below an arbitrary flight floor above ground level. Likewise, UAVs have defined flight ceilings because of technical and legal restrictions. Defining a restricted amount of discrete flight levels can help reduce the search space. Still, the observation planning algorithm must perform non-trivial motion planning in a cluttered environment between the independent observation trajectory waypoints.

A promising approach to plan 3D UAV paths dealing with mountainous environments is [Filippis 2012]. This publication introduces a variant of the A* path search algorithm to generate paths of the low altitude type, but its concept does not consider the medium and high altitude trajectories. Additionally, its implementation imposes an additional computation cost that reduces the feasibility of sampling when searching for the optimal wildfire observation plan.

[Váňa 2017] considers a data collection problem in a 3D environment with several common elements to the wildfire observation problem. The paper proposes a planning algorithm for the Dubins Airplane Orieenteering Problem based on a Randomized Variable Neighborhood Search (RVNS) method. However, the proposed approach does not scale well to the large number of observable cells the wildfire monitoring problem has.

8.2.2 Improvements to wildfire situation assessment

As it was briefly discussed in chapter 5, data assimilation algorithms to correct wildfire forecasts from observations have recently appeared in the literature [Rochoux 2013, Rios 2016]. The proposed algorithms are very promising to improve the parameters of forward propagation models that cannot be measured, and by that means to produce better wildfire predictions. Unfortunately, these algorithms are not yet available as an off-the-shelf components and the required computational power and execution time exceeds reasonable limits for real time usage. Still, the foreseeable future should bring the technological advances allowing deep integration of data assimilation into autonomous wildfire monitoring.

The definition of the utility function, exploited by the observation planner to assess the interest on perceiving a cell, could also benefit from a deeper integration with the situation assessment process. Currently, the utility map is initialized on the basis of the expected fire rate of spread to prioritize faster propagating areas that are potentially more dangerous as well. Then the adequacy of observation plans is calculated with respect to this base estimate, and previous monitoring missions, without acknowledging their fitness to improve the situation assessment process. Instead, if the algorithm that combines the observed and predicted wildfire maps was able to generate an estimate of the error on the assessment, the base utility map could be updated based on these results so the observation missions are given more priority to areas difficult to predict. Assessing errors during the fire mapping process can be straightforwardly done using the well know model of uncertainty grid maps in robotics, but the fusion of these data structure with the prediction of a fire model requires that the fire model itself is able to provide commensurable uncertainty estimates.

8.2.3 Planning improvements for long-lasting UAV wildfire monitoring missions

Wildfires are events lasting from a few hours to several days and during this time monitoring missions have to be managed considering together fire magnitude and expected UAV availability and endurance.

The VNS-based planner depicted in this thesis is adapted to a time scale in the order of hours. It is possible to manage UAV endurance by setting time windows, and maximum trajectory duration, but without dedicated human intervention the planner has an overall greedy behavior that makes use of UAVs as much as possible and as early as possible. This is an expected design considering firefighters need to gather rapidly the most possible amount of data and the utility function driving the planning process ensures this objective is accomplished. Given that the wildfire perimeter is always expanding, unless extinguished, there is always some location to visit that increases the overall plan success score confirming the algorithm greediness.

The problem with this behavior is that on the long term, without the operator mindful intervention to contain exploration, UAV endurance can be depleted too early leaving out the resources to monitor the wildfire afterwards. Ideally, there should be a setting to tweak resource allocation and provide a heuristic feedback on what would be the available autonomy of the fleet of UAVs in the long run. Then operators could decide whether monitoring plans can allocate resources aggressively, or on conservative many to protect UAV endurance. This long-life vision of the wildfire monitoring mission, does not take tough the current observation planner obsolete but puts it in a lower coordination level behind a bigger long-term meta-planning algorithm.

This kind of high-level or hierarchical coordination scheme is far beyond the main scope of this thesis, but future work of a successor of the FireRS project should evaluate this concept. The work of [Wu 2016] provides some insights about possible follow-ups.

Adaptive wildfire monitoring The current approach to plan wildfire observations is based on the assumption that the wildfire forecast is mostly correct and valid in future for around one hour. However, due to the inherent uncertain nature of wildfires and modeling errors discussed in chapter 3, the real wildfire growth may diverge from the predicted one.

In case of small discrepancies, UAVs continue now to execute the observation plan without realizing that the trajectory is not optimal. Later on, once the divergence is noticed, the observation planner refines the original plan, thus correcting the unsuitable paths according to the updated wildfire map. However, if the current wildfire perimeter estimation is far from reality, the observation plans are not able to actually map the fire because they are derived from the wrong assumptions. A better approach would be to grant UAVs more autonomy to give up their current mission when it is no longer good and correct their course according to their perception of the wildfire perimeter.

Future wildfire monitoring platforms may have increased active perception capabilities to plan their own trajectory directly from observations. Instead of providing to every UAV a trajectory defined as sequence of waypoints generated by an external observation planner, missions will be described as higher level commands such as "*follow this perimeter*" or "*guard this area*". The result of this new monitoring strategy, that could be depicted as *adaptive wildfire monitoring*, is an upgrade of the system decisional autonomy to the *adaptive autonomous planning* level, as it was defined in chapter 2.

8.2.4 System supervision

As of today, the supervision sub-system is mainly playing the role of a human-machine interface where the operator, an expert on wildland firefighting, has to issue high level commands. While this allows for some level of operational autonomy, as the user is liberated from specific tasks like creating flight plans and manually processing fire observations, decisions like when to start the plans and how many UAVs are necessary regarding long-term monitoring objectives are still required. Future work on the FireRS system should address this issue by improving the autonomy of the supervision module so fewer tasks and decisions are left to the users.

8.2.5 Robust integration of independent simulators

The simulation of complex systems, where many entities of diverse nature participate in the environment, require thoughtful integration of specialized simulation software because no independent simulator can model the entire world by itself. As a result, the interconnection of these components within the main simulation loop impose the consideration of a dedicated orchestration infrastructure to coordinate the disparities in simulation accuracy, interaction and time constraints. Additionally, if mixed-reality simulation is considered, time management becomes essential as hard real-time restrictions are dictated.

The proposed wildfire monitoring simulation infrastructure, depicted in chapter 7, has been designed on an ad-hoc basis to fulfill the requirements of the Fire RS project regarding the demonstration scenarios for which testing over a real wildfire was infeasible. Coordination of the different platforms that take part in the simulation —SAOP, Dune, Neptus, Morse and WindNinja— within the current context and expectations is relatively simple, but has not been designed to follow a formal approach to ensure simulation integrity and repeatability: The simulation infrastructure presented in this thesis is sufficient as a proof-of-concept and with respect to the main objectives of this thesis, but cannot be easily extended to other setups. Future work should consider a general and more robust approach to tackle the interconnection and orchestration of the independent simulators.

A solution to the problem of orchestrating distributed simulations has already been considered and studied by the community. The IEEE 1516 High Level Architecture (HLA) standard [iee 2010] describes the rules to build diverse simulation infrastructures. The HLA standard prescribes a centralized component model with a *Run-time Infrastructure (RTI)* program orchestrating independent simulators, *Federates*, according to a *Federation Object Model* that defines the way data is exchanged. This standard is popular in industrial and military contexts, but it has not been widely accepted within the scientific community because it is cumbersome to implement and there are not many free-software, actively developed, RTIs.

Bibliography

[Albini 1976] Frank A Albini. *Estimating Wildfire Behavior and Effects*. Technical Report INT-GTR-30, U.S. Department of Agriculture, Forest Service, Intermountain Research Station, Ogden, UT, 1976. (Cited in page 35.)

[Ambrosia 2003] VG Ambrosia, SS Wegener, DV Sullivan, SW Buechel, SE Dunagan, JA Brass, J Stoneburner and SM Schoenung. *Demonstrating UAV-Acquired Real-Time Thermal Data over Fires*. Photogrammetric Engineering & Remote Sensing, vol. 69, no. 4, pages 391–402, April 2003. (Cited in pages 11 and 21.)

[Ambrosia 2011] V. G. Ambrosia, S. Wegener, T. Zajkowski, D. V. Sullivan, S. Buechel, F. Enomoto, B. Lobitz, S. Johan, J. Brass and E. Hinkley. *The Ikhana Unmanned Airborne System (UAS) Western States Fire Imaging Missions: From Concept to Reality (2006-2010)*. Geocarto International, vol. 26, no. 2, SI, pages 85–101, 2011. (Cited in pages 12 and 21.)

[Anderson 1983] H. E. Anderson. *Predicting Wind-Driven Wild Land Fire Size and Shape*. USDA Forest Service Research Paper INT (USA), 1983. (Cited in pages 39 and 40.)

[Anderson 2013] Ross P. Anderson and Dejan Milutinović. *The Dubins Traveling Salesperson Problem With Stochastic Dynamics*. In Volume 2: Control, Monitoring, and Energy Harvesting of Vibratory Systems; Cooperative and Networked Control; Delay Systems; Dynamical Modeling and Diagnostics in Biomedical Systems; Estimation and Id of Energy Systems; Fault Detection; Flow and Thermal Systems; Haptics and Hand Motion; Human Assistive Systems and Wearable Robots; Instrumentation and Characterization in Bio-Systems; Intelligent Transportation Systems; Linear Systems and Robust Control; Marine Vehicles; Nonholonomic Systems, Dynamic Systems and Control Conference, Palo Alto, California, USA, October 2013. American Society of Mechanical Engineers. (Cited in page 80.)

[Andrews 2011] Patricia L. Andrews, Faith Ann Heinsch and Luke Schelvan. *How to Generate and Interpret Fire Characteristics Charts for Surface and Crown Fire Behavior*. Technical Report RMRS-GTR-253, U.S. Department of Agriculture, Forest Service, Rocky Mountain Research Station, Ft. Collins, CO, 2011. (Cited in page 36.)

[Andrews 2014] Patricia L. Andrews. *Current Status and Future Needs of the BehavePlus Fire Modeling System*. International Journal of Wildland Fire, vol. 23, no. 1, pages 21–33, February 2014. (Cited in pages 35 and 37.)

[Archetti 2007] Claudia Archetti, Alain Hertz and Maria Grazia Speranza. *Metaheuristics for the Team Orienteering Problem*. Journal of Heuristics, vol. 13, no. 1, pages 49–76, February 2007. (Cited in page 79.)

[Bailon-Ruiz 2018] Rafael Bailon-Ruiz, Arthur Bit-Monnot and Simon Lacroix. *Planning to Monitor Wildfires with a Fleet of UAVs*. In 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pages 4729–4734, Madrid, Spain, October 2018. IEEE. (Cited in pages 5 and 22.)

[Bailon-Ruiz 2020] Rafael Bailon-Ruiz and Simon Lacroix. *Wildfire Remote Sensing with UAVs: A Review from the Autonomy Point of View*. In International Conference on Unmanned Aircraft Systems (ICUAS 2020), September 2020. (Cited in page 5.)

[Beachly 2018] Evan Beachly, Carrick Detweiler, Sebastian Elbaum, Brittany Duncan, Carl Hildebrandt, Dirac Twidwell and Craig Allen. *Fire-Aware Planning of Aerial Trajectories and Ignitions*. In 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pages 685–692, Madrid,Spain, 2018. IEEE. (Cited in page 10.)

[Berndt 2004] Jon Berndt. *JSBSim: An Open Source Flight Dynamics Model in C++*. In AIAA Modeling and Simulation Technologies Conference and Exhibit, Providence, Rhode Island, August 2004. American Institute of Aeronautics and Astronautics. (Cited in page 112.)

[Bit-Monnot 2018] Arthur Bit-Monnot, Rafael Bailon-Ruiz and Simon Lacroix. *A Local Search Approach to Observation Planning with Multiple Uavs*. In Twenty-Eighth International Conference on Automated Planning and Scheduling, June 2018. (Cited in pages 4, 22, and 90.)

[Bookstein 1989] F.L. Bookstein. *Principal Warps: Thin-Plate Splines and the Decomposition of Deformations*. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 11, no. 6, pages 567–585, June 1989. (Cited in page 70.)

[Bookstein 1991] Fred L. Bookstein. *Thin-Plate Splines and the Atlas Problem for Biomedical Images*. In Alan C. F. Colchester and David J. Hawkes, editors, Information Processing in Medical Imaging, Lecture Notes in Computer Science, pages 326–342, Berlin, Heidelberg, 1991. Springer. (Cited in page 69.)

[Bresenham 1965] Jack E. Bresenham. *Algorithm for Computer Control of a Digital Plotter*. IBM Systems journal, vol. 4, no. 1, pages 25–30, 1965. (Cited in page 67.)

[Broomhead 1988] David S. Broomhead and David Lowe. *Radial Basis Functions, Multi-Variable Functional Interpolation and Adaptive Networks*. Technical Report, Royal Signals and Radar Establishment Malvern (United Kingdom), 1988. (Cited in page 70.)

[Büttner 2014] György Büttner. *CORINE Land Cover and Land Cover Change Products*. In Ioannis Manakos and Matthias Braun, editors, Land Use and Land Cover Mapping in Europe: Practices & Trends, Remote Sensing and Digital Image Processing, pages 55–74. Springer Netherlands, Dordrecht, 2014. (Cited in page 44.)

[Casbeer 2006] David W. Casbeer, Derek B. Kingston, Randal W. Beard and Timothy W. McLain. *Cooperative Forest Fire Surveillance Using a Team of Small Unmanned Air Vehicles*. International Journal of Systems Science, vol. 37, no. 6, pages 351–360, May 2006. (Cited in pages 13, 18, and 21.)

[Chao 1996] I-Ming Chao, Bruce L. Golden and Edward A. Wasil. *The Team Orienteering Problem*. European Journal of Operational Research, vol. 88, no. 3, pages 464–474, February 1996. (Cited in page 79.)

[Chitsaz 2007] H. Chitsaz and S. M. LaValle. *Time-Optimal Paths for a Dubins Airplane*. In 2007 46th IEEE Conference on Decision and Control, pages 2379–2384, December 2007. (Cited in page 127.)

[Christensen 2015] B. R. Christensen. *Use of UAV or Remotely Piloted Aircraft and Forward-Looking Infrared in Forest, Rural and Wildland Fire Management: Evaluation Using Simple Economic Analysis*. New Zealand Journal of Forestry Science, vol. 45, September 2015. (Cited in page 10.)

[Ciullo 2018] Vito Ciullo, Lucile Rossi, Tom Toulouse and Antoine Pieri. *Fire Geometrical Characteristics Estimation Using a Visible Stereovision System Carried by Unmanned Aerial Vehicle*. In 2018 15th International Conference on Control, Automation, Robotics and Vision (ICARCV), pages 1216–1221, Singapore, 2018. IEEE. (Cited in pages 10 and 22.)

[Clough 2002] Bruce Clough. *Metrics, Schmetrics! How Do You Track a UAV's Autonomy?* In 1st Technical Conference and Workshop on Unmanned Aerospace Vehicles, Portsmouth, Virginia, May 2002. American Institute of Aeronautics and Astronautics. (Cited in page 15.)

[Collins 1998] Robert T Collins, Yanghai Tsin, J Ryan Miller and Alan J Lipton. *Using a DEM to Determine Geospatial Object Trajectories*. In DARPA Image Understanding Workshop, volume 1, page 15, 1998. (Cited in page 67.)

[Degroote 2016] Arnaud Degroote, Pierrick Koch and Simon Lacroix. *Integrating Realistic Simulation Engines within the MORSE Framework*. In 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pages 2723–2728. IEEE, 2016. (Cited in page 113.)

[Dijkstra 1959] Edsger W. Dijkstra. *A Note on Two Problems in Connexion with Graphs*. Numerische mathematik, vol. 1, no. 1, pages 269–271, 1959. (Cited in page 44.)

[Dovis 2001] F Dovis, L Lo Presti, E Magli, P Mulassano and G Olmo. *Stratospheric Platforms: A Novel Technological Support for Earth Observation and Remote Sensing Applications*. In Fujisada, H and Lurie, JB and Weber, K, editor, Sensors, Systems and Next-Generation Satellites V, volume 4540 of *Proceedings of SPIE*, pages 402–411, Bellingham, WA, USA, 2001. Spie-Int Soc Optical Engineering. (Cited in pages 13 and 21.)

[Dubins 1957] L. E. Dubins. *On Curves of Minimal Length with a Constraint on Average Curvature, and with Prescribed Initial and Terminal Positions and Tangents*. American Journal of Mathematics, vol. 79, no. 3, pages 497–516, 1957. (Cited in page 53.)

[Echeverria 2012] Gilberto Echeverria, Séverin Lemaignan, Arnaud Degroote, Simon Lacroix, Michael Karg, Pierrick Koch, Charles Lesire and Serge Stinckwich. *Simulating Complex Robotic Scenarios with MORSE*. In David Hutchison, Takeo Kanade, Josef Kittler, Jon M. Kleinberg, Friedemann Mattern, John C. Mitchell, Moni Naor, Oscar Nierstrasz, C. Pandu Rangan, Bernhard Steffen, Madhu Sudan, Demetri Terzopoulos, Doug Tygar, Moshe Y. Vardi, Gerhard Weikum, Itsuki Noda, Noriaki Ando, Davide Brugali and James J. Kuffner, editors, Simulation, Modeling, and Programming for Autonomous Robots, volume 7628, pages 197–208. Springer Berlin Heidelberg, Berlin, Heidelberg, 2012. (Cited in page 113.)

[Esposito 2007] Francesco Esposito, Giancarlo Rufino, Antonio Moccia, Paolo Donnarumma, Marco Esposito and Vincenzo Magliulo. *An Integrated Electro-Optical Payload System for Forest Fires Monitoring from Airborne Platform*. In 2007 IEEE Aerospace Conference, pages 2015–2027, Big Sky, Montana, 2007. IEEE. (Cited in pages 12 and 21.)

[Filippi 2009] Jean Baptiste Filippi, Frédéric Bosseur, Céline Mari, Christine Lac, Patrick Le Moigne, Bénédicte Cuenot, Denis Veynante, Daniel Cariolle and Jacques-Henri Balbi. *Coupled Atmosphere-Wildland Fire Modelling*. Journal of Advances in Modeling Earth Systems, vol. 1, no. 4, 2009. (Cited in pages 34 and 41.)

[Filippis 2012] Luca De Filippis, Giorgio Guglieri and Fulvia Quagliotti. *Path Planning Strategies for UAVS in 3D Environments*. Journal of Intelligent & Robotic Systems, vol. 65, no. 1-4, pages 247–264, January 2012. (Cited in page 127.)

[Finney 1998] Mark A. Finney. *FARSITE: Fire Area Simulator-Model Development and Evaluation*. Technical Report RMRS-RP-4, U.S. Department of Agriculture, Forest Service, Rocky Mountain Research Station, Ft. Collins, CO, 1998. (Cited in pages 35, 37, and 41.)

[Finney 2002] Mark A Finney. *Fire Growth Using Minimum Travel Time Methods*. Canadian Journal of Forest Research, vol. 32, no. 8, pages 1420–1424, August 2002. (Cited in page 44.)

[Forthofer 2014] Jason M. Forthofer, Bret W. Butler and Natalie S. Wagenbrenner. *A Comparison of Three Approaches for Simulating Fine-Scale Surface Winds in Support of Wildland Fire Management. Part I. Model Formulation and Comparison against Measurements*. International Journal of Wildland Fire, vol. 23, no. 7, pages 969–981, November 2014. (Cited in page 43.)

[Fromm 2006] Michael Fromm, Andrew Tupper, Daniel Rosenfeld, René Servranckx and Rick McRae. *Violent Pyro-Convective Storm Devastates Australia's Capital and Pollutes the Stratosphere*. Geophysical Research Letters, vol. 33, no. 5, 2006. (Cited in page 33.)

[Gisborne 1927] H. T. Gisborne. *The Objectives of Forest Fire-Weather Research*. Journal of Forestry, vol. 25, no. 4, pages 452–456, April 1927. (Cited in page 34.)

[Golden 1987] Bruce L. Golden, Larry Levy and Rakesh Vohra. *The Orienteering Problem*. Naval Research Logistics (NRL), vol. 34, no. 3, pages 307–318, 1987. (Cited in page 78.)

[Haksar 2018] Ravi N. Haksar and Mac 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), IEEE International Conference on Intelligent Robots and Systems, pages 1067–1074, Madrid,Spain, 2018. IEEE. (Cited in pages 10, 15, 19, and 22.)

[Hansen 2001] Pierre Hansen and Nenad Mladenović. *Variable Neighborhood Search: Principles and Applications*. European Journal of Operational Research, vol. 130, no. 3, pages 449–467, May 2001. (Cited in page 81.)

[Hansen 2010] Pierre Hansen, Nenad Mladenović and José A. Moreno Pérez. *Variable Neighbourhood Search: Methods and Applications*. Annals of Operations Research, vol. 175, no. 1, pages 367–407, March 2010. (Cited in page 81.)

[Hawley 1926] L. F. Hawley. *Theoretical Considerations Regarding Factors Which Influence Forest Fires*. Journal of Forestry, vol. 24, no. 7, pages 756–763, November 1926. (Cited in page 34.)

[Howden 2013] David J. Howden. *Fire Tracking with Collective Intelligence Using Dynamic Priority Maps*. In 2013 IEEE Congress on Evolutionary Computation (CEC), pages 2610–2617, Cancun, Mexico, 2013. IEEE. (Cited in pages 14 and 21.)

[Huang 2007] Hui-Min Huang. *Autonomy Levels for Unmanned Systems (ALFUS) Framework: Safety and Application Issues*. In Proceedings of the 2007 Workshop on Performance Metrics for Intelligent Systems, PerMIS '07, pages 48–53, Washington, D.C., USA, 2007. ACM. (Cited in page 16.)

[iee 2010] *IEEE Standard for Modeling and Simulation (M & S) High Level Architecture (HLA)– Framework and Rules*. IEEE Std 1516-2010 (Revision of IEEE Std 1516-2000), pages 1–38, August 2010. (Cited in page 130.)

[Julian 2019] Kyle D. Julian and Mykel J. Kochenderfer. *Distributed Wildfire Surveillance with Autonomous Aircraft Using Deep Reinforcement Learning*. Journal of Guidance Control and Dynamics, vol. 42, no. 8, pages 1768–1778, August 2019. (Cited in pages 15 and 22.)

[Kantor 1992] Marisa G. Kantor and Moshe B. Rosenwein. *The Orienteering Problem with Time Windows*. Journal of the Operational Research Society, vol. 43, no. 6, pages 629–635, June 1992. (Cited in page 79.)

[Keane 2015] Robert E. Keane. Wildland Fuel Fundamentals and Applications. Springer International Publishing, Cham, 2015. (Cited in pages 30 and 31.)

[Lacroix 2002] S. Lacroix, I. K. Jung and A. Mallet. *Digital Elevation Map Building from Low Altitude Stereo Imagery*. Robotics and Autonomous Systems, vol. 41, no. 2, pages 119–127, November 2002. (Cited in page 67.)

[Lewyckyj 2007] N. Lewyckyj, J. Biesemans and J. Everaerts. *OSIRIS: A European Project Using a High Altitude Platform for Forest Fire Monitoring*. In Guarascio, M and Brebbia, CA and Garzia, F, editor, Safety and Security Engineering II, volume 94 of *WIT Transactions On The Built Environment*, pages 205–213, Southampton, England, 2007. WIT Press/Computational Mechanics Publications. (Cited in pages 12 and 21.)

[Lin 2015] Zhongjie Lin and Hugh H. T. Liu. *Enhanced Cooperative Filter for Wildfire Monitoring*. In 54th IEEE Conference on Decision and Control (CDC), pages 3075–3080, Osaka, Japan, 2015. IEEE. (Cited in pages 15 and 22.)

[Lin 2018] Zhongjie Lin and Hugh H. T. Liu. *Topology-Based Distributed Optimization for Multi-UAV Cooperative Wildfire Monitoring*. Optimal Control Applications and Methods, vol. 39, no. 4, pages 1530–1548, 2018. (Cited in pages 15 and 22.)

[Lin 2019] Zhongjie Lin, Hugh H. T. Liu and Mike Wotton. *Kalman Filter-Based Large-Scale Wildfire Monitoring With a System of UAVs*. IEEE Transactions on Industrial Electronics, vol. 66, no. 1, pages 606–615, January 2019. (Cited in pages 15 and 22.)

[Lopes 2002] A. M. G. Lopes, M. G. Cruz and D. X. Viegas. *FireStation — an Integrated Software System for the Numerical Simulation of Fire Spread on Complex Topography*. Environmental Modelling & Software, vol. 17, no. 3, pages 269–285, 2002. (Cited in page 41.)

[López Peña 2016] F. López Peña, P. Caamaño, G. Varela, F. Orjales and A. Deibe. *Setting up a Mixed Reality Simulator for Using Teams of Autonomous UAVs in Air Pollution Monitoring*. International Journal of Sustainable Development and Planning, vol. 11, no. 4, pages 616–626, August 2016. (Cited in page 112.)

[Luo 2018] He Luo, Zhengzheng Liang, Moning Zhu, Xiaoxuan Hu and Guoqiang Wang. *Integrated Optimization of Unmanned Aerial Vehicle Task Allocation and Path Planning under Steady Wind*. PLOS ONE, vol. 13, no. 3, page e0194690, March 2018. (Cited in page 80.)

[Macharet 2014] Douglas G. Macharet and Mario F. M. Campos. *An Orientation Assignment Heuristic to the Dubins Traveling Salesman Problem*. In Ana L.C. Bazzan and Karim Pichara, editors, Advances in Artificial Intelligence – IBERAMIA 2014, Lecture Notes in Computer Science, pages 457–468. Springer International Publishing, 2014. (Cited in pages 80 and 88.)

[Martínez-de Dios 2005] J. Ramiro Martínez-de Dios, Luis Merino and Aníbal Ollero. *Fire Detection Using Autonomous Aerial Vehicles With Infrared And Visual Cameras*. In IFAC Proceedings Volumes, volume 38 of *16th IFAC World Congress*, pages 660–665, January 2005. (Cited in page 66.)

[Martínez-de Dios 2011] José Ramiro Martínez-de Dios, Luis Merino, Fernando Caballero and Anibal Ollero. *Automatic Forest-Fire Measuring Using Ground Stations and Unmanned Aerial Systems*. Sensors, vol. 11, no. 6, pages 6328–6353, June 2011. (Cited in pages 10, 13, and 21.)

[McGee 2005] Timothy G. McGee, Stephen Spry and J. Karl Hedrick. *Optimal Path Planning in a Constant Wind with a Bounded Turning Rate*. In AIAA Guidance, Navigation, and Control Conference and Exhibit, pages 1–11. Reston, VA, 2005. (Cited in page 54.)

[Merino 2005] Luis Merino, Fernando Caballero, J. R. Martínez-de Dios and Aníbal Ollero. *Cooperative Fire Detection Using Unmanned Aerial Vehicles*. In Proceedings of the 2005 IEEE International Conference on Robotics and Automation, pages 1884–1889, Barcelona, Spain, 2005. IEEE. (Cited in pages 13, 18, 20, and 21.)

[Merino 2012] Luis Merino, Fernando Caballero, J. Ramiro Martínez-de-Dios, Iván Maza and Aníbal Ollero. *An Unmanned Aircraft System for Automatic Forest Fire Monitoring and Measurement*. Journal of Intelligent & Robotic Systems, vol. 65, no. 1, pages 533–548, January 2012. (Cited in pages 13, 14, 19, 20, and 21.)

[Milgram 1994] Paul Milgram and Fumio Kishino. *A Taxonomy of Mixed Reality Visual Displays*. IEICE TRANSACTIONS on Information and Systems, vol. 77, no. 12, pages 1321–1329, 1994. (Cited in page 111.)

[Montemanni 2009] Roberto Montemanni and Luca Maria Gambardella. *An Ant Colony System for Team Orienteering Problems with Time Windows*. Foundation Of Computing And Decision Sciences, vol. 34, no. 4, page 287, 2009. (Cited in page 79.)

[Moulianitis 2019] V. C. Moulianitis, G. Thanellas, N. Xanthopoulos and Nikos A. Aspragathos. *Evaluation of UAV Based Schemes for Forest Fire Monitoring*. In Nikos A. Aspragathos, Panagiotis N. Koustoumpardis and Vassilis C. Moulianitis, editors, Advances in Service and Industrial Robotics, Mechanisms and Machine Science, pages 143–150, Cham, 2019. Springer International Publishing. (Cited in pages 13 and 16.)

[Nercellas 2019] Aarón Nercellas, José Miguel Lago and Fernando Aguado. *FIRE-RS DEMO-1 Definition*. Technical Report LUME1-12000-RPT-002, Universidade de Vigo, October 2019. (Cited in page 106.)

[Pěnička 2017] R. Pěnička, J. Faigl, P. Váňa and M. Saska. *Dubins Orienteering Problem*. IEEE Robotics and Automation Letters, vol. 2, no. 2, pages 1210–1217, April 2017. (Cited in page 80.)

[Pérez-Lissi 2018] Franco Pérez-Lissi, Fernando Aguado-Agelet, Antón Vázquez, Pablo Yañez, Pablo Izquierdo, Simon Lacroix, Rafael Bailon-Ruiz, Joao Tasso, Andre Guerra and Maria Costa. *FIRE-RS: Integrating Land Sensors, Cubesat Communications, Unmanned Aerial Vehicles and a Situation Assessment Software for Wildland Fire Characterization and Mapping*. In 69th International Astronautical Congress, Bremen, Germany, October 2018. (Cited in pages 5 and 22.)

[Pham 2017] Huy X. Pham, Hung M. La, David Feil-Seifer and Matthew Deans. *A Distributed Control Framework for a Team of Unmanned Aerial Vehicles for Dynamic Wildfire Tracking*. In Bicchi, A and Okamura, A, editor, 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pages 6648–6653, Vancouver, Canada, 2017. IEEE. (Cited in pages 15 and 22.)

[Pinto 2013] Jose Pinto, Paulo S. Dias, Ricardo Martins, Joao Fortuna, Eduardo Marques and Joao Sousa. *The LSTS Toolchain for Networked Vehicle Systems*. In OCEANS'13 MTS/IEEE Bergen, pages 1–9, Bergen, Norway, June 2013. IEEE. (Cited in page 104.)

[Price 1963] Derek John de Solla Price. Little science, big science. Columbia University Press, 1963. (Cited in page 8.)

[Protti 2007] Marco Protti and Riccardo Barzan. *UAV Autonomy-Which Level Is Desirable?-Which Level Is Acceptable? Alenia Aeronautica Viewpoint*. Technical Report, Alenia Aeronautica SpA, 2007. (Cited in page 16.)

[Rabinovich 2018] Sharon Rabinovich, Renwick E. Curry and Gabriel H. Elkaim. *Toward Dynamic Monitoring and Suppressing Uncertainty in Wildfire by Multiple Unmanned Air Vehicle System*. Journal of Robotics, 2018. (Cited in pages 14 and 22.)

[Reeves 2009] Matthew C. Reeves, Kevin C. Ryan, Matthew G. Rollins and Thomas G. Thompson. *Spatial Fuel Data Products of the LANDFIRE Project*. International Journal of Wildland Fire, vol. 18, no. 3, page 250, 2009. (Cited in page 44.)

[Restas 2006] Agoston Restas. *Forest Fire Management Supporting by UAV Based Air Reconnaissance Results of Szendro Fire Department, Hungary*. In First International Symposium on Environment Identities and Mediterranean Area, pages 84–88, Corte-Ajaccio, France, 2006. IEEE. (Cited in pages 12 and 21.)

[Rios 2016] O. Rios, E. Pastor, M. M. Valero and E. Planas. *Short-Term Fire Front Spread Prediction Using Inverse Modelling and Airborne Infrared Images*. International Journal of Wildland Fire, vol. 25, no. 10, pages 1033–1047, October 2016. (Cited in pages 63, 68, and 128.)

[Rochoux 2013] Mélanie C. Rochoux, Bénédicte Cuenot, Sophie Ricci, Arnaud Trouvé, Blaise Delmotte, Sébastien Massart, Roberto Paoli and Ronan Paugam. *Data Assimilation Applied to Combustion*. Comptes Rendus Mécanique, vol. 341, no. 1, pages 266–276, January 2013. (Cited in pages 63, 68, and 128.)

[Rollo 2015] Milan Rollo, Martin Selecky, Paul Losiewicz, John Reade and Nicholas Maida. *Framework for Incremental Development of Complex Unmanned Aircraft Systems*. In 2015 Integrated Communication, Navigation and Surveillance Conference (ICNS), pages 1–14, Herdon, VA, USA, April 2015. IEEE. (Cited in page 111.)

[Rothermel 1972] Richard C. Rothermel. *A Mathematical Model for Predicting Fire Spread in Wildland Fuels*. USDA Forest Service, vol. Research Paper INT-115, 1972. (Cited in pages 34 and 35.)

[Rothermel 1991] Richard C. Rothermel. *Predicting Behavior and Size of Crown Fires in the Northern Rocky Mountains*. Technical Report INT-RP-438, U.S. Department of Agriculture, Forest Service, Intermountain Research Station, Ogden, UT, 1991. (Cited in page 36.)

[Scott 2005] Joe H. Scott and Robert E. Burgan. *Standard Fire Behavior Fuel Models: A Comprehensive Set for Use with Rothermel's Surface Fire Spread Model*. Technical Report RMRS-GTR-153, U.S. Department of Agriculture, Forest Service, Rocky Mountain Research Station, Ft. Collins, CO, 2005. (Cited in pages 39 and 40.)

[Scott 2014] Andrew C. Scott, David M. J. S. Bowman, William J. Bond, Stephen J. Pyne and Martin E. Alexander. Fire on earth: An introduction. Wiley-Blackwell, 1st edition, January 2014. (Cited in pages 32 and 35.)

[Selecky 2017] Martin Selecky, Jan Faigl and Milan Rollo. *Mixed Reality Simulation for Incremental Development of Multi-UAV Systems*. In 2017 International Conference on Unmanned Aircraft Systems (ICUAS), pages 1530–1538, Miami, FL, USA, June 2017. IEEE. (Cited in page 111.)

[Siciliano 2008] Bruno Siciliano and Oussama Khatib, editors. Springer Handbook of Robotics. Springer, Berlin, 2008. (Cited in page 101.)

[Silberholz 2010] John Silberholz and Bruce Golden. *The Effective Application of a New Approach to the Generalized Orienteering Problem*. Journal of Heuristics, vol. 16, no. 3, pages 393–415, June 2010. (Cited in page 79.)

[Skeele 2016] Ryan C. Skeele and Geoffrey A. Hollinger. *Aerial Vehicle Path Planning for Monitoring Wildfire Frontiers*. In Wettergreen, DS and Barfoot, TD, editor, Field and Service Robotics: Results of the 10th International Conference, volume 113 of *Springer Tracts in Advanced Robotics*, pages 455–467, Toronto, Canada, 2016. (Cited in pages 12 and 21.)

[Sullivan 2009a] Andrew L. Sullivan. *Wildland Surface Fire Spread Modelling, 1990 - 2007. 1: Physical and Quasi-Physical Models*. International Journal of Wildland Fire, vol. 18, no. 4, page 349, 2009. (Cited in page 34.)

[Sullivan 2009b] Andrew L. Sullivan. *Wildland Surface Fire Spread Modelling, 1990 - 2007. 2: Empirical and Quasi-Empirical Models*. International Journal of Wildland Fire, vol. 18, no. 4, page 369, 2009. (Cited in page 34.)

[Sullivan 2009c] Andrew L. Sullivan. *Wildland Surface Fire Spread Modelling, 1990 - 2007. 3: Simulation and Mathematical Analogue Models*. International Journal of Wildland Fire, vol. 18, no. 4, page 387, 2009. (Cited in page 41.)

[Tang 2005] Hao Tang and Elise Miller-Hooks. *A Tabu Search Heuristic for the Team Orienteering Problem*. Computers & Operations Research, vol. 32, no. 6, pages 1379–1407, 2005. (Cited in page 79.)

[Tang 2015] Lina Tang and Guofan Shao. *Drone Remote Sensing for Forestry Research and Practices*. Journal of Forestry Research, vol. 26, no. 4, pages 791–797, December 2015. (Cited in pages 9 and 20.)

[Techy 2009] Laszlo Techy and Craig A. Woolsey. *Minimum-Time Path Planning for Unmanned Aerial Vehicles in Steady Uniform Winds*. Journal of Guidance, Control, and Dynamics, vol. 32, no. 6, pages 1736–1746, November 2009. (Cited in page 54.)

[Torresan 2017] Chiara Torresan, Andrea Berton, Federico Carotenuto, Salvatore Filippo Di Gennaro, Beniamino Gioli, Alessandro Matese, Franco Miglietta, Carolina Vagnoli, Alessandro Zaldei and Luke Wallace. *Forestry Applications of UAVs in Europe: A Review*. International Journal of Remote Sensing, vol. 38, no. 8-10, pages 2427–2447, May 2017. (Cited in page 9.)

[Toth 2002] Paolo Toth and Daniele Vigo, editors. The vehicle routing problem. SIAM Monographs on Discrete Mathematics and Applications. Society for Industrial and Applied Mathematics, Philadelphia, Pa, 2002. (Cited in page 78.)

[Tricoire 2010] Fabien Tricoire, Martin Romauch, Karl F. Doerner and Richard F. Hartl. *Heuristics for the Multi-Period Orienteering Problem with Multiple Time Windows*. Computers & Operations Research, vol. 37, no. 2, pages 351–367, 2010. (Cited in page 79.)

[Twidwell 2016] Dirac Twidwell, Craig R. Allen, Carrick Detweiler, James Higgins, Christian Laney and Sebastian Elbaum. *Smokey Comes of Age: Unmanned Aerial Systems for Fire Management*. Frontiers in Ecology and the Environment, vol. 14, no. 6, pages 333–339, August 2016. (Cited in pages 9 and 10.)

[Valavanis 2007] Kimon P. Valavanis, editor. Advances in unmanned aerial vehicles: State of the art and the road to autonomy. Number Vol. 33 in International Series on Intelligent Systems, Control and Automation: Science and Engineering. Springer, Dordrecht, 2007. (Cited in page 7.)

[Valero 2017] Mario M. Valero, Oriol Rios, Christian Mata, Elsa Pastor and Eulàlia Planas. *An Integrated Approach for Tactical Monitoring and Data-Driven Spread Forecasting of Wildfires*. Fire Safety Journal, vol. 91, no. Supplement C, pages 835–844, July 2017. (Cited in pages 10 and 22.)

[Váňa 2017] P. Váňa, J. Faigl, J. Sláma and R. Pěnička. *Data Collection Planning with Dubins Airplane Model and Limited Travel Budget*. In 2017 European Conference on Mobile Robots (ECMR), pages 1–6, September 2017. (Cited in page 127.)

[Vansteenwegen 2009a] Pieter Vansteenwegen, Wouter Souffriau, Greet Vanden Berghe and Dirk Van Oudheusden. *Metaheuristics for Tourist Trip Planning*. In Kenneth Sörensen, Marc Sevaux, Walter Habenicht and Martin Josef Geiger, editors, Metaheuristics in the Service Industry, Lecture Notes in Economics and Mathematical Systems, pages 15–31. Springer Berlin Heidelberg, Berlin, Heidelberg, 2009. (Cited in page 79.)

[Vansteenwegen 2009b] Pieter Vansteenwegen, Wouter Souffriau, Greet Vanden Berghe and Dirk Van Oudheusden. *Iterated Local Search for the Team Orienteering Problem with Time Windows*. Computers & Operations Research, vol. 36, no. 12, pages 3281–3290, December 2009. (Cited in page 79.)

[Vansteenwegen 2011] Pieter Vansteenwegen, Wouter Souffriau and Dirk Van Oudheusden. *The Orienteering Problem: A Survey*. European Journal of Operational Research, vol. 209, no. 1, pages 1–10, February 2011. (Cited in page 79.)

[von Wahl 2010] N. von Wahl, S. Heinen, H. Essen, W. Kruell, R. Tobera and I. Willms. *An Integrated Approach for Early Forest Fire Detection and Verification Using Optical Smoke, Gas and Microwave Sensors*. In Perona, G and Brebbia, CA, editor, Modelling, Monitoring and Management of Forest Fires II, volume 137 of *WIT Transactions on Ecology and the Environment*, pages 97+, Southampton, England, 2010. WIT PRESS. (Cited in pages 14, 19, and 21.)

[Wang 2008] Xia Wang, Bruce L. Golden and Edward A. Wasil. *Using a Genetic Algorithm to Solve the Generalized Orienteering Problem*. In The Vehicle Routing Problem: Latest Advances and New Challenges, pages 263–274. Springer, 2008. (Cited in page 79.)

[Webster 2010] Mark Webster and H. David Sheets. *A Practical Introduction to Landmark-Based Geometric Morphometrics*. The Paleontological Society Papers, vol. 16, pages 163–188, October 2010. (Cited in page 69.)

[Wu 2016] Guohua Wu, Witold Pedrycz, Haifeng Li, Manhao Ma and Jin Liu. *Coordinated Planning of Heterogeneous Earth Observation Resources*. IEEE Transactions on Systems, Man, and Cybernetics: Systems, vol. 46, no. 1, pages 109–125, January 2016. (Cited in page 129.)

[Yuan 2015] Chi Yuan, Youmin Zhang and Zhixiang Liu. *A Survey on Technologies for Automatic Forest Fire Monitoring, Detection, and Fighting Using Unmanned Aerial Vehicles and Remote Sensing Techniques*. Canadian Journal of Forest Research, vol. 45, no. 7, pages 783–792, July 2015. (Cited in page 9.)