Coventry University
Institute for Future Transport and Cities

Towards an autonomous wildfire suppression system based on swarms of self-organising drones

Paolo Grasso

Supervisor: Dr Mauro S. Innocente

A thesis submitted in partial fulfilment of the University's requirements for the degree of Doctor of Philosophy

March 13, 2022

Declaration

All sentences or passages quoted in this document from other people's work have been specifically acknowledged by clear cross-referencing to author, work and page(s). Any illustrations that are not the work of the author of this report have been used with the explicit permission of the originator and are specifically acknowledged. I understand that failure to do this amounts to plagiarism and will be considered grounds for failure.

Two main journal papers has been used in the writing of this thesis. These has been fruit of a collaboration with my DoS Dr Mauro S. Innocente. I hereby identify the publications and where such material has been used in this thesis. The detailed contributions list can be found in the attached Contributions Declaration document.

- *Physics-based model of wildfire propagation towards faster-than-real-time simulations*, doi: 10.1016/j.camwa.2020.05.009. The content of the paper has been used in Chapter 3.
- *Self-organising swarms of firefighting drones: Harnessing the power of collective intelligence in decentralised multi-robot systems*, doi: 10.1016/j.jocs.2019.04.009. The content of the paper has been used in Chapter 4.

Name: Paolo Grasso

Signature: Content removed on Data-Protection grounds

Date: March 13, 2022

Acknowledgement

I would like to thank the following people for helping with this research project:

My supervisor Dr Mauro S. Innocente for his impeccable guidance and support both professionally and personally throughout my PhD experience.

My best friend Daizy Rajput, with who I shared wonderful and challenging moments in these years, bringing both warming heart-felt support and sincere friendly advice in time of need.

Last but not least, my parents for the sacrifices they made throughout their lives to be able to support my intellectual and personal growth; they made this possible.

Abstract

This research aims at contributing in the development of a firefighting technology, which uses swarms of self-organising autonomous drones. Wildfires are indeed one of the most threatening cataclysms induced by climate change that endangers wildlife and human lives, and which devastates health, social, economic and environment spheres. Hence, the urge to find a new solution to tackle more efficiently and effectively critical wildfires by either extinguishing or containing them. The frontier of autonomous swarm robotics seems to provide good chances to succeed in this venture thanks to many strong points, for instance: the capability to work in hazardous areas, the robustness of the system with respect to loss of a few drones, and no need of highly trained pilots. The research presented in this thesis spans between various fields such as thermo-fluid-dynamics, swarm robotics, and Swarm-in-the-Loop (SwiL) simulation. More specifically the main research topics are wildfire propagation modelling, autonomous multi-agent robotics for firefighting application, stigmergic collision-avoidance systems for autonomous flight, and study and improvement of indoor positioning systems (IPSs). While the first three research subjects are clearly interlaced with each other, the last topic regarding IPS consists of some relevant theoretical and experimental work towards the development of a SwiL platform to test the fire suppression system and new self-organisation algorithms. The main contributions of the presented research are: a faster-than-real-time physics-based propagation model, namely FireProM-F; self-organisation algorithms for swarm of autonomous firefighting drones; a stigmergy-based collision-avoidance algorithm for autonomous swarm; a study of the precision, accuracy, and failure of UWB-based IPSs; and the development of a debiasing filter for the improvement of the IPS's accuracy.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

With climate change wildfires are one of the major and most threatening cataclysms that put wildlife and human lives in extreme danger because of both immediate damage and long term repercussions. A wildfire is an unplanned and uncontrolled vegetation fire which can have devastating health, social, economic and environmental impacts [1]. The deadliest wildfires in Portugal's history erupted in 2017, killing 66 and injuring 204 people. In 2018, several major series of wildfires broke out around the world, including in the United States (US), Canada, Australia, Greece, Portugal and the United kingdom (UK): California saw the largest wildfires on record; British Columbia saw the largest total burn-area during wildfire season on record; Sydney's bushfire season started two months early, in winter; more than 80 people were killed in Athens; forest fires wreaked havoc in the Algarve region; whilst a record-breaking series of wildfires burnt across the UK. In February 2019, New Zealand saw the worst wildfires in over 50 years, with 155 firefighters, 23 helicopters and three planes deployed to tackle the blaze. One helicopter crashed fighting the Nelson wildfire, with the pilot suffering moderate injuries. The Forestry Commission predicts that destructive wildfires will increase in frequency due to increased land-use pressure and climate change [2]. The development of more effective and safer means to fight wildfires is one of the world's most pressing challenges of our time [1]. Even though many organisations put a lot of effort in preventing wildfires by improving lands' resiliency, fires still occur and cannot be completely avoided. When a wildfire happens it is hardly extinguished and causes immense damage. Hence the urge to find a new strategy and technology to tackle wildfires by either containing them or, in the best case scenario, extinguishing them. The frontier of autonomous swarm robotics seems to provide good chances to succeed in this venture thanks to many strong points. Some advantages can be the following: the capability to work in hazardous areas without putting human lives in danger, the robustness of the system with respect to loss of a few agents, and no need of highly trained pilots to command directly each drone.

Besides, even though the presented research is focussed on firefighting swarms, it should not be misinterpreted as an attempt to suppress any wildfire on our planet and prevent any new ones from happening: if this would be even possible, it would have a detrimental impact on the worldwide ecosystem. In fact, wildfires act as oxygen-level controllers as supported by a fairly recent study on the relation between terrestrial ecosystems, fires and oxygen levels [116]; therefore, not all wildfires are supposed to be suppressed. Hence, the proposed technology should be used sparingly and thoughtfully evaluating the importance of the eventual ecosystem the fire is about to destroy and the impact on the local population's lives.

1.2 Aims and Objectives

The aim of the research is to contribute on various aspects of the eventual development of a firefighting technology that uses swarms of autonomous drones. The general aim is to develop control and cooperative decision-making algorithms for self-organising drones and study the dynamics of such system interacting with a mechanistic wildfire propagation model.
The main objectives of the presented research are the following:

1. To obtain a mechanistic parametric and realistic wildfire spread model, and to calibrate and validate it against realistic scenarios.

2. To design algorithms for a swarm of self-organising autonomous firefighting drones at different levels of autonomy – e.g. high-level behavioural algorithms; and medium- and low-level algorithms for collision-avoidance and flight-dynamics control.

3. To test the previously formulated algorithms for the swarm of drones over the simulated wildfire by performing a controlled indoor Swarm-in-the-Loop (SwiL) simulation.

Due to time constraints and major uncontrollable events (e.g. covid-19 pandemic) a great part of the work has been done towards SwiL simulations, but the last objectives has been accomplished by means of Software-in-the-Loop (SiL) simulations.

Moreover, some additional objectives formed throughout the research process. They has been considered to be essential for a reliable modelling of such system. They are:

4. the study of precision, accuracy and failure of utra-wide-band (UWB) based indoor positioning system and its improvement,

5. and the formulation of a stigmergy-based collision-avoidance algorithm for multi-agent systems.

1.3 Methodology

Given the multidisciplinary nature of this research, the work has been organised in three distinct phases, that would eventually fit together in order to create the final desired system.

1. After thoroughly assessing the already available wildfire propagation models, I realise there is no model that fits our requirements of being both mechanistic, parametric, realistic, and faster-than-real-time. Therefore, the first phase consists in developing a wildfire spread model that meets these criteria. Furthermore, it needs to be calibrated for a variety of vegetation fuels and validated against realistic scenarios.

2. This phase embeds the previously formulated wildfire propagation model in a simulation of the self-organising dynamics of the swarm. Each particle in the swarm is able to interact with the environment (e.g. by measuring temperature and dropping water). This is a first step towards SiL and SwiL simulations. This allows to quickly evaluate the effects of new high-level and low-level control and behavioural algorithms on the dynamics of the swarm, before passing to more complicated and computational expensive SiL simulations.

3. Once the simulation platform has been developed, new control/self-organisation algorithms and firefighting strategies can be tested.

1.4 Contributions

The contributions to this research amount of the following:

- **FireProM-F**, a faster-than-real-time physics-based wildfire propagation model;
- **self-organisation algorithms** for swarm of autonomous firefighting drones, and firefighting strategies;
- a **SiL platform** for testing algorithms (fire and swarm dynamics on Matlab);
- a stigmergic **collision-avoidance algorithm** based on the gradient of the cumulative signal strength field generated by the drones;
- and some work towards **Swarm-in-the-Loop platform**, by developing and improving tools in order to perform a hardware in the loop simulation of the developed self-organisation algorithms (this comprises the main effort in the study and improvement of the indoor positioning system);

More insight is provided in the following chapters.

1.5 Overview of the Report

The content of this Thesis is organised as follows:

1. *Chapter 2* collects the current theoretical knowledge and state-of-art technologies for each covered research topic in four sections, namely: wildfire propagation modelling, autonomous multi-agent robotics, collision-avoidance systems, and indoor positioning systems.

2. *Chapter 3* is focused on the wildfire propagation model and introduces FireProM-F. It contains the mathematical formulation of the model, the adopted numerical solution, and finally numerical experiments for calibration and testing.

3. *Chapter 4* presents the self-organisation algorithm for a swarm of autonomous firefighting drones, and related simulation results and discussion.

4. *Chapter 5* provides the formulation of a stigmergy-based collision-avoidance algorithm for the firefighting swarm depicted in the previous chapter. Some numerical experiments are also performed and discussed.

5. *Chapter 6* collects some work done towards the establishment of a Swarm-in-the-Loop platform, more specifically the topic is the indoor positioning system (IPS). This chapter provides a detailed study of the precision, accuracy, and failure of the IPS. Furthermore, a filtering process is proposed in order to improve the IPS performance. Specifically, a debiasing filter is formulated, calibrated, and validated against static and dynamic experiments.

6. *Chapter 7* contains a comprehensive conclusion of the presented research. The diagram 7.1 helps the reader in understanding how the various covered topics interlace and how they are working together in the grand fire suppression picture.

Chapter 2

Literature Survey

In this chapter an introductory research background is provided about the four main research topics of this thesis: wildfire propagation modelling, autonomous multi-agent robotics, collision-avoidance systems, and indoor positioning systems.

2.1 Wildfire Propagation Modelling

Modelling the propagation of wildfires is an incredibly challenging endeavour because of the complexity that arises from its multiphysics and multiscale nature. According to what phenomena, scales and modelling technique are considered, various classifications of wildfire mathematical models are possible (e.g. [3, 4, 5, 6, 1]).

Following the criteria set out in [1] and shown in Fig. 2.1 , a general classification is given by distinguishing between *data-driven*, *theoretical*, and *mechanistic surrogate* models. Data-driven models are built so as to fit available data, theoretical models are mechanistic and based on governing laws directly associated with the phenomena or system being modelled, whilst mechanistic surrogate models are based on a few leading assumptions that act as governing laws seemingly unrelated to the problem in question despite making predictions that appear sufficiently accurate. A model in any of these three classes may present continuous or discrete variables, responses or structures; and may be fully deterministic or include different degrees of stochasticity to model uncertainty [7].

Data-driven models which are purely data-fitting are called black-box models, a.k.a. phenomenological or empirical models. At the other end of the spectrum, theoretical models derived purely from first principles are referred to as white-box models. The combination of governing laws and empirical data results in grey-box models, which are semi-empirical. The latter are considered theoretical if mostly based on governing laws with some parameters calibrated using empirical data (light-grey-box), whereas they are considered data-driven if mostly driven by data but built upon a predefined structure derived from associated theories (dark-grey-box)[7].

Early models of wildfire propagation consisted of one-dimensional models of fire

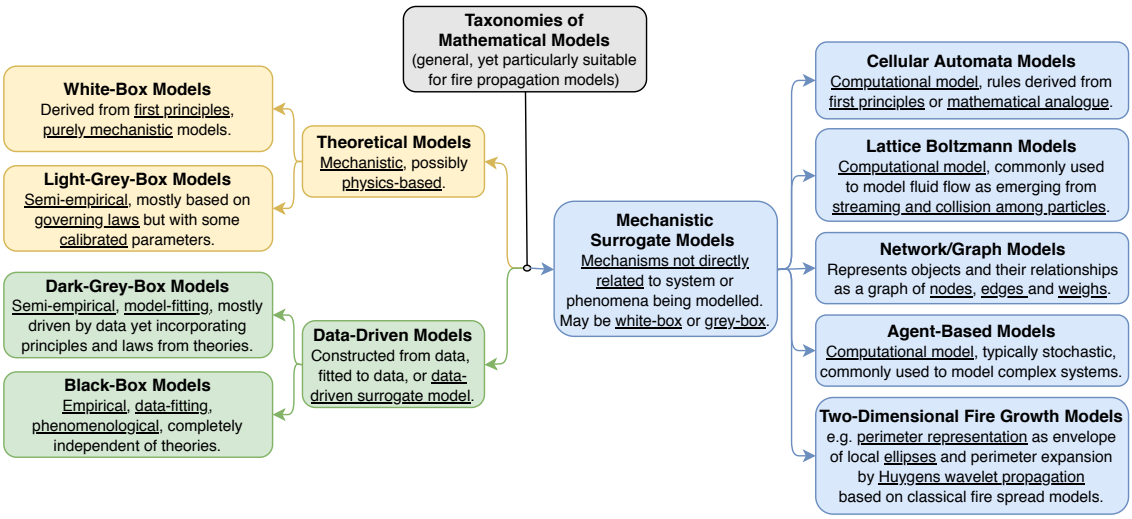


Figure 2.1: *General taxonomies of mathematical models suitable for the family of wildfire propagation models (modified version from [1]).*

behaviour based on the empirical determination of key characteristics such as the local rate of spread (RoS) at the headfire or the height and angle of the flames. Based on a heat balance model and empirical data, a prominent example is Rothermel's model [8] which predicts the RoS in the direction of the wind in an environment specified by fuel, weather and topography descriptors [3]. The model was incorporated into the point-based fire modelling system BehavePlus [9] and the Fire Area Simulator FARSITE [10]. For further reading on this type of models, refer to [3, 5, 11, 1].

Theoretical models are often physics-based, typically including balances of mass, momentum and energy with mathematical formulations expressed as systems of coupled partial differential equations (PDEs). Physics-based white-box models are often three-dimensional (3D), and attempt to describe all relevant phenomena with model parameters aimed to be mathematically derived. The scale of the model must be finer than the smaller representative scale of the modelled dynamics. In the context of fire modelling, they typically solve convection-reaction-diffusion-radiation equations. Whilst 3D multi-domain, multiscale and multiphase models of wildfire propagation can be found in the literature (e.g. [12, 13, 14, 15, 16]), they are computationally intensive. Physics-based light-grey-box models tend to be two-dimensional (2D), based on simple laws such as energy balance (e.g. [17, 18, 19]), and often require empirical parameters and experimental data to calibrate them for particular conditions. The research presented in this thesis focuses on this type of 2D models. Thus, Ferragut et al. [17] propose a system of two coupled PDEs, one for the energy balance and the other for the convection model. The energy balance is as in Eq. (2.1), where the energy (e) is an element of a multivalued operator that considers the latent heat of evaporation and the pyrolysis heat, whilst $S(T, X_{\text{fuel}})$ is a source term that accounts for combustion

and non-local radiation. Transport due to wind, conduction and convection are also considered. The two main variables are temperature (T) and fuel mass fraction (X_{fuel}) [7].

$$\frac{\partial e}{\partial t} + \overbrace{u \cdot \nabla e}^{\text{transport}} - \overbrace{\nabla(\kappa(T)\nabla T)}^{\text{conduction}} + \overbrace{\alpha T}^{\text{convection}} = \overbrace{S(T, X_{\text{fuel}})}^{\text{source}}, \tag{2.1}$$

Similarly, Margerit, Séro-Guillaume et al. [18, 20] propose a 2D anisotropic propagation model consisting of only one energy balance equation, as shown in Eq. (2.2):

$$(1-\Phi)\rho(c_{\text{s}} + H_{\text{u}}c_{\text{l}})\frac{\partial T}{\partial t} = \overbrace{\nabla \cdot (\kappa \nabla T)}^{\text{conduction}} + R_c + \overbrace{\chi(T_{\text{a}} - T)}^{\text{convection}} + \overbrace{(1-\Phi)\rho L_{\text{ev}}\delta_{T=T_{\text{ev}}}\frac{\partial H_{\text{u}}}{\partial t}}^{\text{evaporation}} + M_{\text{r}} \tag{2.2}$$

where R_c is the combustion heat source, Φ is the porosity of the medium, and c_{s}, c_{l} are the heat capacities of the solid and liquid parts, respectively. Interestingly, the local self-radiation heat flux is neglected while the non-local radiative term (M_{r}) is considered. M_{r} is the integral of all radiation coming from the flame above the simulation domain. The heat loss due to evaporation of a moisture fraction (H_{u}) is considered via the latent heat (L_{ev}) triggered by the evaporation temperature (T_{ev}), where δ is the Dirac distribution [7].

Mechanistic surrogate models are those designed disregarding the theories that underlie the system or phenomena being modelled, yet based on a few leading assumptions which may be somewhat physics-based [1]. Examples are Cellular Automata (CA) models, Lattice Boltzmann models (LBMs), Network models, and 2D Fire Growth models [7].

A CA model consists of a system of a high number of elements of simple geometry locally connected following a predefined scheme. Hence, two main constituents can be recognised: the *cellular space* and the *transition rule*. The former is a lattice of many identical finite-state machines whereas the transition rule evaluates the new state of the cell taking into account also adjacent cells identified by a *connection scheme*. CA models demonstrate a high level of efficiency and robustness when simulating complex physics. As far as wildfire propagation modelling is concerned, CA are mostly mechanistic surrogates. For instance, the CA model in [21] is not derived from fire dynamics theories but based on some main assumptions which may nonetheless have a physics origin. Other CA-based fire models can be found in [22, 23, 24, 25, 26, 27] [7].

The fundamental idea behind LBMs is a simplification of Boltzmann's representation of fluids as composed of a large but discrete number of particles. This model consists of a discretised representation of the Boltzmann transport equation that relates the particles' distribution to their velocities by means of a *collision operator*. An example is the simulation of combustion in a three-dimensional porous structure in [28].

Another discrete modelling approach, rarely adopted, is based on small-world networks. A square lattice is used to model short-range phenomena like radiation, convection, and diffusion, whilst fire-spotting processes are described by long-range con-

nections. Two examples are the modelling of initiation of spot-fires due to transport of firebrands [29] and fire-spread onboard naval vessels [30].

Two-Dimensional Fire Growth Models are sometimes referred to as vector-based models. They consist of predicting the fire front line in the form of an envelope curve making use of Huygens' principle [6]. At the core of these models is the notion of the RoS (called R in Rothermel's model), which is the local normal velocity of a fire front. Vector-based models can be further subdivided into two main branches, one using *level sets* and the other based on *markers* [7]. Level-set-based models (LSMs) such as [31, 32, 33] consist of tracking the interface of the fire front. The 2D curve (ϕ) described in Eq. (2.3) evolves in time following the rates of spread R_{u} and R_{v} in the x and y directions, respectively [7].

$$\frac{\mathrm{d}\phi}{\mathrm{d}t} = \frac{\partial\phi}{\partial t} + R_{\mathrm{u}}\frac{\partial\phi}{\partial x} + R_{\mathrm{v}}\frac{\partial\phi}{\partial y} = 0 \tag{2.3}$$

Differently, marker-based models (MMs) predict the movement of individual points (markers) to be connected to give shape to the fire line. Each marker behaves as a new ignition point for the next time step. The basic propagation geometry is an ellipse focused on the marker, which size, shape and orientation depend on the fuel type, wind intensity and local slope [20, 10]. One major drawback in the use of MMs is the need to change the number of markers throughout the simulation in order to maintain a certain level of precision [7]. Whilst the mathematical formulation of LSMs automatically handles the case of merging fire fronts, MMs need to carefully recognise which side delimited by the fire front is the burnt area. Widely known vector-based MMs are the Fire Area Simulator FARSITE [10], the Canadian wildland fire growth simulator Prometheus [34], and the Australian Bushfire Risk Management Tool Phoenix [35].

What modelling approach is best at predicting fire behaviour is still an open question, and the subject of much heated debate. For instance, whilst some researchers [**?** 5] adamantly support FARSITE given that it has been calibrated to reliably reproduce historical fires, others prefer approaches which are mostly physics-based and reliant on the rigorous description of the underlying fire dynamics. As an example of the latter, the *Fire Dynamics Simulator* (FDS) [36] is the result of 26 years of collaboration between several research centres around the world, FIRETEC [12] has been refined for 22 years, whilst the fire-spread models in [14, 18] have been developed during 16 years of rigorous work by the *Laboratoire d'Energétique et de Mécanique Théorique et Appliquée* (LEMTA). This research does not intend to answer this *burning* question. Instead, a physics-based approach is adopted as a requirement, since the model is intended to be used to underpin research on innovative firefighting technologies. Therefore, the physics-based interaction between the fire and suppressants will need to be modelled. Nonetheless, the proposed model also attempts to harness the ability of a commercial simulator to predict historical fires by calibrating a few parameters within realistic intervals [7].

2.2 Autonomous Multi-Agent Robotics

Unmanned aerial vehicle (UAV) technology has progressed rapidly for the past two decades, extending its capabilities and the kinds of problems it can help tackle. Modern UAVs – a.k.a. drones– can be equipped with a range of advanced cameras and sensors which enable them to operate in remote areas, dangerous environments, and even through solid smoke. Current applications include aerial photography and filming, information gathering for human decision-makers, provision of essential supplies, support for search and rescue operations, mapping of inaccessible locations, field surveying, and crop health monitoring. With regards to firefighting operations, drone technology has been applied to forest surveillance, building fire risk maps, forest fire detection and monitoring [37], post-fire recovery monitoring [38], bushfire hotspot detection [39], and support for disaster relief operations. Compared with their use in forest fire monitoring and detection, research and development on UAV-based fire suppression is still scarce [40]. Yet, given the hazardous nature of the activity, fighting fires using UAVs in place of humans is of special interest. Swarm intelligence (SI) is a route to artificial intelligence (AI) which stems from decentralised and self-organising behaviour observed in groups of social animals in nature. By way of collaboration, a form of collective intelligence emerges enabling them to accomplish tasks that are far beyond the aggregation of their individual capabilities. SI is the branch of AI that deals with the collective behaviour that emerges from decentralised self-organising systems, where individuals only interact locally with one another and with the environment. Swarm robotics (SR) is an approach to the self-coordination of large numbers of simple robots which emerged as the application of SI to multi-robot systems. It differs from other SI studies in that it emphasises the physical embodiment of individuals [41], and from distributed robotics in that it promotes scalability. There is a limited number of UAVs which can be remotely controlled and coordinated to operate simultaneously, therefore restricting the achievable fire suppression capabilities. There are also difficulties associated with centralised communication with ground control during wildfire events. Conversely, swarm robotic systems allow for a high number of self-coordinating agents with only local drone-to-drone communication and no central control. Furthermore, the use of swarms of decentralised collaborative and self-organising robots results in a robust and resilient system with collective decision-making able to cope with uncertainty, errors, local perturbations, and the failure or loss of a few units. While the use of drones to support firefighting operations is fast becoming common practice, the design of self-organising swarms of drones to directly engage in the suppression of fires remains unexplored [1].

Three main fields form the wealth of knowledge for the related research on self-organising swarm of firefighting drones presented in Chapter 4: swarm intelligence (SI), swarm robotics (SR), and particle swarms algorithms. Follows a brief introduction of such topics.

Swarm intelligence (SI) is a paradigm of AI which stems from decentralised and self-organising behaviour observed in groups of simple social animals in nature such as

ant, termite and bee colonies, fish schools and bird flocks. By way of collaboration, a form of collective intelligence emerges enabling these animals to accomplish tasks that are beyond the simple aggregation of their individual capabilities. That is, the whole is more than the sum of its parts. Thus, SI is the branch of AI that deals with the collective behaviour that emerges from decentralised self-organising systems. Self-Organisation occurs with no central control or sense of purpose, as individuals only interact locally with one another and with the environment inducing the emergence of global patterns. SI is the emergent collective intelligence of groups of simple agents [42]. As an AI discipline, it is concerned with the design of intelligent multi-agent systems. A SI system should satisfy the following conditions: (a) be composed of a number of simple agents or individuals, as some critical mass is required for self-organisation to occur; (b) be composed of agents or individuals who are similar to one another, typically identical or belonging to a few typologies; (c) have local interactions based on simple behavioural rules that exploit information exchanged locally among individuals in a direct manner or indirectly through the environment (stigmergy); (d) exhibit an emergent global behaviour which results from the interactions of individuals with one another and with the environment. This self-organised intelligent behaviour at the swarm level is not known at the individual level. These characteristics make SI systems scalable, parallel, robust and fault-tolerant. Scalability guarantees that the system can change size without redefining its behaviour. Since interactions are local, individual behaviour is marginally affected by changes in the swarm size. In turn, the population-based nature of these systems results in a parallel search that acquires information in a distributed manner. This makes them robust to local perturbations whilst the exchanges of information among agents enlarge the pool of knowledge decreasing uncertainty. Fault tolerance is due to their decentralised, self-organised and scalable nature as well as to the similarity among agents. If an agent is faulty or removed, the system does not cease to function [1].

Swarm robotics (SR) is an approach to the self-coordination of large numbers of simple, relatively inexpensive robots which emerged as the application of SI to multi-robot systems. Different from other SI studies, emphasis is on the physical embodiment of individuals [41]. SR systems differ from distributed robotic ones in that the former are scalable, which means that performance can be improved by increasing the size of the swarm without the need to redefine or reprogram the system. As a SI system, SR must abide by the conditions discussed previously. In addition, the agents must be autonomous robots operating in the physical world with the ability to sense and actuate in a real environment. These robots must possess individual capabilities that are limited relative to the task to be carried out at the system level. In other words, they must be unable to solve the problem absent collaboration. This simplicity carries the added benefit that the robots are inexpensive and less prone to failure.SR aims to study how a large number of simple robots can be designed so that a desired collective behaviour emerges from the local interactions among themselves and with the environment [43] [1].

Particle swarm optimisation (PSO) is one of the most successful SI algorithms, orig-

inally developed as a model of social behaviour inspired by earlier bird-flock simulations within the field of social psychology. In particular, Reynolds' boids [44] and Heppner and Grenander's artificial birds [45] strongly influenced early developments. It was also influenced by experiments and theories in social psychology such as Sherif's experiments, Bandura's no-trial learning, and Latané's Social Impact Theory [46]. Therefore, the method is closely related to other simulations of social processes and experimental studies in social psychology whilst also having strong roots in optimisation and AI as well as applications in SR. PSO is a global optimiser in the sense that it is able to escape poor suboptimal attractors thanks to a parallel search carried out by a swarm of cooperative particles. Its overall behaviour results from a combination of each particle's individual behaviour and the social behaviour that emerges from their interactions. The individual behaviour materialises as the trajectory of a particle pulled by an attractor. In most versions of the algorithm, this attractor results from some stochastic weighted average of an individual attractor (a particle's best experience) and a social attractor (best experience of neighbouring particles). The social behaviour is governed by how individually acquired information is propagated throughout the swarm. The individual and social behaviours are linked by the update of the social attractor in the trajectory equation. While the social behaviour is controlled by the neighbourhood structure (sociometry), the individual behaviour is controlled by the settings of the coefficients in the trajectory equation [1].

2.3 Collision-Avoidance Systems

Multi-agent systems such as drone swarms consist of a multitude of decision-making individuals that interact directly and indirectly within the environment in order to achieve one or more predefined goals. In such systems, one of the most critical issues to be dealt with is collision avoidance. In the related Chapter 5, I adhere to the classification of Collision-Avoidance Systems (CASs) suggested in [47] and [48], which consists of the following classes: 1) predefined collision avoidance; 2) protocol-based collision avoidance; 3) optimised escape trajectory approaches; 4) E-field / potential field methods; and 5) other CAS approaches. The first two categories are the original and most trivial approaches. Predefined collision avoidance follows a fixed set of unchangeable rules requiring no computation, whilst protocol-based collision avoidance relies on both a set of rules and continuous exchange of information among agents with regards to their states. The former results in a faster reaction time whereas the latter is safer. The most advanced CASs fall within the remaining three categories.

The optimised escape trajectory approach consists of solving an optimisation problem that combines the drone kinematic model with a set of physical and behavioural constraints. The following decentralised multi-agent algorithms fall within this group: 1) the Reciprocal Velocity Obstacles (RVO) and the derived Optimal Reciprocal Collision-Avoidance (ORCA) algorithms [49]; 2) the Cooperative Dynamic (CoDy) algorithm [50], which is able to solve dead-lock situations; 3) the Context-Aware Route Planning

(CARP) algorithm [51, 52], which is a graph routing algorithm aimed at finding the shortest trajectory and avoiding collisions; and 4) other optimisation-based algorithms such as [53] and [54].

Among the potential field (E-field) methods, the original ones suffer from the insidious problem of drones getting randomly trapped in local minima (e.g. [55]) though this issue has been dealt with (e.g. [56, 57]). The advantages are their low computational cost and short reaction time, whilst two major disadvantages still persist: 1) the requirement of high-level flight guidance, and 2) the potential occurence of hard-to-perform sharp discontinuities in the commanded manoeuvre.

In the miscellaneous category, other algorithms can be found. For example, a *sense-and-avoid* algorithm has been formulated for a structured multi-agent system in [58] and [48], where one leader is followed blindly by other drones. Such approach is claimed to have remarkable computational savings compared to other CASs for this specific application. Other researchers explore more exotic solutions to the problem, such as a two-stage reinforcement learning approach for multi-drones under imperfect sensing [59]. Their aim is to train a policy to plan a collision-free trajectory by leveraging local noisy observations. Others try to infer the state of the overall swarm through stigmergy, assuming this would be useful for collision avoidance purposes. For example, an approach based on anticipatory stigmergic collision avoidance (ASCA) under noise is proposed in [53], which consists of using pheromone information in a rather unusual fashion: instead of leaving a trail of pheromones over past positions, all drones in the swarm share information in the same indirect manner but about future intended positions instead. The drones will then optimise their trajectories in order to avoid locations with high concentrations of pheromone.

Numerous CASs have been proposed in the literature, the majority of which are suitable for Autonomous Ground Vehicles only. This is because CASs formulations typically rely on two main assumptions [53]: *stationary vehicles* (i.e. vehicles can stop suddenly and remain still indefinitely) and *perfect information* (accurate noiseless perception). This is certainly not the case for Unmanned Aerial Vehicles (UAVs) or Unmanned Underwater Vehicles, which might need to move with limited awareness of other vehicles' locations due to vision obstructions (e.g. smoke, cluttered environments, cloudy water) as well as high variance and bias of the positioning system being used. From here forth, drones will refer to UAVs only.

2.4 Indoor Positioning Systems

In robotics and autonomous systems, accurate and precise positioning constitute a crucial technology. A variety of positioning systems exist, which make use of different technologies, signal properties, and positioning algorithms [60]. Technologies include inertial navigation systems (INS) [61, 62], sound waves [63, 64], infrared [65], visible light [66], and radio frequency – including Ultra-WideBand (UWB) [67], bluetooth [68], Wireless Local Area Network (WLAN) [69, 70], and Wireless Underground Sensor Net-

work [71]. Signal properties used for positioning include Angle of Arrival (AoA) [65], Time of Arrival (ToA) [71], Time Difference of Arrival (TDoA) [72], and Received Signal Strength Indication [60]. Positioning algorithms include triangulation, trilateration [73, 74], proximity, and Two-Way Ranging algorithms [75]. Technologies which make use of a prediction model (e.g. INS), and are highly sensitive to external disturbances, typically rely on sensor fusion approaches that use Kalman filter and neural networks [76, 77].

Global Navigation Satellite Systems (GNSSs) are suitable for efficient outdoor long-range positioning. Whilst the most common technology is the Global Positioning System (GPS), the European Galileo started providing services in 2016 with a constellation of 26 satellites [78]. GNSSs allow an electronic receiver to determine its position by trilateration using radio signals travel times (ToA) from at least four satellites [79]. However, since these signals cannot penetrate walls or objects, using this technology for Indoor Positioning Systems (IPSs) is infeasible. Conversely, UWB technology is well-suited for IPSs given that they present high data transfer rates over short distances and at low power densities, whist also having the ability to penetrate obstacles [67]. In addition to that, UWB-based IPSs constitute one of the most accurate and precise positioning technologies at present. All things considered, they are arguably the best choice amongst current technology [80, 67]. However, they are susceptible to interferences, which may be caused by metallic materials or by systems working on similar frequencies. Recently developed hybrid solutions make use of UWB technology and LIDAR-based range finders for near-wall localisation in GNSS-denied Environments [81].

Traditionally, the precision of positioning systems is studied by performing a Cramér–Rao Lower Bound (CRLB) analysis [82] from the signal perspective, and then applying coefficients such as Geometric Dilution of Precision (GDoP) [83] to capture the geometrical features – e.g. to identify where there is a sudden drop of the IPS's performance. CRLB analysis is widely accepted for systems where the node to be localised is far away from the anchors –e.g. for GNSSs– but is generally deemed unsuitable for IPSs [84, 85], in which anchors and nodes are located within a confined space.

Chapter 3

FireProM-F: wildfire propagation model

This chapter presents the mathematical formulation, numerical solution, calibration and testing of a physics-based model of wildfire propagation aimed at faster-than-real-time simulations. Despite a number of simplifying assumptions, the model is comprehensive enough to capture the major phenomena that govern the behaviour of a real fire –namely, the pyrolysation of wood; the combustion of a mono-phase medium composed of premixed gas of fuel and air; and the heat transferred by conduction, convection, radiation, mass diffusion and transport due to atmospheric wind. The model consists of two coupled partial differential equations, one representing the mass formation of each chemical species involved in the combustion and the other ensuring the balance of enthalpy. Dimensionality reduction is sought by modelling these three-dimensional phenomena in a two-dimensional space, which has been achieved by means of heat-sources and heat-sinks to account for the third dimension in the energy balance equation. Once calibrated with a widely used non-physics-based commercial wildfire simulator, the proposed **Fire Pro**pagation **M**odel for **F**ast simulations (FireProM-F) is tested, returning similar predictions in terms of the size and shape of the burnt area, although similarity deteriorates for windy conditions. It has the added benefit of being both physics-based and computationally inexpensive, so that its interaction with fire suppressants may also be modelled in the future and simulated in real time [7].

The remainder of this chapter is organised as follows: Section 3.1 presents the mathematical formulation of the proposed model, including detailed descriptions of its components; Section 3.2 deals with the numerical method adopted to solve the formulated initial boundary value problem; Section 3.3 is concerned with numerical experiments, where Section 3.3.1 presents the calibration process, Section 3.3.2 presents case studies to test the calibrated model, and Sections 3.3.3-3.3.4 show applications of the proposed model on a realistic scenario; finally, conclusions and recommendations for future work are offered in section 3.4 followed by a brief summary of the chapter's topic, scope, method and findings in section 3.5.

3.1 Proposed model

In line with [17] and [18], the vegetation stratum is modelled as a virtual premixed layer of pyrolysis gases and air. The considered reference irreversible chemical reaction is the combustion of methane in air:

$$\theta_1 \mathrm{CH_4} + \theta_2 \mathrm{O_2} \rightarrow \theta_3 \mathrm{CO_2} + \theta_4 \mathrm{H_2O} \tag{3.1}$$

The reactants in Eq. (3.1) are the gaseous fuel and oxygen, mixed in air. The latter is considered to be composed of oxygen, carbon dioxide, water vapour, and nitrogen. Even though five chemical species are considered in this model, the nitrogen mass can be assumed constant, therefore leading to only four mass balance equations. The fire spread model can be represented by a system of five coupled PDEs: Eq. (3.2) for the enthalpy balance, where F_T is the summation of all heat sources and sinks; and Eq. (3.3) for the consumption or formation of each of the four chemical species – i.e. fuel, O_2, CO_2 and H_2O [7].

$$\frac{\partial}{\partial t}\left(c_{\mathrm{p}} T\right) = F_T\left(T, X_i\right) \tag{3.2}$$

$$\frac{\partial X_i}{\partial t} = F_{X_i}\left(T, X_i\right), \quad i = 1, \ldots, 4 \tag{3.3}$$

In Eq. (3.3), F_{X_i} is the molar fraction formation rate. The system of Eqs. (3.2)–(3.3) can be expanded as shown in the system of Eqs. (3.4)–(3.5) [7].

$$\rho \frac{\partial}{\partial t}\left(c_{\mathrm{p}} T\right) = R_{\mathrm{c}} + Q_{\mathrm{w}} - \nabla \cdot \left(\mathbf{q}_{\mathrm{c}} + \mathbf{q}_{\mathrm{d}} + \mathbf{q}_{\mathrm{r}}\right) + Q_{\mathrm{conv}} + Q_{\mathrm{rz}} \tag{3.4}$$

$$\frac{\partial X_i}{\partial t} = -\frac{\theta_i}{\theta_1}\frac{M_i}{M_1} r, \tag{3.5}$$

where the following terms can be identified:

- ▷ R_{c}: *Combustion* energy source - i.e. energy released by consumption of the fuel (see section 3.1.1);
- ▷ Q_{w}: Transport term due to *wind* - i.e. horizontal convective preheating (see section 3.1.2);
- ▷ $\mathbf{q}_{\mathrm{c}}$: *Conductive heat flux* (small compared to other terms but not negligible, see section 3.1.3);
- ▷ $\mathbf{q}_{\mathrm{d}}$: *Interdiffusional enthalpy flux* - i.e. heat transported by species diffusion (see section 3.1.4);
- ▷ Q_{conv}: *Vertical convection* - i.e. buoyancy flux, convective heat sink outside the 2D simulation domain (see section 3.1.5);
- ▷ $\mathbf{q}_{\mathrm{r}}$: *2D radiation* heat flux - i.e. horizontal radiative preheating (see section 3.1.6);

▷ Q_{rz}: *Vertical radiation* - i.e. radiative heat loss outside the 2D simulation domain (see section 3.1.7).

Notice that the radiation heat flux is split into two contributions: the auto-radiation in the 2D domain, and the emitted radiation outward from the domain in the vertical direction. Thus, the proposed **Fire Pro**pagation **M**odel for **F**ast simulations (FireProM-F) is governed by Eqs. (3.4)-(3.5) [7].

3.1.1 Combustion energy

The combustion energy in Eq. (3.6) depends on the combustion enthalpy (h_c) and on the reaction rate (r), both of which depend on the temperature (T), the mole fraction of fuel (X_1), and the mole fraction of oxidiser (X_2) [7].

$$R_{\mathrm{c}} = -\rho c_{\mathrm{h}} h_{\mathrm{c}} \frac{M}{M_1} r \tag{3.6}$$

The role of the enthalpy proportionality coefficient (c_{h}) is explained in Section 3.3.1. The combustion enthalpy in Eq. (3.7) consists of the summation of all formation enthalpies H_i at the specific local temperature T. The reference empirical values $H_{i,\mathrm{ref}}$ and T_{ref} can be found in [86] [7].

$$h_{\mathrm{c}} = \frac{H_{\mathrm{c}}(T)}{\bar{M}} = -\frac{1}{\bar{M}} \sum_{i=1}^{5} \theta_i H_i(T) = \frac{1}{\bar{M}} \sum_{i=1}^{5} \theta_i \left(H_{i,\mathrm{ref}} + M_i c_{\mathrm{p}i} \left(T_{\mathrm{ref}} - T\right)\right) \tag{3.7}$$

The combustion rate in Eq. (3.8) represents the rate of fuel consumption and follows the law of mass action –i.e. the exponential Arrhenius law [7]. The pre-exponential coefficient (A_r) and the activation temperature (T_{a}) are empirical parameters that depend on the fuel structural and chemical properties [87].

$$r = -\delta^{+}_{(T,X_{1,2})} A_r T X_1^{0.5} X_2 \exp\left(-\frac{T_{\mathrm{a}}}{T}\right) \tag{3.8}$$

In Eq. (3.8), $\delta^{+}_{(T,X_{1,2})}$ is the Kronecker delta as defined in Eq. (3.9), which represents a simple extinction model: if the temperature is lower than the ignition temperature (T_{ig}) or if either the fuel mass fraction or the oxidant mass fraction is lower than the corresponding flame extinction value (X_{1e}, X_{2e}), the combustion is deactivated –i.e. the combustion rate is null. It is assumed here that the ignition temperature is equal to the activation temperature [7].

$$\delta^{+}_{(T,X_{1,2})} = \begin{cases} 1 & \text{if} \quad T > T_{\mathrm{ig}} \wedge X_{1,2} > X_{1,2e} \\ 0 & \text{otherwise} \end{cases} \tag{3.9}$$

3.1.2 Transport due to wind

The transport term in Eq. (3.10) models windy conditions, where $\mathbf{u} = (u_1, u_2)$ is the atmospheric wind velocity. The meaning of the wind reduction coefficient (c_{w}) is explained in Section 3.3.1 [7].

$$Q_{\mathrm{w}} = -\rho c_{\mathrm{w}} \mathbf{u} \cdot \nabla \left(c_{\mathrm{p}} T \right) \tag{3.10}$$

3.1.3 Conductive heat flux

The conductive heat flux is modelled as shown in Eq. (3.11), where the thermal conductivity (κ) is assumed to be constant.

$$\mathbf{q}_{\mathrm{c}} = -\kappa \nabla T \tag{3.11}$$

3.1.4 Interdiffusional enthalpy flux

The adopted interdiffusional enthalpy flux is as in Eq. (3.13), based on the Fickian approximation of the diffusive mass flux ($\mathbf{J}_i$) shown in Eq. (3.12) [88]. The diffusivity (D_i) is assumed to be the same for every chemical species and it has been expressed in Eq. (3.13) as a function of κ to reduce the number of parameters [89].

$$\mathbf{J}_i \approx -\rho \left(D_i \nabla Y_i - Y_i \sum_j D_j \nabla Y_j \right) \tag{3.12}$$

$$\mathbf{q}_{\mathrm{d}} = \sum_i h_i J_i \approx \sum_i h_i \rho D_i \nabla Y_i \approx \frac{\kappa T}{c_{\mathrm{p}} \bar{M}} \sum_i c_{\mathrm{p}i} \nabla \left(X_i M_i \right) \tag{3.13}$$

3.1.5 Vertical convection

The heat loss due to vertical convection is governed by the simple formulation in Eq. (3.14), where the atmospheric domain above the vegetation is at ambient temperature (T_{amb}) [7].

$$Q_{\mathrm{conv}} = \chi \left(T_{\mathrm{amb}} - T \right) \tag{3.14}$$

As suggested by Séro-Guillaume et al. [20], this simple linear model should be improved. Nonetheless, one could assume the disregarded nonlinearities in the vertical convection dynamics to be implicitly embedded in the radiation heat loss in Eq. (3.22). Though this is not a rigorous theoretical observation, it could serve well for the purpose of the proposed wildfire model [7].

3.1.6 2D radiation heat flux

Radiation is a volumetric phenomenon proportional to the 4^{th} power of the temperature of the source [90]. The hemispherical radiation power ($\mathbf{q}$) emitted by a finite radiative volume is shown in Eq. (3.15) in terms of the Boltzmann constant (σ), the emittance (ε), and the radiation aspect (ϕ) [7].

$$\mathbf{q} = \sigma\varepsilon\phi T^4 \tag{3.15}$$

Let us consider two sources separated by the absorption length (δx), which are at different temperatures as in Eq. (3.16). Then, the radiative heat fluxes, $\mathbf{q}_{\text{A}}$ and $\mathbf{q}_{\text{B}}$, are as in Eq. (3.17) [7].

$$\begin{cases} T_{\text{A}} = T \\ T_{\text{B}} = T + \frac{\partial T}{\partial x}\delta x \end{cases} \tag{3.16}$$

$$\begin{cases} \mathbf{q}_{\text{A}} = \sigma\varepsilon\phi T^4 \\ \mathbf{q}_{\text{B}} = \sigma\varepsilon\phi\left(T + \frac{\partial T}{\partial x}\delta x\right)^4 \end{cases} \tag{3.17}$$

The 4^{th} power of the temperature at point B in Eq. (3.17) may be simplified by neglecting some small differentials, as shown in Eq. (3.18).

$$\left(T + \frac{\partial T}{\partial x}\delta x\right)^4 \approx T^4 + 4T^3\frac{\partial T}{\partial x}\delta x \tag{3.18}$$

A formulation of the net heat flux ($\mathbf{q}_{\text{r}}$) in the x direction ($q_{\text{r}x}$) can be obtained as in Eq. (3.19) [7].

$$q_{\text{r}x} = \mathbf{q}_{\text{A}} - \mathbf{q}_{\text{B}} = -4\sigma\varepsilon T^3\frac{\partial T}{\partial x}\delta x \tag{3.19}$$

In order to consider the energy variation induced by radiation on a finite control volume of dimension $dx \times dy \times dz$, one could evaluate the energy change due to radiation heat flux through the faces perpendicular to the domain plane:

$$Q_{\text{r}xy} = -\nabla\cdot\mathbf{q}_{\text{r}} = 4\sigma\varepsilon\delta_x\left[\frac{\partial}{\partial x}\left(T^3\frac{\partial T}{\partial x}\right) + \frac{\partial}{\partial y}\left(T^3\frac{\partial T}{\partial y}\right)\right] = 4\sigma\varepsilon\delta_x\nabla\cdot\left(T^3\nabla T\right) \tag{3.20}$$

Hence, the local 2D self-radiation heat flux expression results as shown in Eq. (3.21) [7].

$$\mathbf{q}_{\text{r}} \approx -4\sigma\varepsilon\delta_x T^3\nabla T \tag{3.21}$$

The absorption length δ_x (or optical thickness) is assumed to be isotropic, and it has to be equal to or smaller than the smallest cell size for the numerical solution to be valid [7].

3.1.7 Vertical radiation

The vertical radiation heat loss shown in Eq. (3.22) [90] consists of the emission along the z direction –i.e. perpendicular to the simulated xy domain [7].

$$Q_{\mathrm{rz}} = \sigma\varepsilon\delta_z^{-1}\left(T_{\mathrm{amb}}^4 - T^4\right) \tag{3.22}$$

3.1.8 Closure equations

Finally, the system is closed with the following equations:

$$\bar{M} = \sum_{i=1}^{5} X_i M_i \tag{3.23}$$

$$c_{\mathrm{p}} = \sum_{i=1}^{5} X_i \frac{M_i}{\bar{M}} c_{\mathrm{p}i} \tag{3.24}$$

The *molar mass* ($\bar{M}$) of the mixture in Eq. (3.23) is a linear combination of the molar masses of each chemical species (M_i) multiplied by the respective mass fraction (X_i). Similarly, the total *heat capacity coefficient* at constant pressure (c_{p}) of the mixture in Eq. (3.24) is obtained by weighted summation of the partial heat capacities ($c_{\mathrm{p}i}$) of each chemical species [7].

3.2 Numerical solution

The numerical scheme used to solve the system of PDEs in Eqs. (3.4) and (3.5) is given by the 2$^{\mathrm{nd}}$ order centred finite difference method for the spatial integration and the 4$^{\mathrm{th}}$ order Runge-Kutta method (RK4) for the time integration. For example, time integration for Eq. (3.2) is as shown in Eq. (3.25), where $\tilde{T} = c_{\mathrm{p}}T$:

$$\begin{cases} \tilde{T}^{k+1} &= \tilde{T}^k + F_T \Delta t \\ F_T &= (1/6)\left(F_{T,1} + 2F_{T,2} + 2F_{T,3} + F_{T,4}\right) \\ F_{T,1} &= F_T\left(\tilde{T}^k, X_i^k\right) \\ F_{T,2} &= F_T\left(\tilde{T}^k + F_{T,1}\frac{\Delta t}{2}, X_i^{k+\frac{1}{2}}\right) \\ F_{T,3} &= F_T\left(\tilde{T}^k + F_{T,2}\frac{\Delta t}{2}, X_i^{k+\frac{1}{2}}\right) \\ F_{T,4} &= F_T\left(\tilde{T}^k + F_{T,3}\Delta t, X_i^{k+1}\right) \end{cases} \tag{3.25}$$

This is an accurate and efficient (explicit) method, although it is still conditionally stable. For instance, the maximum time-step to ensure stability for a grid-size of 0.25 m^2 is 5 s.

All numerical derivatives in space are 2^{nd} order accurate except for when applying the flux limiter operator in the transport term, which gradually reduces accuracy to 1^{st} order by using an upwind scheme [7].

As far as Boundary Conditions (BCs) are concerned, the main idea is to keep the simulation domain as unconstrained as possible, allowing free heat fluxes through the boundaries. Therefore, I applied Dirichlet BCs on both the temperature (ambient) and the chemicals mass fractions. The fuel mass fraction right at the boundary and some neighbouring cells is set to zero, creating a safe band where the fire cannot spread. In order to avoid eventual artificial effects on the solution due to boundary assignment, the flaming area is kept far from the domain edges [7].

Regarding the initial conditions, the initial temperature field is set to the atmospheric temperature at the considered altitude. The ignition point is represented by a Gaussian distribution, which peak temperature is 300 K higher than the pyrolysis temperature to avoid numerical effects related to discontinuities or excessively steep gradients. The flammable gas mixture is lean, setting the initial fuel molar fraction to 10% with the remaining 90% air, which results in an almost unlimited availability of oxygen for the combustion to take place [7].

A flux limiter is applied to the transport term Q_{w} in Eq. (3.10) in order to constrain the heat fluxes with the wind direction. The flux limiter in Eq. (3.26) is formulated in order to have a gradual switch from centred differences at no-wind condition towards upwind differences scheme for very strong winds [7].

$$Q_{\text{w}} = -\rho c_{\text{w}} \overline{\mathbf{u} \cdot \nabla \tilde{T}}^{FL} \tag{3.26}$$

Moreover, negative transport heat fluxes due to wind are reduced in order to enforce the permanence of the flame in the vegetation stratum, which is not a volatile gas mixture.

In cases with multiple types of fuels present in the same domain, a problem arises in choosing the parameters for the spatial derivatives which affect the value of the heat fluxes at the interface between two different fuel types. Table 3.1 shows how to choose between the local, the minimum, and the maximum values. While it would have made sense to choose local values for δ_z^* and χ, better predictions were observed with the choices shown in Table 3.1 [7].

parameter	$\boldsymbol{c}_{\text{h}}$	$\boldsymbol{A}_{\text{r}}$	$\boldsymbol{\kappa}$	$\boldsymbol{\delta}_{\boldsymbol{x}}^*$	$\boldsymbol{\delta}_{\boldsymbol{z}}^*$	$\boldsymbol{\chi}$	$\mathbf{T}_{\text{ig}}$	$\boldsymbol{c}_{\text{w}}$
value	local	local	min	min	min	max	local	local

Table 3.1: *Choice of parameters' values at the interface between two different fuels as either local, minimum or maximum [7].*

The model's predictions have been compared against the simulated dynamics obtained by other models and experimental results from the literature. For instance, Fig. 3.1 shows the radial temperature profile of a circular flame growing radially from a single ignition point in a grass bed in the centre of the domain. A qualitatively and

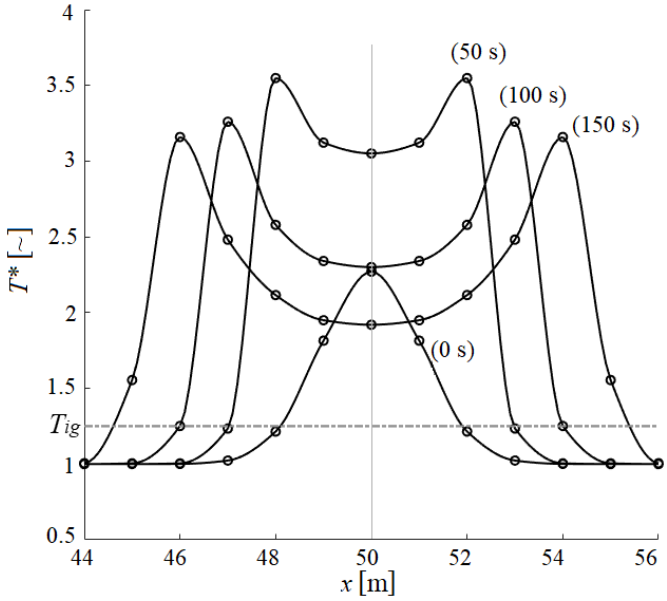


Figure 3.1: *Temperature profile along a radial direction across an ignition point, where $T^* = T/T_{amb}$ [7].*

quantitatively similar result was reported by Ferragut et al. [17], whose model was previously introduced in Eq. (2.1) [7].

Furthermore, in order to observe the combustion dynamics, I studied the evolution in time of temperature, fuel mass fraction, products mass fraction (e.g. CO_2), and fuel mass loss rate [7]. This is shown in Fig. 3.2, where some non-dimensional values are defined as follows:

i) Temperature:
$(T - T_{\text{amb}})^\star = (T - T_{\text{amb}})/T_{\text{amb}}$

ii) Fuel mole fraction:
$X^\star_{\text{CH}_4} = X_{\text{CH}_4}/X^{t=0}_{\text{CH}_4}$

iii) Product mole fraction:
$X^\star_{\text{CO}_2} = X_{\text{CO}_2}/X^{t\to\infty}_{\text{CO}_2}$

In Fig. 3.2, it is easy to recognise the heating phase ($t_1 < t < t_2$), combustion phase ($t_2 < t < t_4$), ignition of a neighbouring cell ($t_3 < t < t_4$), and suppression/cooling phase ($t > t_4$). The local combustion dynamics obtained with the proposed model is also qualitatively and quantitatively similar to those reported from other models and experiments such as those in [91, 92, 93].

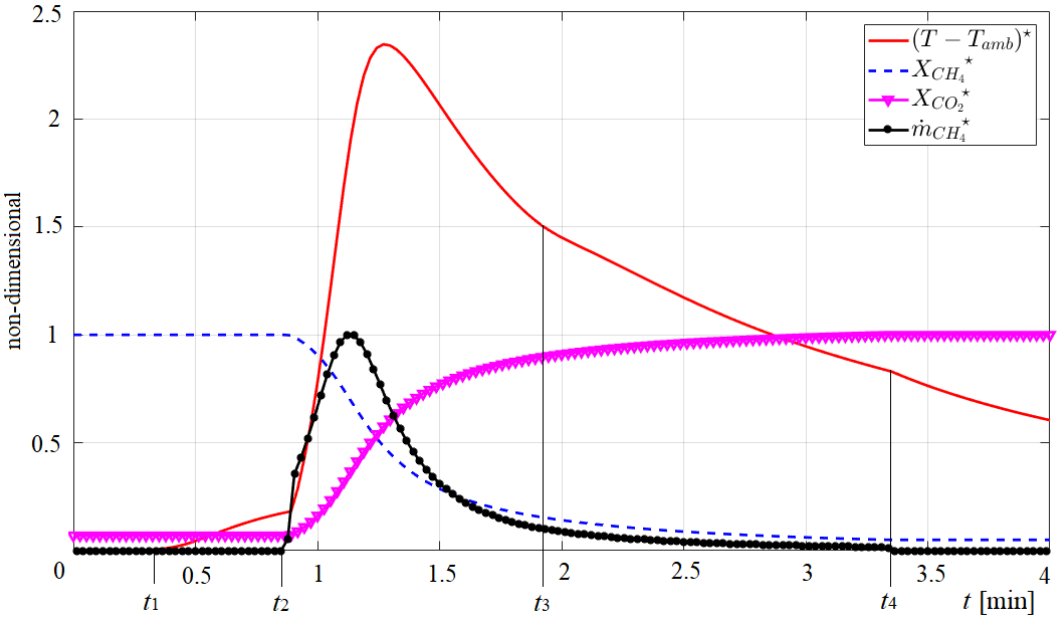


Figure 3.2: *Ignition dynamics of a cell showing the average temperature normalised by the ambient temperature, the molar fraction of fuel by its initial value, the molar fraction of the carbon dioxide by its final value and the mass loss rate of fuel normalised by its maximum value. The specific times t_i, with i from 1 to 4, point out a crucial event of the simulated environment [7].*

3.3 Numerical experiments

Absent access to detailed wildfires data-bases, the proposed fire propagation model is calibrated against a commercial simulator for single-species and uniform wind scenarios. The commercial simulator of choice is FARSITE [10], as it has been validated against historical fires and is equipped with an extensive collection of fuel models. Given the similarity between FireProM-F and the model proposed by Séro-Guillaume et al. [20], I was tempted by using their model for calibration and testing purposes. However, three conditions made this choice prohibitive (compared to the obvious choice for FARSITE): the lack of validation with historic wildfires, the lack of different fuel models, and the low availability of the model. The calibrated model is then tested for different wind speeds and on different mixed-fuels scenarios. Finally, a realistic scenario is designed for further testing, which also includes atmospheric wind and barriers to the propagation of the wildfire. All propagations predicted by this model in the numerical experiments are compared against those returned by the commercial simulator [7].

3.3.1 Model calibration

The calibration methodology adopted is described in this section, where the reference values are obtained by carrying out simulations of fire propagation especially designed for this purpose using the popular Fire Area Simulator FARSITE.

The methodology consists of starting a fire from a single ignition point within a uniform fuel bed in the centre of a square domain of 100 m × 100 m in the absence of wind, atmospheric conditions at sea level, and 20% of fuel moisture. Evidently, the fire is to propagate radially displaying a circular propagation front. The ignition point is represented by a Gaussian distribution, as dicussed in Section 3.2. Whilst one of the most relevant features of wildfires is the rate of spread (RoS), an indirect way to take this into account while also considering a 2D propagation is by using as a reference the burnt area (A_b) once the wildfire has propagated for a predefined length of time. As a trade-off between computational time and reliability, the fire is allowed to propagate for 10 min before extracting this reference value. Note that A_b is defined as the area that, at any time during the simulation, has had some of its fuel burning [7].

Thus, the objective is to minimise the error between the burnt area predicted by the presented model (A_b) and the one predicted by FARSITE (A_ref). The optimisation problem is formulated in Eq. (3.27) where $\mathbf{x}$ is a vector gathering seven calibration variables, namely: the combustion heat proportionality coefficient (c_h), the pre-exponential Arrhenius coefficient (A_r), the thermal conductivity (κ), the modified optical thickness on the simulation plane ($\delta_x^\star$), the modified optical thickness in the vertical direction ($\delta_z^\star$), the turbulent convection coefficient (χ), and the effective ignition temperature (T_ig). Calibration variables and their feasible ranges are shown in Table 3.2 [7].

$$\begin{aligned} \underset{\mathbf{x}}{\text{Minimise}} \quad & (A_\mathrm{b} - A_\mathrm{ref})^2 \\ \text{Subject to} \quad & T_\mathrm{max} = T_\mathrm{ref} \\ & \mathbf{x} \in [\mathbf{x}_\mathrm{min}, \mathbf{x}_\mathrm{max}] \end{aligned} \tag{3.27}$$

T_ref = 1,200 K is the temperature of wood combustion in the absence of wind and crown fire [91]. Since higher temperature peaks are possible under different circumstances, this is imposed as a soft constraint by setting a small tolerance. The optimisation problem is solved using SQP, with variables calibrated for three different types of fuel. Adopting the same standard classification as FARSITE [94], these are: FM1 (short grass), FM2 (timber), and FM6 (dormant brush and hardwood slash). Resulting values of the calibrated variables are provided in Table 3.3.

$\boldsymbol{c}_\mathrm{h}$ [∼]	$\boldsymbol{A}_\mathrm{r}$ [$\mathrm{K^{-1}s^{-1}}$]	$\boldsymbol{\kappa}$ [$\mathrm{Wm^{-1}K^{-1}}$]	$\boldsymbol{\delta_x^*}$ [m]	$\boldsymbol{\delta_z^*}$ [m]	$\boldsymbol{\chi}$ [$\mathrm{Wm^{-3}K^{-1}}$]	$\boldsymbol{T}_\mathrm{ig}$ [K]
[0.7,1.2]	[1,10]e-5	[1,10]e-1	[1,100]e-2	[1,100]e-1	[1,10]e-2	[$\mathrm{T_{amb}^+$,800]

Table 3.2: *Feasible range for calibration variables of the proposed fire propagation model [7].*

All calibration variables in the model have physical meaning, as described below:

FM	$\boldsymbol{c}_{\mathrm{h}}$ [$\sim$]	$\boldsymbol{A}_{\mathrm{r}}$ [$\mathrm{K^{-1}s^{-1}}$]	$\boldsymbol{\kappa}$ [$\mathrm{Wm^{-1}K^{-1}}$]	$\boldsymbol{\delta}^*_{\boldsymbol{x}}$ [m]	$\boldsymbol{\delta}^*_{\boldsymbol{z}}$ [m]	$\boldsymbol{\chi}$ [$\mathrm{Wm^{-3}K^{-1}}$]	$\boldsymbol{T}_{\mathrm{ig}}$ [K]
1	0.995	5.757e-5	0.978	3.43e-2	1.67	4.92e-2	374.5
2	0.988	4.324e-5	0.499	3.65e-2	1.49	5.16e-2	458.3
6	0.803	4.154e-5	0.255	2.98e-2	1.91	2.16e-2	431.6

Table 3.3: *Variables calibrated with respect to the burnt area (A_b) predicted by FARSITE 10 min after ignition [7].*

- $\boldsymbol{c}_{\mathrm{h}}$ is a proportionality coefficient that takes into account the possibility of burning fuels that have different specific combustion energy with respect to gaseous methane. Considering the reaction in Eq. (3.1), the energy output due to combustion is predefined by the reference enthalpy of each chemical species involved between products and reactants, as shown in Eq. (3.7). In order to consider a wider spectrum of fuels that might have different reference enthalpy than that of methane, the combustion enthalpy of the gaseous mixture of methane and air is multiplied by $\boldsymbol{c}_{\mathrm{h}}$. Refer to Eq. (3.6) [7].

- $\boldsymbol{A}_{\mathrm{r}}$ is the pre-exponential Arrhenius coefficient, which affects the reaction rate. Similar to c_{h}, the reference chemical reaction is the combustion of methane that has a precise reaction rate. Therefore, in order to consider different combustion rates, A_{r} is a critical parameter to be calibrated. This will sensibly change the combustion power output [7].

- $\boldsymbol{\kappa}$ is the thermal conductivity, which controls the heat transferred by conduction.

- $\boldsymbol{\delta}^{\star}_{\boldsymbol{x}}$ is the modified optical thickness in the simulation plane as defined in Eq. (3.28), where δ_x is the optical thickness in x whose value depends on factors such as porosity and smoke presence, and must be smaller than the cell-size as a model assumption (see Section 3.1.6) [7].

- $\boldsymbol{\delta}^{\star}_{\boldsymbol{z}}$ is the modified optical thickness in the vertical direction as defined in Eq. (3.29), where δ_z is the optical thickness in z whose value depends on the height of the vegetation stratum and on smoke presence [7].

- $\boldsymbol{\chi}$ is the turbulent convection coefficient, which strongly depends on the combustion power strictly affecting the intensity of the buoyancy fluxes.

- $\boldsymbol{T}_{\mathrm{ig}}$ is the effective ignition temperature, which can be far lower than the actual ignition temperature for the reference fuel (CH_4) since I am considering the average value in the cell and modelling wood combustion. The activation temperature T_{a} for the combustion to take place is set equal to T_{ig} [7].

$$\delta^{\star}_{x} = \varepsilon \delta_x \tag{3.28}$$

$$\delta_z^\star = \delta_z/\varepsilon \tag{3.29}$$

Note that $\boldsymbol{\varepsilon}$ in Eqs. (3.28)-(3.29) is the radiation emissivity, which depends on both porosity of the vegetation medium and smoke presence.

The calibration described thus far applies to no-wind conditions. Hence, a new calibration variable is considered to account for the wind effect, namely the wind reduction coefficient ($\boldsymbol{c}_{\mathrm{w}}$). The latter is always smaller than one, and reduces the atmospheric asymptotic wind velocity to an effective wind velocity due to drag at the interface and inside the vegetation layer. This parameter should depend on porosity, average height, and crown coverage [7].

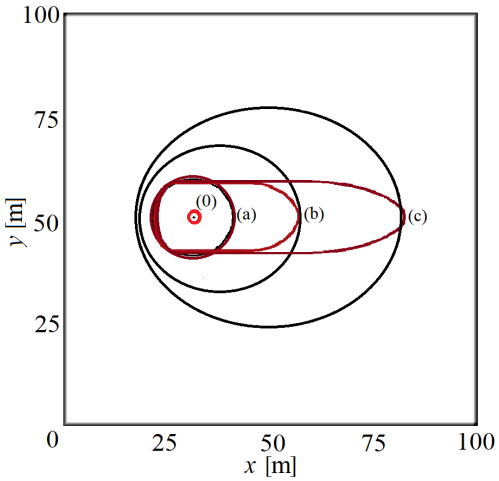


Figure 3.3: *Fire perimeter predictions by FireProM-F (red) and by FARSITE (black) 10 min after ignition with (a) 0 m/s, (b) 5 m/s and (c) 10 m/s wind speeds blowing from left to right. The ignition point is marked as (0), and a slow-burning fuel is used [7].*

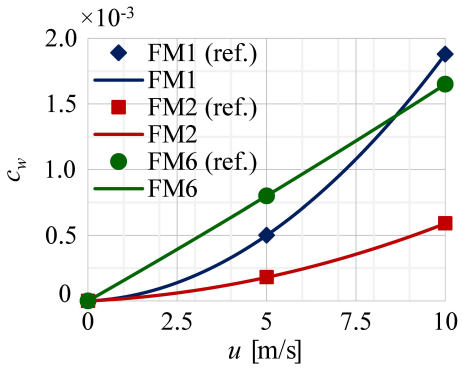


Figure 3.4: *Wind reduction coefficient laws in relation to wind speed for three fuel types, where ♦, ■ and • are the reference calibration points [7].*

Two different wind velocities are considered, 5 m/s and 10 m/s, with the previous case of 0 m/s added as a third case scenario (no-wind). Interestingly, the shape of the fire front predicted by the proposed *physics-based model* and the one predicted by the *mechanistic surrogate model* embedded in FARSITE differ considerably for windy conditions. Therefore, it makes little sense to use A_{b} as the reference response. Instead, the downwind flame-front run ($\widehat{\Delta X}$) is used, and the objective function in Eq. (3.27) is replaced by the one in Eq. (3.30) for the calibration of $\boldsymbol{c}_{\mathrm{w}}$ [7].

$$\underset{c_{\mathrm{w}}}{\text{Minimise}} \quad \left(\widehat{\Delta X} - \widehat{\Delta X_{\mathrm{ref}}}\right)^2 \tag{3.30}$$

The fire front for each of these three scenarios is shown in Fig. 3.3, where the predictions made by the proposed *calibrated* physics-based model are displayed in red lines whereas those made by the commercial simulator FARSITE are displayed in black lines. As can be observed, the $\widehat{\Delta X}$ predicted by the two models coincide whereas the predicted values of the A_b show great discrepancies. In particular, FARSITE predicts significant expansions of the fire perimeter against the wind and on both flanks. This is an artificial effect introduced by the geometrical construction of the flame front based on elliptic waves with their back focuses forcibly placed on the ignition points in the fire perimeter (Huygens principle). This is a known issue associated with the embedded 2D fire growth model. It is important to note that, while the A_b depends on the elliptical growth (mechanistic surrogate) model, the size of the major axis depends only on the RoS in the direction of the wind calculated in advance in FARSITE using Rothermel's model. This supports the case for using $\widehat{\Delta X}$ rather than A_b as the reference variable for the calibration of $\boldsymbol{c}_\mathrm{w}$ [7].

Since aerodynamic coefficients are usually modelled proportionally to a certain power of the wind velocity [95], it is assumed here that c_w follows a quadratic law, as shown in Fig. 3.4. Thus, the calibrated $\boldsymbol{c}_\mathrm{w}$ corresponding to the three wind velocities are interpolated with a quadratic polynomial for each of the three fuels considered. Fig. 3.4 also shows that FM1 (blue) and FM2 (red) follow marked parabolic trends whereas FM6 (green) displays an almost linear one. This is because FM6 consists of bushes with foliage which, despite being very flammable, are coarse. Hence fire propagation is strongly driven by wind, which can almost freely penetrate the highly porous media somewhat justifying the quasi-linear trend of c_w.

Two examples of fire spread from a single source predicted by the calibrated model are shown in Figs. 3.5–3.8 for fuel FM1 uniformly distributed in the domain. Figs. 3.5 and 3.6 show the temperature and fuel energy fields, respectively, for the no-wind case whereas Figs. 3.7 and 3.8 show them for a 10 m/s wind. Note that any non-white region in the fuel energy field contributes to the total A_b. The outer solid red lines in Figs. 3.6 and 3.8 are the fire perimeters modelled by FARSITE. Clearly, agreement is almost perfect for the no-wind case and deteriorates once wind is introduced, as previously discussed. The inner dotted contours in Figs. 3.7 and 3.8 enclose regions with $T < T_\mathrm{ig}$.

3.3.2 Model tests

In order to test the proposed FireProM-F beyond the calibration scenarios, two sets of numerical experiments are carried out in this section: one for different wind speeds, and the other for different inhomogeneous fuel mixtures. Predicted fire perimeters are compared against those modelled by FARSITE, and the satisfaction of the soft T_max constraint is checked.

Tests under different wind speeds

The first set of experiments consists of testing the calibration curves in Fig. 3.4 to set the c_{w} coefficient in the transport term in Eqs. (3.4) and (3.10) as a function of the wind velocity. Thus, wind velocities were selected within the calibration range [0 m/s–10 m/s], but also outside by extrapolating the curves up to 15 m/s. I believe that the parabolic formulation of c_{w} would still work for stronger winds, though this has not been tested here.

As shown in Figs. 3.9, 3.10 and 3.11, the discrepancies in the predictions of the fire-front runs ($\widehat{\Delta X}$) by this calibrated model and FARSITE are smaller than 5% for each of the three fuels, even for wind velocities larger than 10 m/s. This calibration may also be applied to the remaining standard fuels described by Anderson [94].

Tests under different fuel mixtures

While the model has been calibrated for individual fuels, the aim here is to test its predictions for a mixture of them. Thus, the three different fuel types have been allocated randomly throughout the domain in order to simulate a plausible inhomogeneous distribution of fuel energy. The fuels are mixed randomly but predefining their occupational percentages (C_{FM1}, C_{FM2} and C_{FM6}), with each cell containing only one type of fuel. Tests are performed for two ignition cases: single-point ignition in the centre of the domain (as during calibration), and multiple ignitions from four sources. An example of the fire propagation when mixing the three fuel models in equal percentages and for a fire initiated from four ignition points is shown Figs. 3.12 and 3.13. Since calibration was performed for reference values extracted 10 min after ignition, simulations are carried out for an extended period of 20 min to confirm that the agreement persists [7].

As can be observed in Figs. 3.12 and 3.13, the predictions of the fire perimeter by the calibrated model and by FARSITE are in clear agreement. Also note that the maximum temperature in the field is around the value set for $T_{\max}$ during calibration (1,200 K).

Percentage errors (ϵ) for pairwise mixtures of the three fuels in different occupational percentages are shown in Figs. 3.14, 3.15, and 3.16. It was expected that ϵ for the mixture would monotonically vary between the values at the two extremes with single fuel, e.g. along the red lines in the figures. However, the discrepancies between the predictions of the A_{b} increase when mixing fuels, even if there is some skewness towards the fuel with higher ϵ. This can be caused by a different modelling of the fire propagation in the neighbourhood of an interface between two different fuels. For instance, FARSITE averages the properties of the fuels traversed by the fire front during the time-step considered [10, 96], whereas FireProM-F chooses the parameters as previously explained in Table 3.1. Nonetheless, discrepancies of predictions are consistently maintained below 5% for different mixtures of fuels [7].

Analysing the performance of the model for cases with multiple ignitions, a 3%

increment of ϵ is observed for a fire originating from four ignition points when compared to one propagating from a single source. This may be –at least partially– due to the geometrical formulation of the fire front in FARSITE, as marker methods are known to have difficulties in handling situations such as when the edges of a concave fire front –or two separate fire fronts– collide. With the current crossover management algorithm, the superposed area is simply eliminated disregarding the energy that theoretically has been generated in producing it. Furthermore, in cases of junction fires and concave fire fronts, experiments show that the RoS should momentarily increase and then gradually slow down to reach normal speed once the shape becomes convex [97]. As opposed to FARSITE, the presented model appears to reproduce this phenomenon to some extent [7].

3.3.3 Reproducing real fire experiments

This section is aimed at assessing qualitatively FireProM-F. The predictions are compared with findings of well established historical experiments [98, 99]. The reference field experiments are performed on 1 hectar square lots with different kind of grass fuels for characterisation. Some properties are the burning rate, the RoS, susceptibility to wind, and flame intensity. In this section the main feature that is going to be considered is the evolution of the shape of the fire front predicted with the presented model [7]. Since FireProM-F has not yet been calibrated for a wide variety of fuels, here other features (such as the RoS) are not considered. As suggested buy [98, 99] experiments, the dynamics of the shape of the firefront are the following (refer to Fig. 3.17):

1. Using torches, two firemen ignite one edge of the fuel bed (west in Fig. 3.17), starting from the centre and walking apart from each other aiming for the vertices.

2. The wind (10 m/s) blowing from west to east, forces the forming fire front to assume a characteristic Gaussian shape (a).

3. This will quickly evolve into a triangular (b) shape due to larger heat transport towards convex areas.

4. Eventually the flame front obtains a more rounded shape from tear-drop (c) to final parabolic (d) shape.

As demonstrated by Fig. 3.17, FireProM-F is capable to predict also this evolutionary feature of real fires.

3.3.4 Simulation under realistic scenario

This section aims to simulate the fire propagation over a realistic large-scale scenario in order to demonstrate the reliability of the predictions made by the proposed FireProM-F. The realistic scenario consists of a square lot of forest of 1500×1500 m^2 extracted from a tutorial example provided by FARSITE. Topographic maps like this one are

usually provided by the Geographic Information System (GIS), and can provide information about the vegetation cover and type. The lot, shown in Fig. 3.18, comprises fuels FM1, FM2 and FM6, as well as an obstacle through which the fire is not allowed to propagate but the wind may cut through unaffected. Such an obstacle may be given by a natural barrier like a lake or rocks. It is important to note that the precision of the vegetation raster provided by the GIS (≈ 30 m) is not the same as the cell-size used in the simulations (1 m^2). Slope and elevation are set to flat and sea-level conditions, as the presented model does not yet include these features. Moisture level is set to 20% as during calibration [7].

Two experiments are carried out for a wildfire propagating from a single ignition point, one under no-wind condition and the other with a 5 m/s wind blowing at a 60° angle. The temperature field after 3 hrs of propagation for the first experiment is shown in Fig. 3.19, where the flame front is displayed by the outer dotted white contour. The inner white contour encloses a region with $T < T_{\text{ig}}$. The fire front predicted by FARSITE is not displayed because it overlaps almost exactly with the one predicted by FireProM-F. For the second experiment, the temperature field after 3 hrs of propagation is shown in Fig. 3.20, where the meaning of the dotted white contours is the same as in the first experiment. However, the fire perimeter predicted by FARSITE is now displayed in a solid magenta contour. Predictions in this case are not perfectly overlapping but still appreciably similar [7].

It is interesting to notice from Figs. 3.19 and 3.20 that certain areas within the field are observed to retain higher temperatures than neighbouring areas. If these figures are analysed in conjunction with Fig. 3.18, it can be inferred that these areas correspond to the most energetic fuels, namely FM2 and FM6, which are harder to be ignited but provide more energy. All things considered, the results from the two experiments are satisfactory, fulfilling the expectations [7].

In Table 3.4 the computational times for different domain sizes are collected. It can be noticed that for a fairly large domain of 1 hectare the simulation is faster than real time. The turning point when the run time and the simulation time are identical is about 4 hectares [7].

Domain size $[m^2]$	$\boldsymbol{N}$ $[\times 10^3]$	**Simulation time** [h]	**Run time** [h]	**Ratio** $[\sim]$
100×100	10	0.33	0.25	0.75
600×600	360	2.00	3.00	1.50
1500×1500	2250	3.00	28.00	9.33

Table 3.4: *Computational time for different domain sizes (N number of points) performed with $\Delta x = 1$ m and $\Delta t = 10$ s, using Matlab R2017a on a Windows machine with an Intel Xeon CPU @3.20 GHz [7].*

3.4 Conclusion

Wildfires are dangerous uncontrolled phenomena which can have devastating health, social, economic and environmental impacts. Their frequency and severity have been increasing for decades. Therefore, the development of tools to support the management of wildfires is gaining interest as this becomes an increasingly pressing issue worldwide [7].

Traditionally, fire propagation models were aimed either at the development of fire danger rating systems or at the accurate quantification of wildfire events [3]. I propose that their use be extended to two new crucial applications: simulation-based testing of fire management and suppression technologies; and integration into fire suppression technologies as a predictive tool to support autonomous decision-making [7].

Advanced physics-based models which incorporate a wide range of fire dynamics phenomena such as FDS [15] are computationally intensive and cannot be used in real time. In turn, operational fire spread models such as FARSITE [10] are useful as real-time management tools, but they are not physics-based. This chapter presented the mathematical formulation, numerical solution, calibration and testing of a two-dimensional physics-based **Fire Pro**pagation **M**odel for **F**ast simulations (FireProM-F). A major motivation is to support the design of innovative fire management and suppression technologies and strategies, as well as to function as a decision-support tool to assist firefighters in the use of current technology. Hence the need for the model to be physics-based and computationally inexpensive: the modelling of interactions between fire and suppressant must be feasible, and simulations must be in real-time or faster [7].

Despite a number of simplifying assumptions aimed at making the model quick, the underlying physics is comprehensive enough to capture the major phenomena that govern the behaviour of real fires. As a first step towards validation, eight parameters with physical meaning were calibrated within realistic ranges using FARSITE to generate reference values, as this widely used commercial simulator has been shown to reliably reproduce historical fires. To this end, seven parameters were calibrated to match the burnt area and one to match the downwind fire-front run for three different uniform fuels and three wind speeds [7]. This calibration may be straightforwardly extended to the remaining standard fuels described in [94].

Several numerical experiments were carried out to test the calibrated model, including one in a medium-scale realistic scenario with mixed fuels, natural barrier, and atmospheric wind. The fire perimeters predicted by FireProM-F and by FARSITE show nearly perfect agreement for no-wind conditions. Agreement deteriorated once wind was introduced, especially in terms of rates of spread (RoS) against the wind and on the flanks. This is likely to be due to the elliptical growth geometrical approach adopted by FARSITE, which is believed to overestimate the RoS in those directions [7].

Overall, the numerical experiments showed that FireProM-F is able to produce fast and reliable predictions of the fire perimeter, with the added benefit of being physics-

based. For a grid of 4×10^4 points on a standard computer with Intel Xeon CPU (@3.20 GHz), FireProM-F simulates 1 min of fire dynamics in about 58 s (faster than real-time). For larger-scale experiments the computational effort required is evidently higher so the simulations are slower than real-time [7].

Among other things, the use of autonomous technologies to operate in dangerous and hazardous environments, such as wildfires, are currently gathering momentum and gaining interest from researchers and practitioners. For instance, Innocente and Grasso [1] propose the use of self-organising swarms of drones for autonomous firefighting. These types of technologies require models like FireProM-F for their development and extensive testing under realistic scenarios. Furthermore, a faster-than-real time model may also be used as a predictive tool to enable more advanced autonomous firefighting strategies [7].

For future work, it is paramount that FireProM-F incorporate the effects of topography. A possibly more challenging endeavour consists of the development of efficient means to incorporate balances of mass and momentum into the 2D model without noticeably increasing the computational weight. While the aim of this research was the development of a fast 2D physics-based model, a more advanced 3D high-fidelity model is currently under development for offline simulations aiming to provide better insight into the dynamics of the interactions between the fire and different suppressants [7].

3.5 Summary

Scope: Develop a two-dimensional wildfire propagation model for further integration with the Swarm-in-the-Loop system for designing firefighting strategies.
The wildfire propagation model ought to be:

- *mechanistic*: physics based in order to accurately model the effect of different extinction technologies (e.g. water dropping, foams, retardants, powders, disrupting sound or shock waves, etc.);
- *parametric*: in order to model a multitude of fire dynamics associated with different vegetation fuel types (e.g. short or tall grass, shrubs, trees, peat, mixed) and moisture conditions;
- *realistic*: which reproduces with very good appreciable similarity what would be the actual natural fire dynamics with heterogeneous fuel distribution (e.g. rate of spread, fire front shape, temperature profile, consumed fuel areas, etc.);
- *faster-than-real-time*: computationally lightweight, necessary for each time integration to be computed fast enough to be integrated in real time in loop with some hardware, such as a swarm of drones indoor.

Method, studies, and experiments:

- *Mathematical formulation of the fire propagation model* consisting of a combustion reaction equation for mass balance and an energy equation (subdivided into combustion energy source, wind transport, conduction, interdiffusional enthalpy, horizontal local radiation, outward convection and radiation sinks terms).
- *Numerical solution* with finite differences method (FDM): 2^{nd} order centred FDM for the spatial integration and the 4^{th} order Runge-Kutta method (RK4) for the time integration.
- *Calibration of eight parameters* with physical meaning bounded by realistic ranges using FARSITE [10] to generate reference values for three different vegetation fuel types (FM1: short grass; FM2: timber; and FM6: dormant brush and hardwood slash). Seven parameters were calibrated to match the burnt area, and one to match the downwind fire-front run.
- *Numerical experiments* to test the calibrated model. Nearly perfect agreement on fire perimeters between FireProM-F and by FARSITE, apart across the flanks and against the eventually introduced wind.

Main outcomes and findings:

- *FireProM-F*: a mechanistic, parametric, realistic and faster-than-real-time 2D wildfire propagation model. It is able to produce fast and reliable predictions of the fire perimeter, with the added benefit of being physics-based.
- The solution of the model is *not as fast as FARSITE*, but it has an acceptable compromise of the properties listed above. In any case, the development of FireProM-F was never a competition with FARSITE.
- A model of *smoke generation* should be added in order to emulate the decreased visibility for the Swarm-in-the-Loop (SwiL) simulation.
- A model of *vertical wind draft* associated to the buoyancy fluxes (vertical convection) should be formulated in order to realistically affect the flight dynamics of each drone in the SwiL.

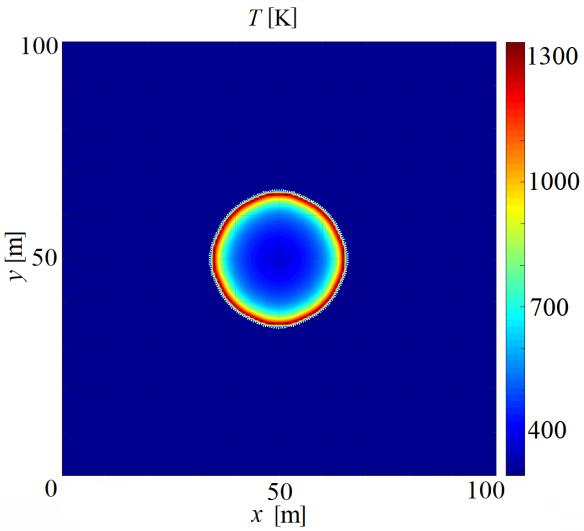


Figure 3.5: *Temperature field 10 min after single-point ignition of FM1 fuel bed under no-wind condition [7].*

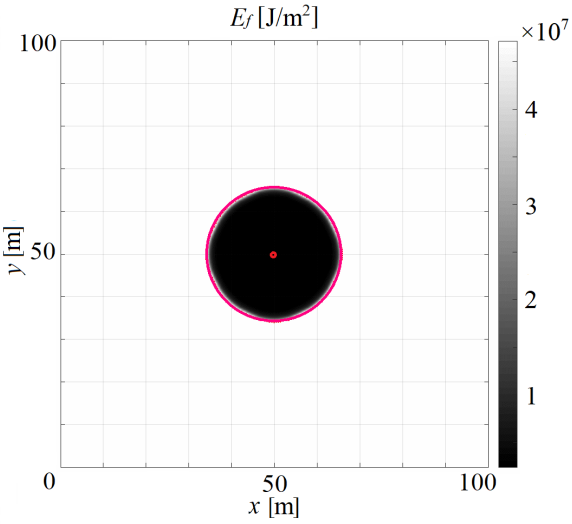


Figure 3.6: *Energy fuel field 10 min after single-point ignition of FM1 fuel bed under no-wind condition. The grey region shows the burnt area (A_b), the red dot shows the ignition point, and the red circle is the FARSITE prediction [7].*

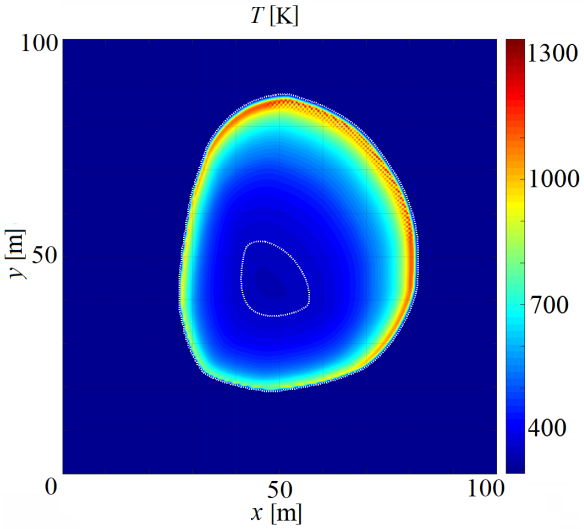


Figure 3.7: *Temperature field 10 min after single-point ignition of FM1 fuel bed with 10 m/s wind. The inner contour encloses a region with $T < T_{ig}$ [7].*

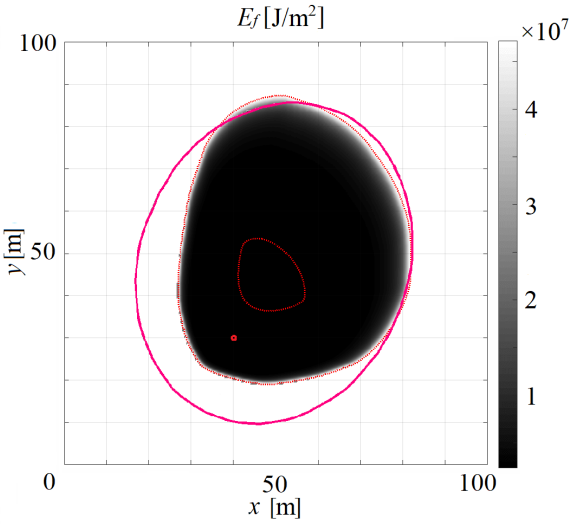


Figure 3.8: *Energy fuel field 10 min after single-point ignition of FM1 fuel bed with 10 m/s wind. The grey region shows the burnt area (A_b), and the red ellipse is the FARSITE prediction. The inner contour encloses a region with $T < T_{ig}$ [7].*

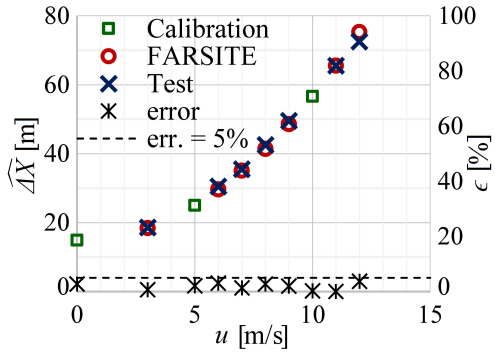


Figure 3.9: *Fire-front run $\widehat{\Delta X}$ predicted by the model and by FARSITE, and their difference (error), for a range of wind velocities and FM1 [7].*

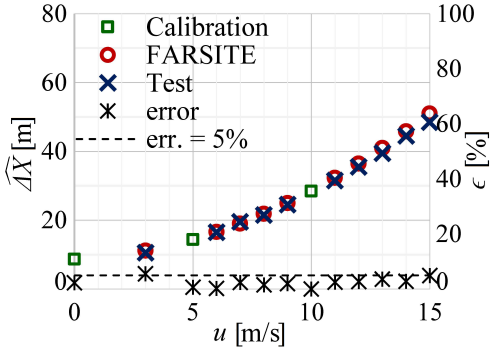


Figure 3.10: *Fire-front run $\widehat{\Delta X}$ predicted by the model and by FARSITE, and their difference (error), for a range of wind velocities and FM2 [7].*

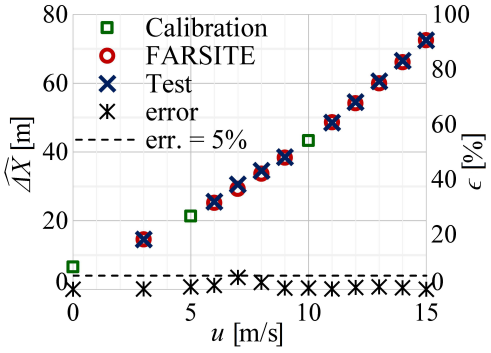


Figure 3.11: *Fire-front run $\widehat{\Delta X}$ predicted by the model and by FARSITE, and their difference (error), for a range of wind velocities and FM6 [7].*

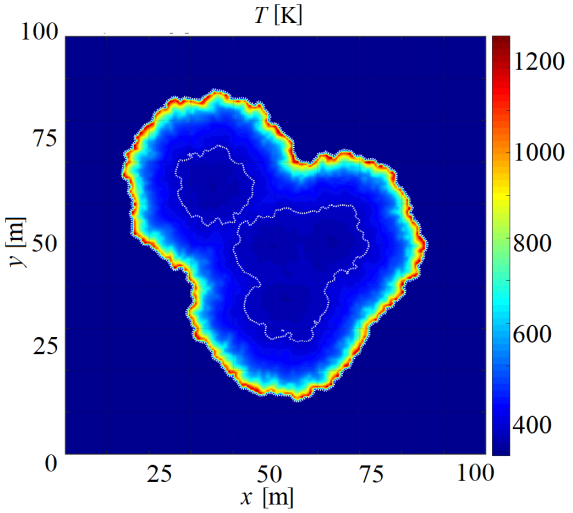


Figure 3.12: *Temperature field 20 min after multiple ignition of a randomly mixed fuel bed with equal partition of FM1, FM2 and FM6. The inner contour encloses a region with $T < T_{ig}$ [7].*

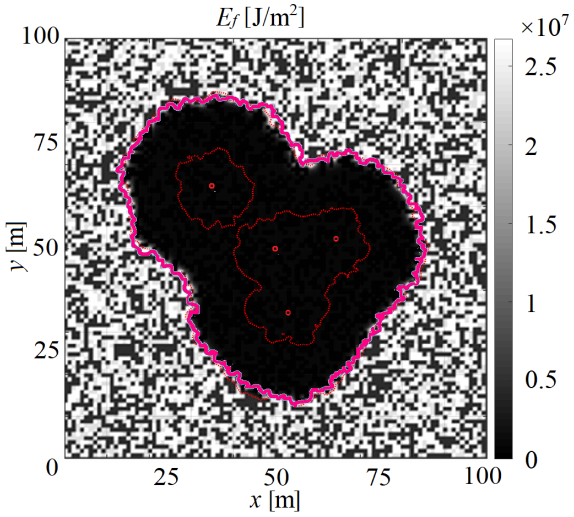


Figure 3.13: *Fuel energy field and fire perimeter 20 min after multiple ignition (red dots) of a randomly mixed fuel bed with equal partition of FM1, FM2 and FM6. The uniform dark region with low density of fuel energy depicts the A_b predicted by FireProM-F whereas the red contour is FARSITE's prediction of the fire perimeter. The inner contour encloses a region with $T < T_{ig}$ [7].*

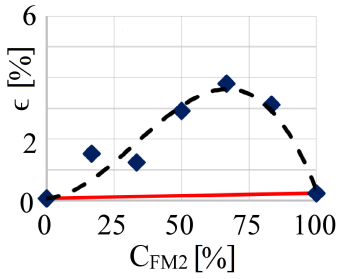


Figure 3.14: *Estimation error distribution for different percentages of two randomly mixed fuel types: FM1 and FM2. In continuous line the expected trend. At 0% of FM2 corresponds 100% FM1 [7].*

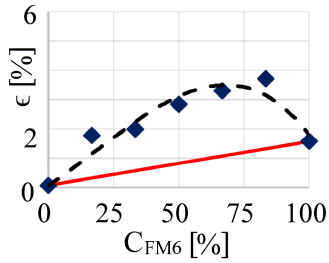


Figure 3.15: *Estimation error distribution for different percentages of two randomly mixed fuel types: FM2 and FM6. In continuous line the expected trend. At 0% of FM6 corresponds 100% FM1 [7].*

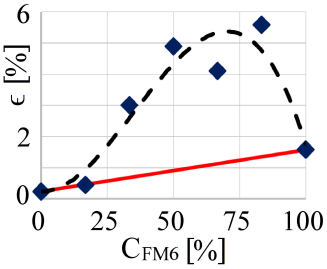


Figure 3.16: *Estimation error distribution for different percentages of two randomly mixed fuel types: FM1 and FM6. In continuous line the expected trend. At 0% of FM6 corresponds 100% FM2 [7].*

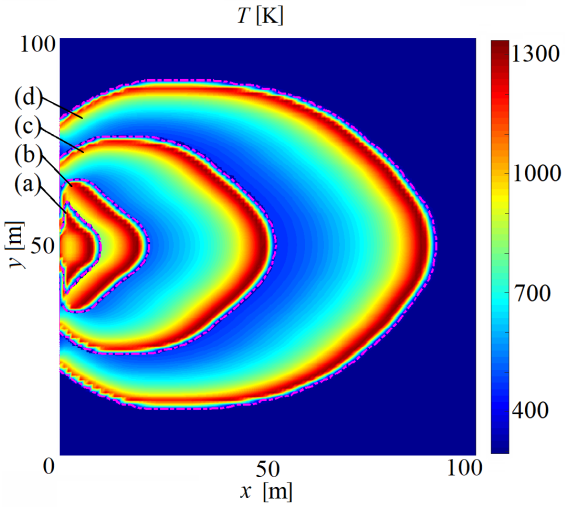


Figure 3.17: *Temperature field after dynamic linear ignition on the western side of a* 100×100 m^2 *FM1 field blown by a 10 m/s wind. It is a composite map of time isopleths of fire perimeter (dotted magenta line) and depth at (a) 48 s* gaussian, *(b) 114 s* triangular, *(c) 301 s* tear-drop, *and (d) 538 s* parabolic *fronts respectively [7].*

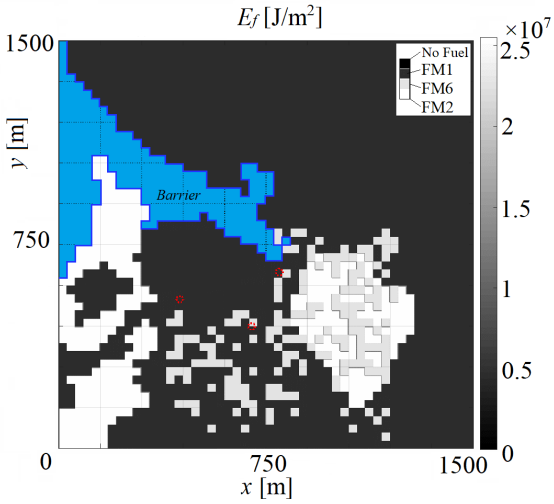


Figure 3.18: *Fuel energy distribution over a 1500×1500* m^2 *wildland lot. The black area is an obstacle, while the three different scales of grey represent the fuel types FM1, FM2, and FM6. Initial flame contour shown in red [7].*

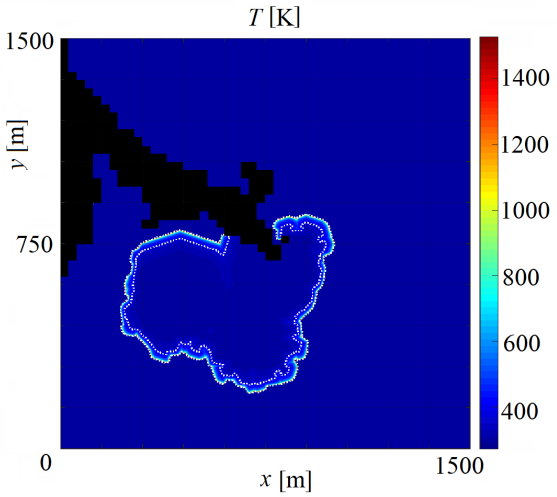


Figure 3.19: *Temperature field 3 h after three-point ignition (see Fig. 3.18 for their location) under no-wind condition. The outer dotted white contour displays the fire front whilst the inner one encloses a region with $T < T_{ig}$ [7].*

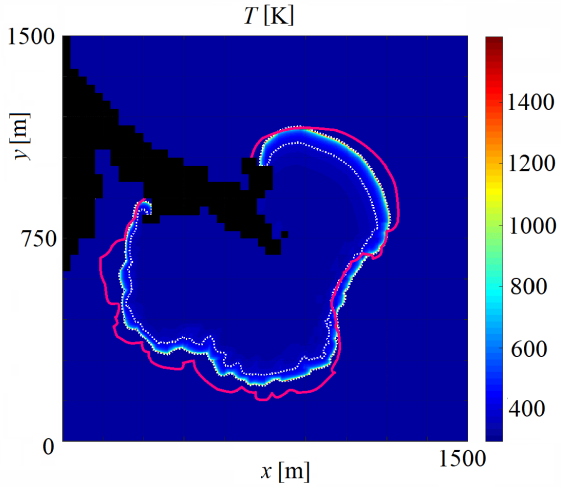


Figure 3.20: *Temperature field 3 h after three-point ignition and 5 m/s wind. The outer dotted white contour displays the fire front whilst the inner one encloses a region with $T < T_{ig}$. The magenta contour is the fire front predicted by FARSITE [7].*

Chapter 4

Autonomous Firefighting

This chapter aims to demonstrate the feasibility and potential of employing swarm robotics (SR) to fight wildfires autonomously. The focus is not on the design of the physical robots but on their self-coordination mechanisms for the desired firefighting behaviour to emerge. To this end, an efficient yet realistic physics-based model of wildfire propagation and a self-organisation algorithm for swarms of fire-fighting drones are developed and coupled, with the collaborative behaviour based on a particle swarm algorithm adapted to individuals operating in physical dynamic environments of high severity and frequency of change. The introductory Chapter 2 presented a background and related work on fire propagation models, swarm intelligence (SI), SR, and the particle swarm algorithm. Section 4.1 describes the proposed system of self-organising swarms of firefighting drones. Numerical experiments are carried out in Section 4.2 and simulation results are discussed. Besides, the used wildfire propagation model is formulated and calibrated in Chapter 3 (FireProM-F). Finally, conclusions and recommendations for future work are offered in section 4.3 followed by a brief summary of the chapter's topic, scope, method and findings in section 4.4.

4.1 Self-organising swarm

Since the PSO algorithm was inspired by models of decentralised flocks of birds, its potential to prescribe the desired behaviour of a swarm of self-organising drones is promising. A detailed study of PSO is beyond the scope of this chapter. It suffices to say that the PSO objective is to find the best position on the considered wildfire domain where a certain quantity is maximised. Such objective could be to find the highest temperature on the collective fire-front in order to act with the selected extinction technology - e.g. dropping water. Deciding the firefighting strategy affects the definition of the objective. Such objective is formulated in the PSO algorithm as local and global attractors, that can be combined into a single attractor at a given time-step (t) as follows [100]:

$$\phi_{ij}^{(t)} = \iota_{ij}^{(t)} + \sigma_{ij}^{(t)} = iw_{ij}^{(t)} U_{0,1} + sw_{ij}^{(t)} U_{0,1} \tag{4.1}$$

$$p_{ij}^{(t)} = \frac{\iota_{ij}^{(t)} xb_{ij}^{(t)} + \sigma_{ij}^{(t)} xb_{kj}^{(t)}}{\phi_{ij}^{(t)}} \tag{4.2}$$

Clearly, the effect of the incorporated randomness is two-fold: it affects the trajectory of a particle towards the overall attractor ($\mathbf{p}$), while also affecting the generation of this attractor as a stochastic convex combination of the individual ($\mathbf{xb}_i$) and the social ($\mathbf{xb}_k$) attractors. Different from classical formulations, these two features of the algorithm are decoupled here. Thus, the attractor is generated at every time-step from a uniform distribution within the rectangle generated such that its edges are parallel to the coordinate axes and it contains the current individual and social attractors as vertices. Once the attractor $\mathbf{p}$ is generated, the trajectory difference equation is as in (4.3) [1].

$$x_{ij}^{(t+1)} = x_{ij}^{(t)} + \omega_{ij}^{(t)} \left(x_{ij}^{(t)} - x_{ij}^{(t-1)} \right) + \phi_{ij}^{(t)} \left(p_{ij}^{(t)} - x_{ij}^{(t)} \right) \tag{4.3}$$

The random variable can be realised as in (4.1) so that the probability distribution is triangular or trapezoidal as in the classical formulation, or a different density function can be chosen. Analysing the trajectory of a single particle with stationary attractor and constant coefficients, the settings of ω and ϕ control the type of behaviour [1]. For optimisation purposes, convergent high-frequency harmonic oscillations are generally preferred, as there is no cost attached to large displacements. Conversely, low-frequency harmonic oscillations and smaller displacements are favoured for swarm robotic systems. The settings in (4.4) are adopted, which ensure convergent low-frequency harmonic oscillations, with randomly generated from a uniform distribution within this range [1].

$$\begin{cases} \omega \in (0,1) \\ \phi \in [(\sqrt{\omega}-1)^2, (\omega+1)] \end{cases} \tag{4.4}$$

A study of the influence of the coefficients in the trajectory difference equation on the types of behaviour of a particle is beyond the scope of this chapter, besides more information can be found in [100, 1]. While local neighbourhood topologies and other types of sociometries like distance-based nearest neighbours and speciation need to be investigated, the global neighbourhood topology is adopted for the initial studies presented in this formulation [1].

The classical PSO requires some adaptations to cope with dynamic environments by addressing the problems of *outdated memory* and *diversity loss* [101]. The propagation of fire leads to a dynamic environment with multiple hot-spots frequently and severely changing location and intensity. This feature is reinforced by drones attempting to suppress the fire and therefore modifying the environment and adding a form of stigmergy to their direct communication. An usual solution adopted in PSO is to reinitialise the memory of the stagnated particle, but this is not possible in the case of SR, since the physical drones cannot be instantly repositioned. Both the outdated memory and the diversity loss problems are addressed by erasing the individual memory of a drone when it has not been updated in the preceding 10 seconds, and then randomly re-initialising the memorised position. Thus, outdated memories are deleted

and diversity is increased smoothly by new memories pulling the drones outwards via the trajectory equation rather than by an instant relocation.These new memorised positions are assigned temperatures one degree below ignition to avoid the search to continue once the fire is believed to have been suppressed [1].

The formulated firefighting strategy is the following: each drone searches for hotspots in the field, and every time it finds a location that is hotter than the hottest one stored in its individual memory, a third of its total water payload is dropped. After three drops of water payload, it must go back to the water source to replenish. Like-wise, once a drone has travelled its total flying range, it must go back to its recharging docking station. Figure 4.1 provides a high-level description of the proposed self-organising swarm of firefighting drones coupled with the developed physics-based model of fire propagation (described in Chapter 3). This flowchart is colour-coded for clarity, identifying five main sections and providing explanatory text-boxes that briefly describe the main tasks performed within each [1].

4.1.1 Initialisation

The settings for the physics-based fire propagation model and those for the agent-based model of the swarm of drones are entered independently, defining both the wildfire scene and the firefighting tool. One or more sparks are randomly generated to initiate the fire, and the drones are placed on their recharging docking stations fully charged and fully loaded with water at the beginning of the simulation ($t = 0$). The variables for the PSO-based self-coordination mechanism are initialised for each drone. Namely, the temperature at its current location (T) is measured, its memory storing the location of its best experience ($\mathbf{xb}$) is generated randomly from a uniform distribution within the field, and the temperature associated is fictitiously set to the ignition temperature ($Tb = T_{\text{ig}}$). Every drone is set to firefighting mode, and its target location is set the same as the individual memory ($\mathbf{xt} = \mathbf{xb}$) [1].

4.1.2 Update of fire model

The 2D fire model is updated by numerically computing its evolution for one time increment of the simulation, therefore requiring the discretisation of the space and time domains (Δx, Δy, Δt). From the initial fire sparks until the swarm of drones is launched, the fire propagates unchecked. In a real scenario, automated wildfire detection technology would include stationary (ground-based) visual systems, ground-based sensors, manned and unmanned surveying aircrafts, and/or satellite monitoring. For the purpose of these studies, the fire is simply allowed to propagate unchecked for a given amount of time (t_{launch}) before the drones are informed that a fire has been detected within the region of interest (no precise location is given). Once the firefighting activities start, the drones may affect the temperature field between updates of the fire model [1].

4.1.3 First loop through drones

This loop is nested within the time loop simulating the evolution of the whole system from $t = 0$ to $t = t_{\max}$, with a given time increment (Δt) between iterations. For each drone, its current target is identified, which states whether it must proceed to fight the fire, move towards the docking recharging station, or move towards the water source for replenishment. When a drone decides it must move towards the recharging docking station, whatever water is left in the tank is dropped at its current location. In such a case, the fire field needs to be updated by computing the new temperatures at the nodes affected by the drop. It is considered that a drop only affects instantaneously the four nodes from the discretised space which are closest to the coordinates of the drop. Thus, coefficients of influence are calculated for each of these four nodes as shown in (4.5) [1].

$$A_{ij} = \frac{\Delta x - |x - x_j|}{\Delta x} \cdot \frac{\Delta y - |y - y_i|}{\Delta y} \tag{4.5}$$

The fraction of the mass of the water dropped that affects each node is given by its coefficient of influence. The temperature at each of the four nodes is updated by means of a mass-based weighted average between the current nodal temperature and the temperature of the water dropped, as in (4.6) [1].

$$T_{ij} \leftarrow \frac{A_{ij} m_{\text{drop}} T_{\text{water}} + m_{\text{gas}} T_{ij}}{A_{ij} m_{\text{drop}} + m_{\text{gas}}} \tag{4.6}$$

The mass of the gas mixture of fuel and air (m_{gas}) is determined by the density of the gas (ρ) multiplied by the volume represented by a node: $\Delta x \cdot \Delta y \cdot z_{\text{th}}$, where z_{th} is the thickness of the gas layer provided as an input to the fire model. Note that $\mathbf{A}$ is a 2×2 matrix that contains the four coefficients of influence for the nodes in question. The indices (i, j) are meant to facilitate the identification of the nodes they apply to. For instance, A_{21} applies to (x_j, y_i). Whether the update of the fire field has been completed or unnecessary, the drone then moves towards its current target,whichever this may be, always ensuring that the maximum velocity permitted is not exceeded. Evidently, this physical displacement requires the update of the drone's remaining flying range. In this formulation, the latter is simply measured in terms of the distance that can still be travelled before requiring a recharge. It goes without saying that the drone's payload is fully refilled and its flying range reset to the maximum once it reaches the water source and there charging docking station, respectively. The outdated memory and the diversity loss problems are addressed by erasing the individual memory of a drone when it has not been updated in the preceding 10 seconds, re-initialising its memorised best position randomly, and setting the associated temperature one degree below ignition [1].

4.1.4 Update of highest temperature in collective memory

Since the drones operate in parallel, it is more realistic to update the global memory in a synchronous manner once all individual memories are up-to-date. Therefore, it needs to be carried out outside the first loop through drones discussed in the previous section. This update simply consists of extracting the location and temperature of the hottest spot from the current memories of all drones [1].

4.1.5 Second loop through drones

The choice of a synchronous update of the global memory makes this loop necessary. Otherwise, both the asynchronous update of the swarm's best experience and the tasks described below could be performed at the end of the first loop through the drones discussed in Section 4.1.3. Hence the two loops through the drones are sequential and nested within the loop through time that simulates the evolution of the whole system. Note that the second loop is only executed every $5\Delta t$. As discussed before, the PSO formulation proposed here decouples the generation of the overall attractor and the evaluation of the trajectory difference equation. Thus, the overall attractor for drone i at time t ($\mathbf{p}_i^{(t)}$) is stochastically generated from a uniform distribution within the rectangle spanned between the location of the individual best experience ($\mathbf{xb}_i^{(t)}$) and that of the global best experience ($\mathbf{xb}_k^{(t)}$) as in (4.7). Sub-index k identifies the drone holding the global best experience [1].

$$p_{ij}^{(t)} = xb_{ij}^{(t)} + U_{(0,1)}\left(xb_{kj}^{(t)} - xb_{ij}^{(t)}\right) \tag{4.7}$$

The acceleration coefficient ($\phi_{ij}^{(t)}$) is generated independently from the overall attractor as in (4.8), and the couple (ϕ, ω) is kept within the interval associated with low-frequency harmonic oscillatory behaviour [1].

$$phi_{ij}^{(t)} = (\sqrt{\omega} - 1)^2 + U_{(0,1)}\left[(\omega + 1) - (\sqrt{\omega} - 1)^2\right] \tag{4.8}$$

The target location for drone i for the next iteration $\mathbf{xt}_i^{(t+1)}$ is generated as in (4.9). While the resemblance to the PSO trajectory equation is evident, this is not really a difference equation because it returns $\mathbf{xt}^{(t+1)}$ as a function of $\mathbf{x}^{(t)}$ and $\mathbf{x}^{(t-1)}$. This is the drone's target when in firefighting mode [1].

$$xt_{ij}^{(t+1)} = x_{ij}^{(t)} + \omega_{ij}^{(t)}\left(x_{ij}^{(t)} - x_{ij}^{(t-1)}\right) + \phi_{ij}^{(t)}\left(p_{ij}^{(t)} - x_{ij}^{(t)}\right) \tag{4.9}$$

4.2 Simulation results and discussion

The domain in these experiments is a field of 100×100 m^2 with fuel uniformly distributed except for a small band of 9.6 m around the field so that combustion does

not take place near the boundaries. Thus, the total area covered with fuel equals 6529 m^2 whilst the total fuel energy available is 218.30 GJ. The adopted fuel is short grass, referred to as Fuel Model 1 in [94]. No-wind condition is assumed, and all experiments are run for 480 s [1].

Once the model for fire propagation (FireProM-F in Chapter 3) is developed and runs reasonably fast in the chosen environment size, it is coupled with the proposed model of self-organising swarms of firefighting drones as shown in Figure 4.1. Since the focus in this chapter is on the swarm-intelligent coordination mechanisms so that the drones self-organise to develop the ability to fight fires autonomously and collaboratively, the simulated drones are collision-free at this stage [1]. A stigmergic collision-avoidance algorithm is presented and coupled with this swarm dynamics in the following Chapter 5. Moreover, it is important to note that the model of the firefighting system is stochastic, and therefore multiple runs are necessary for each experiment in order to compute basic statistics required to derive reliable conclusions from the results. This statistical study has been carried out in [1] while here the most salient results are provided.

The fire-extinction success rate highly depends on the number of ignition points and on the total number of drones in the swarm. For instance in the case of only one ignition point the success rate with a swarm-size of 30 drones is 60%, while increasing the size to 70 drones results in an increment of the success rate to 90%.

The success rate depends also on the temperature sampling frequency. Being able to only measure temperature above their exact current positions, and only taking measurements once per second whilst $v_{\max} = 10$ m/s, the swarm is unable to fine-tune the search repeatedly overflying small remaining hotspots. This makes the swarm believe that the fire has been suppressed in a few cases, returning to the docking stations prematurely and letting the fire grow back unchecked. An example of this exemption is shown in Figure 4.2.

As far as the influence on the success rate of the number of ignition points is considered, the fire which started from four sources is evidently more challenging, and a 70-drone swarm now fails to suppress it every time, as depicted in Figure 4.3.

The problem of multiple ignition points is solved by increasing the fire-fighting power to 100 drones. Now, a wildfire originated by four random and simultaneous ignition points is extinguished by a swarm-size of 100 drones with a success rate of 100% as depicted in Figure 4.4.

4.3 Conclusion

Destructive wildfires continue to increase in frequency and severity worldwide. Modern drones are often used to operate in such dangerous environments for monitoring and detection of wildfires, but scarcely for their suppression. This part of the research investigated the feasibility and potential of using self-organising swarms of drones to fight the propagation of wildfires autonomously and collaboratively, without risking

human lives. The focus was not on the design of the physical robots but on their self-coordination mechanisms so that the desired firefighting behaviour would emerge. Thus, an efficient physics-based model of fire propagation (presentend in Chapter 3) and a self-organisation algorithm for swarms of firefighting drones were developed and coupled, with the collaborative behaviour based on a particle swarm algorithm adapted to individuals operating within physical dynamic environments of high severity and frequency of change [1].

The numerical experiments carried out in Chapter 4 and in [1] demonstrated the expected scalability and fault-tolerance of the proposed self-organising swarm of drones as an autonomous firefighting system. It can be concluded that the exploitation of self-organising decentralised collaborative behaviour is a powerful approach and a promising line of research to deal with complex dynamic problems such as the suppression of wildfires. In principle, any wildfire could be suppressed given enough time and a sufficiently high number of firefighting drones – even if such number is unrealistically high with current technology [1].

It is important to note that, whilst water was used as the fire suppressant, the ultimate goal is to exploit more sophisticated technology with higher extinguishing power and comprising a lighter payload for the drones. One key challenge for this technology to become practical is the design of efficient, light-weight and environmentally friendly fire-suppression materials which can be carried by relatively small drones. One of the future research directions lies in the design of such materials based on Class-A foam enhanced by nanoparticles [1].

4.4 Summary

Scope: Demonstrate the employability of swarm robotics (SR) to fight wildfires autonomously. The proposed solution is a *firefighting swarm*, which ought to be:

- *autonomous*: there shall not be any direct control of the drones, the swarm shall be able to self-organise by following decentralised decisions (i.e. the overall swarm behaviour shall emerge from the individual decisions each agent is independently taking);
- *adaptable*: the autonomous swarm should be able to adjust to the ever-changing wildfire shapes and intensities, and should be able to detect new fire spots;
- *scalable*: its self-organisation algorithm should work with any swarm size to tackle different wildfire sizes (i.e. the algorithm should work for any critical number of drones);
- *robust*: the lost of a few agents (i.e. drones) should not be detrimental for the final outcome of the firefighting mission.

Method, studies, and experiments:

- Focused on the *self-coordination mechanisms* of a swarm of firefighting drones: the desired behaviour should emerge.
- Formulated a *fire suppression model*, consisting in the combination of *FireProM-F* (fire propagation model from Chapter 3) and the *swarm dynamics* described in this chapter, along with a *fire detection model* and a *fire extinction model*.
- *Numerical experiments* consist in a simulation of various firefighting missions with different initial conditions (e.g. take off time, position of ignition points), boundary conditions, and initialisation parameters (e.g. initial number of drones). Statistics have been obtained in [1] and presented in this chapter.

Main outcomes and findings:

- A *PSO-inspired self-organisation firefighting algorithm* function of ever-changing mission phases (i.e. staggered take-off, water refilling, refuelling, fire detection and firefighting, suppression check, stand-by hovering, organised landing).
- Self-organising decentralised collaborative behaviour is a *very promising approach* to deal with *complex dynamic problems*, such as the suppression of wildfires.
- Quite *simple* fire detection model (i.e. temperature measurement) and fire extinction model (i.e. dropping water) since the focus was on the self-organisation. Future obligated improvement consists of exploiting more sophisticated technologies, such as lighter materials with higher extinguishing power.
- The following topics have been intentionally overlooked at this stage and should be considered for further studies: the *uncertainty in the environment*, the *stochasticity in the algorithms*, and *flight dynamics*.
- Since the drones converge on a single dynamically changing attractor (coordinates of the point with a combination of best experiences - i.e. highest detected temperature), far too many unwanted collisions have been counted, dozens of collisions every second for a swarm of 100 drones; therefore the development of a *collision-avoidance algorithm is of utmost importance*. Given the high number of drones, such c.a. algorithm must be decentralised (see following Chapter 5).

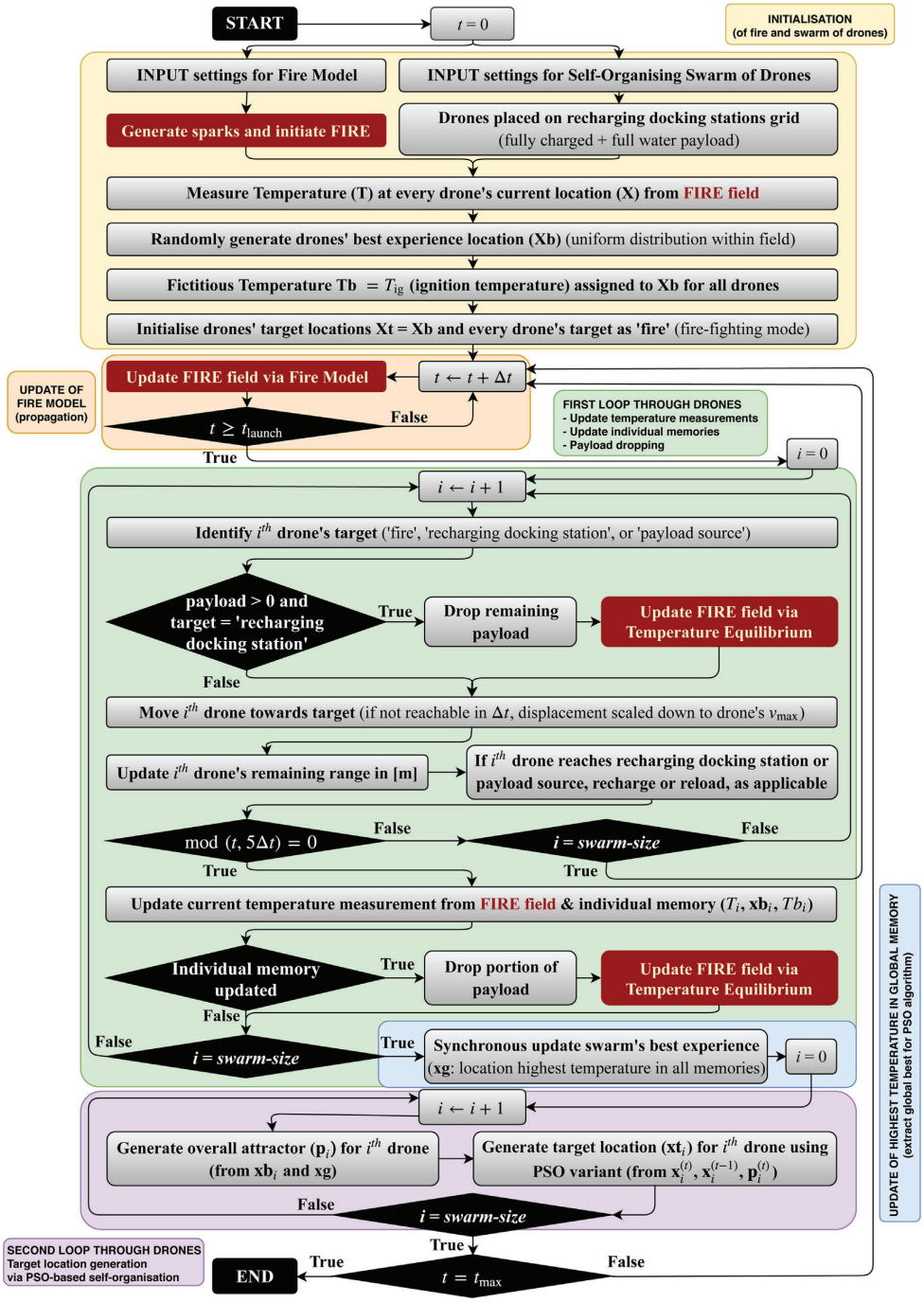


Figure 4.1: *Flowchart with a high-level description of the model implementation for the proposed self-organising swarms of firefighting drones [1].*

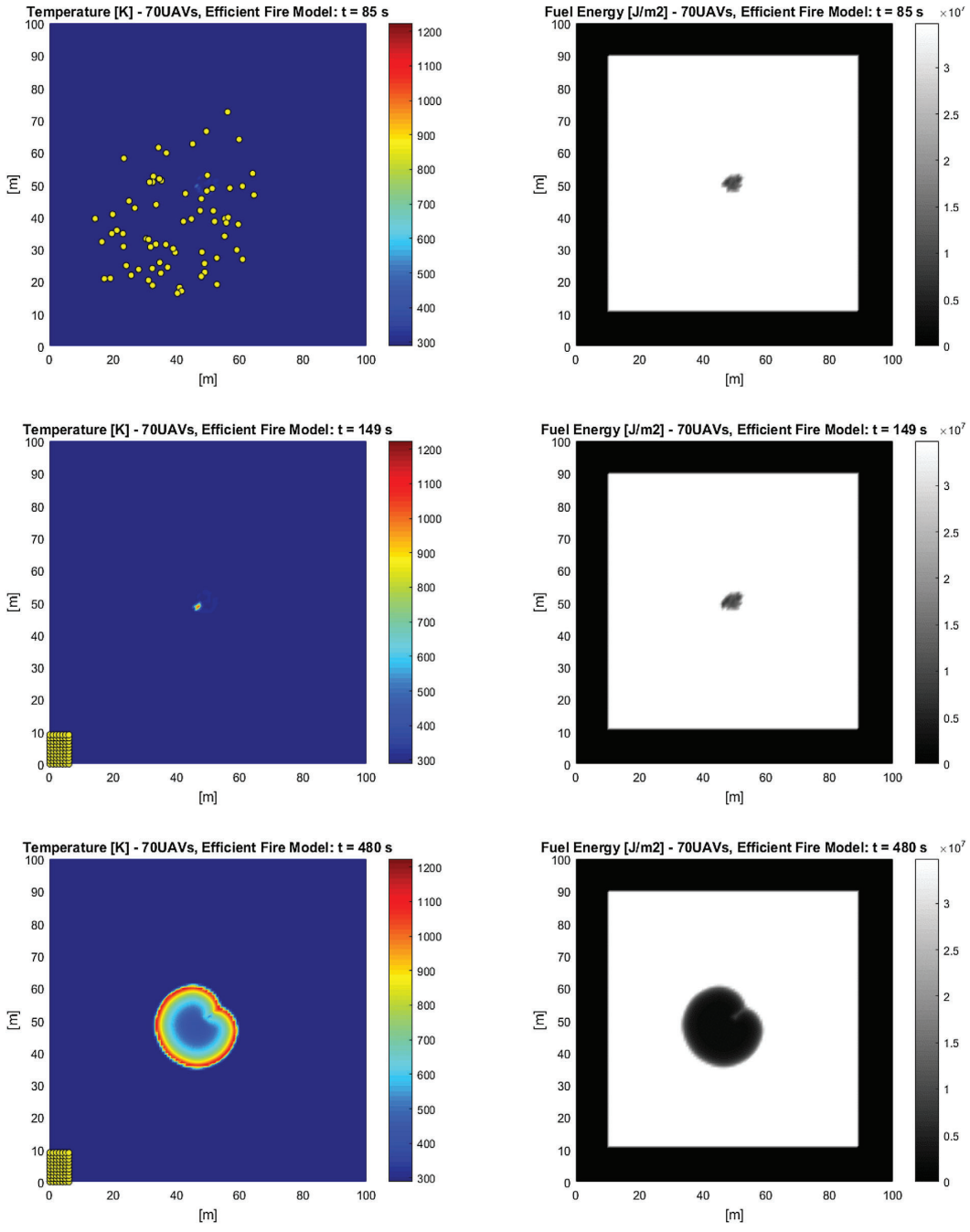


Figure 4.2: *Swarm of 70 drones fighting a single-source fire propagating for 480 s. The figures on the left show the temperature fields at t = 85 s (top), t = 149 s (middle) and t = 480 s (bottom). The figures on the right show the corresponding fuel energy fields. By t = 149 s, the fire is almost suppressed and the swarm returns to the docking stations leaving the fire to propagate unchecked [1].*

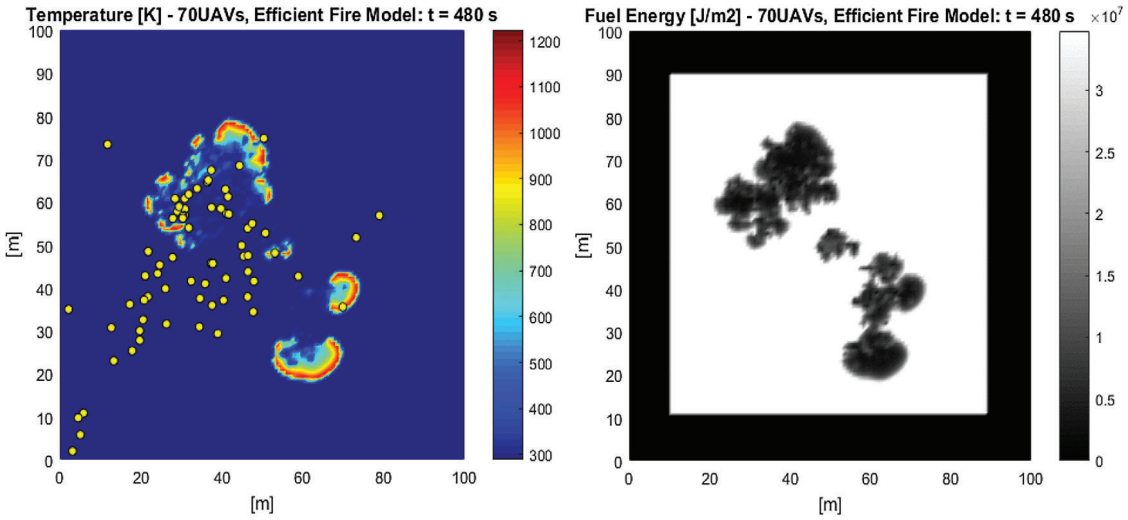


Figure 4.3: *Example of a run for a 70-drone swarm failing to suppress a four-source fire. The figure on the left shows the temperature field while the one on the right shows the fuel energy field at the end of the simulation (t = 480 s) [1].*

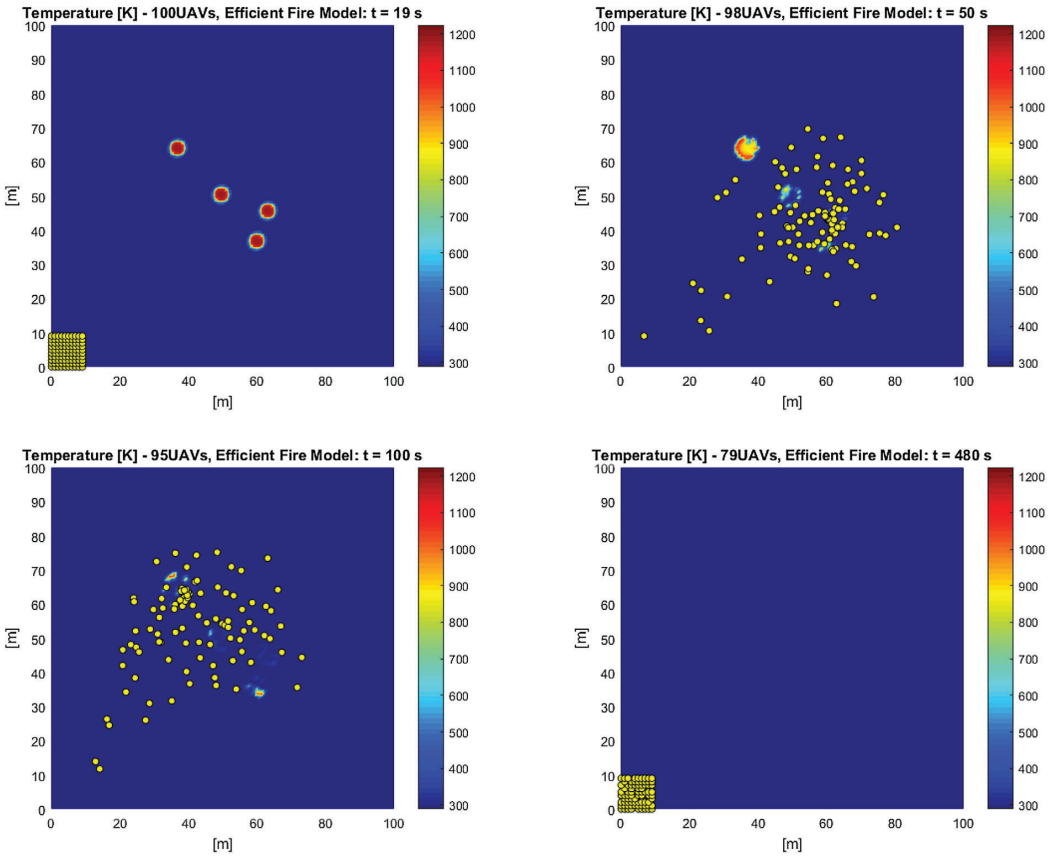


Figure 4.4: *Initial swarm of 100 drones ($\omega = 0.75$) fighting a fire originated from four sources and propagating for 480 s. Drone losses occur during firefighting operations at the constant rate of one unit every 15 s, with the first drone lost at $t = 30$ s. Drones are launched 20 s after the fire is ignited. The figure shows four snapshots of the temperature field at $t = 19$ s, $t = 50$ s, $t = 100$ s and $t = 480$ s. The fire is successfully suppressed, with 79 drones surviving the operations [1].*

Chapter 5

Collision-Avoidance Algorithm

The PSO-inspired self-organisation algorithm, presented in Chapter 4, forces the drones to converge on a single dynamically changing attractor (e.g. target position with the highest detected temperature), this results in a very high count of collisions, which is greatly undesirable. Therefore the development of a *collision-avoidance (c.a.) algorithm* is of utmost importance. Real-time multi-agent c.a. algorithms comprise a key enabling technology for the practical use of self-organising swarms of drones. This chapter proposes a decentralised reciprocal collision-avoidance algorithm based on stigmergy, which can be applied to swarms of any size. The algorithm is based on the gradient of the locally measured dynamic cumulative signal strength field emitted by the swarm, and therefore it is computationally efficient. The signal strength acts as a repulsor on each drone so that it steers away from the 'noisiest' regions (i.e. cluttered environments) therefore avoiding collisions. The magnitudes of these repulsive forces can be tuned to control the relative importance assigned to collision-avoidance with respect to the other phenomena affecting the drone dynamics. As expected, I found that the collision rate can be reduced either by decreasing the cruise speed of the drones and/or by increasing the sampling frequency of the global signal strength field. I carried out numerical experiments on a self-organising swarm of drones aimed at fighting wildfires autonomously. A beneficial side effect of the proposed collision-avoidance algorithm is that it helps maintain diversity in the swarm, thus enhancing exploration.

In this chapter, I propose a reactive decentralised stigmergy-based reciprocal collision-avoidance algorithm to be implemented on a self-organising swarm of drones. Its mathematical formulation can be found in section 5.1. The algorithm is tested on a two-dimensional autonomous firefighting system based on [102, 1] (i.e. with all drones flying at the same altitude) aimed at suppressing a simulated wildfire [7]. Formulation allows for a relatively seamless generalisation to three dimensions, which is beyond the scope of this chapter. The proposed multi-agent collision-avoidance algorithm was inspired by the E-field/potential field methods [47], the charged particles method [103], and the classical Reynold's rules [104, 105]. However, these methods are based on direct communication and inter-drone distance computations. Conversely, the proposed

algorithm is indirect via stigmergy using information that is continuously available and dynamically changing in the flying environment: the cumulative strength of a continuously broadcasted signal which is identically generated by each drone. Thus, the drone behavioural rules are modified by adding a repulsor that pushes the trajectory in the direction of the signal gradient. Numerical experiments are performed, in section 5.2, in order to investigate the effects of cruise speed and sampling frequency on the performance of the algorithm. Finally, conclusions and recommendations for future work are offered in section 5.3 followed by a brief summary of the chapter's topic, scope, method and findings in section 5.4.

5.1 Stigmergic collision avoidance

A stigmergic collision-avoidance algorithm is developed as an efficient and reliable alternative to other multi-agent algorithms proposed in the literature which require direct communication among drones. This approach is expected to be more reliable in real-world applications such as firefighting, since the required information is measured from and in turn modifies the environment, whilst direct drone-to-drone communications may be lost. Given the early stages of this research, comparisons against other methods are not carried out in this chapter. Instead, the aim is to demonstrate that the proposed method works, and to investigate how drone cruise speed and signal sampling time affect its performance.

Considering that the swarm-dynamics model I use as a reference (formulated in Chapter 4 and [1]) is based on the Particle Swarm Optimisation (PSO) method, I intend to embed the collision-avoidance algorithm within the drone trajectory difference equation in the form of a repulsor. Since each drone has no information regarding the location of other drones, indirect communication (stigmergy) comprises a highly desirable feature of the method: communication among drones is carried out by measuring and modifying the surrounding signal strength field in the environment. Thus, the process consists of two main stages: 1) sufficiently high-frequency simultaneous sampling of the signal intensity in a few points surrounding the drone to calculate the gradient, and 2) correction of the attractor by a convex combination of the collision-avoidance repulsor and the original attractor.

The original target formulation of each drone is regularly estimated at a predefined frequency as follows:

$$\begin{cases} \mathbf{x}_i^{(t+1)} = \mathbf{x}_i^{(t)} + \Delta\mathbf{x}_i^{(t)} \\ \Delta\mathbf{x}_i^{(t)} = \boldsymbol{\omega}_i^{(t)} \odot \left(\mathbf{x}_i^{(t)} - \mathbf{x}_i^{(t-1)}\right) + \boldsymbol{\phi}_i^{(t)} \odot \left(\mathbf{p}_i^{(t)} - \mathbf{x}_i^{(t)}\right) \\ \boldsymbol{\phi}_i^{(t)} = (\boldsymbol{\phi}_{\min} + \mathbf{k}_\phi \odot (\boldsymbol{\phi}_{\max} - \boldsymbol{\phi}_{\min})) \\ \boldsymbol{\phi}_{\max}, \boldsymbol{\phi}_{\min} = \mathbf{f}(\boldsymbol{\omega}_i^{(t)}) \end{cases} \tag{5.1}$$

where $\mathbf{x}_i^{(t)}$ is the ith drone position at time t, $\Delta\mathbf{x}_i^{(t)}$ is the intended position update for the following time step (capped by maximum cruise speed), $\mathbf{p}_i^{(t)}$ is the behavioural

attractor (function of the state of the drone), $\boldsymbol{\omega}_i^{(t)}$ represents the inertia of the i^{th} drone, and $\odot$ is the component-wise product. The bounds $\phi_{\min} = (\omega+1)$ and $\phi_{\max} = (\sqrt{\omega}-1)^2$ are defined in [106] and [1] to ensure oscillatory behaviour of the drones (beyond the scope of this chapter). The parameter $\mathbf{k}_\phi$ can change throughout the mission either deterministically or stochastically. In this chapter, I set $\mathbf{k}_\phi = (0.5\ 0.5)^{\mathrm{T}}$ to obtain a deterministic behaviour of the swarm thus facilitating the inference of the effects of other relevant parameters.

The attractor $\mathbf{p}_i^{(t)}$ for the i^{th} drone in (5.1) was originally a function of the best experiences of both the drone and the swarm (e.g. the highest measured temperature when fighting the wildfire) or the individual attractor for the specific drone's prioritised target (e.g. refuelling or water/suppressant collection). The attractor is modified by adding a repulsor associated to the signal strength field for collision avoidance purposes ($\mathbf{q}_i^{(t)}$ in (5.2)):

$$\begin{cases} \mathbf{p}_i^{(t)\star} = (1 - k_{\text{ca}}) \cdot \mathbf{p}_i^{(t)} + k_{\text{ca}} \cdot \mathbf{q}_i^{(t)} \\ 0 \leq k_{\text{ca}} < 1 \end{cases} \tag{5.2}$$

The repulsor $\mathbf{q}_i^{(t)}$ is function of the gradient of the signal strength field, σ, as follows:

$$\begin{cases} q_{ij}^{(t)} = x_{ij}^{(t)} - k_\sigma \cdot \frac{\partial \sigma(\mathbf{x}_i^{(t)})}{\partial x_j} \quad j = 1, 2 \\ \sigma(\mathbf{x}_i^{(t)}) = \Sigma_{m=1}^{N} \sigma_m(\mathbf{x}_i^{(t)}) \end{cases} \tag{5.3}$$

The individual signal intensity fields, σ_m, are normalised and are defined as usual power law with the distance from the m^{th} source:

$$\sigma_m(\mathbf{x}_i^{(t)}) = \begin{cases} r_{\text{ref}}^2 \cdot r_{im}^{-2} & r_{im} > r_{\text{ref}} \\ 1 & 0 \leq r_{im} \leq r_{\text{ref}} \end{cases}, \quad \text{where} \quad r_{im} = \left\| \mathbf{x}_i^{(t)} - \mathbf{x}_m^{(t)} \right\| \tag{5.4}$$

and r_{ref} is a reference radius of a circle delimiting the area of generation of the signal within each drone. Setting r_{ref} approximately the size of the drone or smaller ensures that the second condition in (5.4) should never occur. The proportionality constants in (5.2) and (5.3) depend on the magnitude of the other attractors in the trajectory equation (5.1), and therefore need to be tuned for the problem at hand. In this chapter, I set $k_{\text{ca}} = 0.7$ and $k_\sigma = 1000$. The gradient was calculated on a stencil of eight points surrounding the position of the drone on a structured grid with the same increment in both directions ($\Delta x = \Delta y$), as shown in (5.5).

$$\begin{cases} \frac{\partial \sigma}{\partial x_j} \simeq \frac{d\sigma}{dx_i} \\ \dfrac{d\sigma}{dx_1} = \dfrac{1}{6\Delta x} \cdot (\sigma_{\text{E}} - \sigma_{\text{W}} + 0.5 \cdot (\sigma_{\text{NE}} + \sigma_{\text{SE}} - \sigma_{\text{SW}} - \sigma_{\text{NW}})) \\ \dfrac{d\sigma}{dx_2} = \dfrac{1}{6\Delta y} \cdot (\sigma_{\text{N}} - \sigma_{\text{S}} + 0.5 \cdot (\sigma_{\text{NE}} + \sigma_{\text{NW}} - \sigma_{\text{SW}} - \sigma_{\text{SE}})) \end{cases} \tag{5.5}$$

5.2 Numerical experiments

The multi-agent collision-avoidance algorithm developed in this chapter is tested on a swarm of deterministically self-organising drones [1] fighting a simulated wildfire [7]. The collision-avoidance algorithm is always active whilst drones are airborne during any of the following mission phases for the simulated firefighting operations [1] in Figure 5.1:

- *Initialisation* The drones are positioned in their docking station with no water in their hull, the combustible vegetation fuel is distributed in the domain, and the wildfire is started in multiple ignition points (e.g. three in Fig. 5.1).
- *Firefighting* The drones search for fires by recording their best experience and sharing it with the entire swarm. They will pour a predefined amount of water when flying over their target position.
- *Water collection* When the water payload is less than 30% of the total storage volume, the drone's target is the water reservoir location (top left corner).
- *Refuelling* When the drones are about to finish the fuel, the refuelling area (bottom left corner) becomes their prioritised target.
- *Check* After all fires are supposed to be extinguished, the swarm will float around the domain searching for new hot spots.
- *Landing* The nearest (six) drones have priority to fly towards their original docking stations and land. When landed the collision-avoidance algorithm is deactivated and their position will not be considered any more for the collision count.
- *Hovering* The farthest drones will fly towards a waiting area, hovering until there is no more drones with priority over them for landing.
- *Landed* After every drone has landed, the simulation will continue for another minute to check that all fires are properly extinguished and that no re-ignition occurs.

While the first and the last phases are fixed, the other states are dynamically changing and can coexist: while some drones are moving towards their attractor, others may be flying to collect water or refuel.

Various numerical experiments have been performed with different combinations of maximum cruise speed (v) and sampling frequency of the signal field (f). For each experiment, the total number of collisions and the duration of the mission have been recorded. A few representative cases were collected and are provided in Table 5.1. It is important to note that every drone survives any collision in these numerical

ID	v [m·s^{-1}]	f [Hz]	C [-]	T [s]	$f \cdot v^{-1}$ [m^{-1}]	$C \cdot T^{-1}$ [s^{-1}]
1	5	30	0	551	6.0	0.00
2	10	30	7	264	3.0	0.03
3	15	30	244	182	2.0	1.34
4	20	30	1019	169	1.5	6.03
5	20	40	243	167	2.0	1.46
6	20	50	36	159	2.5	0.12
7	30	80	1	127	2.7	0.01

Table 5.1: *Experiments results at various combination of cruise speed, v, and sampling frequency, f. C is the total collision count, while T is the duration of the simulation.*

experiments. Although this may sound unrealistic, this is for statistical purposes: the rate of collisions would decrease due to a shrinking swarm rather than due to a successful collision-avoidance algorithm.

Experiments 1 to 4 in Table 5.1 were carried out at the same sampling frequency (30 Hz), clearly showing that increasing cruise speed (5, 10 and 15 m/s) exponentially deteriorates the performance of the proposed collision-avoidance algorithm. In turn, experiments 4 to 6 in Table 5.1 were carried out at the same cruise speed (20 m/s), clearly showing that increasing sampling frequency (30, 40 and 50 Hz) dramatically decreases the collision rate. Whilst the available computational resources allowed us to carry out experiments up to a sampling frequency of 80 Hz, I expect that a sampling frequency of 100 Hz would suffice to bring the collision rate to zero in this particular case. Further experiments will be performed in the near future to support this claim.

By studying the values of the frequency-to-velocity and collisions-to-time ratios (last two columns in Table 5.1), one can identify a hyperbolic trend - i.e. for small frequency-to-velocity values the collisions-to-time ratio tends to infinity, while for higher values of the frequency-to-velocity ratio the collisions-to-time count tends to zero. The fact there might exist just one function describing the relation between these two coefficients is suggested by the experiments 3 and 5 performed at 30 Hz and 40 Hz, respectively: in these cases approximately the same ratios are obtained, so they could lie on the same curve. Incidentally, the ratio $f \cdot v^{-1}$ provides a trade-off between the two requirements expected to reduce the collision rate: increasing the sampling frequency, f; or decreasing the cruise speed, v. Further experiments will be carried out in order to better investigate this claim.

Additionally, I observed a beneficial side effect of the proposed collision-avoidance algorithm: it helps maintaining diversity in the swarm, thus enhancing exploration.

5.3 Conclusion

I proposed and developed a reactive multi-agent decentralised stigmergy-based algorithm for reciprocal collision-avoidance. Its potential was demonstrated by implementing it on a swarm of drones self-organised to fight the propagation of simulated wildfires. Arguably, the algorithm could also be used for blind flight –i.e. when the drones have no means of knowing the positions of others. The effects of two main parameters were investigated, namely the cruise speed and the sampling frequency, reaching the expected conclusion that decreasing the former and/or increasing the latter results in the reduction of the collision rate. Furthermore, by analysing the frequency-to-speed and collisions-to-time ratios, I infer that there might be a trend that should be further investigated.

Purposely, no uncertainty in the environment, no stochasticity in the algorithms, and no flight dynamics were considered in order to focus on the impact of the stigmergy-based collision-avoidance algorithm on the collective behaviour and performance of the swarm. Future work will introduce these aspects as well as extend the developments to three-dimensional space.

5.4 Summary

Scope: Propose a collision-avoidance algorithm for a self-organising swarm of drones. This would solve the issue of drones converging on the same attractor, and therefore colliding, in the PSO-inspired firefighting algorithm presented in Chapter 4.
The collision-avoidance algorithm ought to be:

- *decentralised*: it should not need an overarching omniscient controller, it should rely on the individual decisions of each drone;
- *scalable*: can be applied to swarms of any size;
- *cluttered swarm*: chaotic movements of a multitude of drones close together should not be an issue;
- *no line of sight*: the drones should detect the imminent collision even if direct sight is blear or impossible - e.g. in case of smoke or obstruction by vegetation.

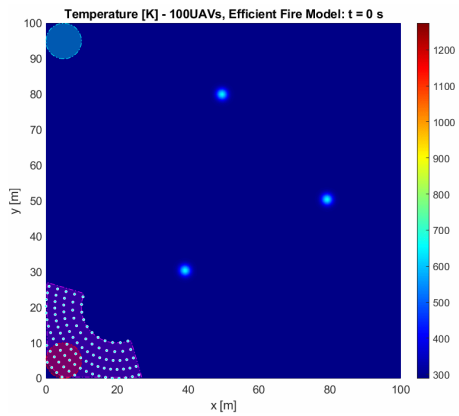
Method, studies, and experiments:

- *Mathematical formulation of the collision algorithm*: the drones' self-organisation rules are modified by adding a *repulsor* that pushes the trajectory in the direction of the signal gradient.
- *Simulation* by using the same system presented in Chapter 4 and adding the collision avoidance.

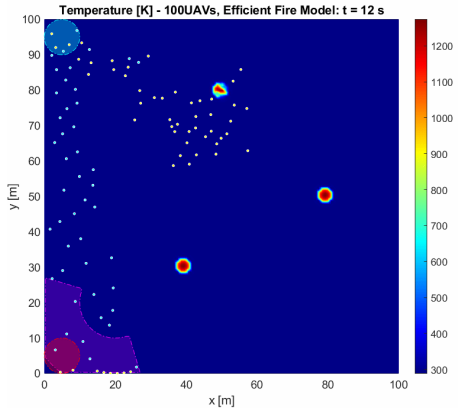
- *Testing* the algorithm against different combinations of cruise speed and sampling frequency of the signal field.

Main outcomes and findings:

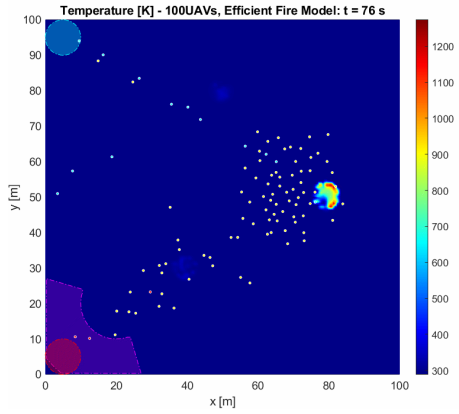
- A reactive multi-agent decentralised stigmergy-based algorithm for reciprocal *collision-avoidance*. It consists in adding a repulsor in the modified attractor for the PSO-inspired swarm self-organisation presented in Chapter 4. The repulsor is function of the spatial and temporal derivatives of the cumulative signal strength originated by each drone in the swarm.
- Two main parameters regulate the success of the algorithm: decreasing the *cruise speed* or increasing the *sampling frequency* result in decreasing the collisions count. This suggests a critical ratio value between the two that assures no collisions occur.
- The developed c.a. algorithm has the potential to be used for *blind flight* - i.e. drones are not aware of other drones positions.
- The following topics have been intentionally overlooked at this stage and should be considered for further studies: the *uncertainty in the environment*, the *stochasticity in the algorithms*, and *flight dynamics*.



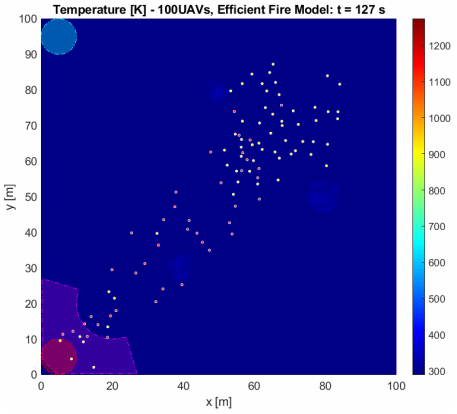
Initial condition



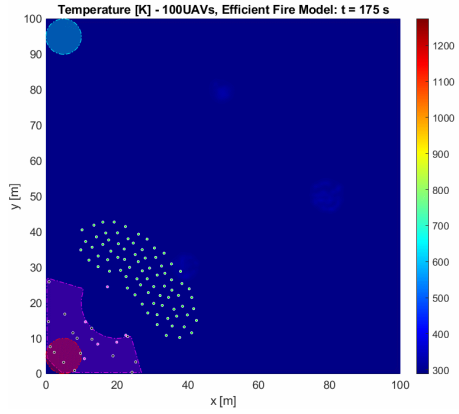
Water collection (ciano)



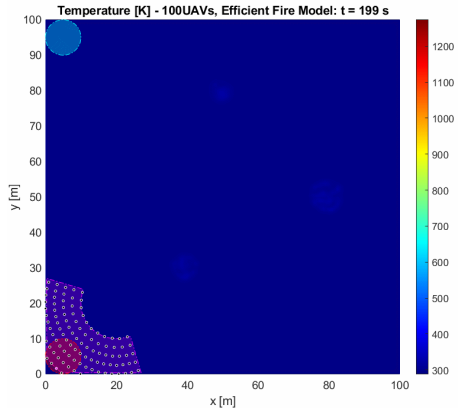
Refuelling (red)



Fire extinction



Hovering (green) & landing (magenta)



Landed (black) on docking station

Figure 5.1: *Top views of a 100m×100m homogeneous distribution of combustible vegetation ignited in three locations (top-left figure) and some representative instances for the main mission phases of a swarm of 100 firefighting drones. The colour scale represents the temperature field in Kelvin, the darkest blue is ambient temperature, while red is high flaming temperature. At the bottom-left corner, the red circle is the refuelling area, while the arena polygon represents the collection of all the landing points of the drones. The blue circle in the top-left corner is the water source used for suppression.*

Chapter 6

Swarm-in-the-Loop: the Indoor Positioning System

This chapter presents some theoretical and experimental research contributions to the development of a Swarm-in-the-Loop system for testing self-organisation algorithms for swarms. Specifically, the interest is in developing a platform to test the fire suppression system previously presented. Here the focus is on the Indoor Positioning System (IPS), which provides an estimation of the positions of each drones in the swarm. Positioning systems comprise a crucial technology in robotics and autonomous systems. In particular, IPSs are used where satellite and other outdoor positioning technology lack precision or fail. Ultra-WideBand (UWB) technology was developed for high data transfer rates over short distances and at low power densities. Therefore it is suitable for IPSs, although signals tend to be disrupted by various objects. This chapter presents a comprehensive study of the precision, accuracy and failure of IPSs based on UWB technology and pseudorange multilateration algorithm using signal Time Difference of Arrival (TDoA). These IPSs consist of a network of more than three transmitting anchors and a single receiving object whose position is to be estimated. Thus, a theoretical study of the precision of the position estimates is performed, which consists of a Cramér–Rao Lower Bound analysis for the case of round-robin scheduling, and of an anisotropic representation of the signal-to-noise ratio function of the reconstructed three-dimensional radiation pattern of the antennae being considered. In addition, a geometrical study of the two-dimensional positioning domain is carried out, thereby defining bifurcation envelopes which bound the areas where the IPS is predicted to fail. Finally, a debiasing filter that relies on the statistics of experimental data is developed to improve the accuracy of the position estimates. Experimental results with four anchors demonstrate the correctness of the theoretical predictions of precision and failure, and the accuracy improvement achieved by the debiasing filter. If the region of interest is the convex hull of the four anchors, the IPS is observed to fail near them, whilst precision is found to be of about ± 3 cm. The average accuracy improvement achieved by the debiasing filter is of about 15 cm for static and 5 cm for dynamic measurements.

This chapter is organised as follows: in Section 6.1, the properties of the IPS under study are specified; Section 6.2 presents the main contributions to the precision, accuracy, and failure analysis of UWB-based IPSs that use multilateration algorithm and signal TDoA; Section 6.3 provides an overview of the filtering process proposed to improve accuracy; the validation and testing experiments are designed in Section 6.4, with results being presented and discussed in Section 6.5. Finally, conclusions and recommendations for future work are offered in section 6.6 followed by a brief summary of the chapter's topic, scope, method and findings in section 6.7.

6.1 Layout of the IPS under study

The purpose of the IPS is to localise a moving object, which behaves as a receiver, based on the spatial distribution of the anchors (transceivers). The designed experimental setup will be used to perform the experiments described in Sections 6.4 and 6.5.

The reference IPS, depicted in Figure 6.1, consists of a drone to be localised and four transceiver anchors positioned at the vertices and facing the centre of a square domain. All the antennae are at a height of 20 cm from the floor. The drone is not flying but sliding parallel to the floor at the same height on a moving stand equipped with a laser pointer, which is aligned with the onboard UWB antenna in order to achieve reference positioning of high precision ($\pm$1mm) and accuracy. The regularly spaced markers on the floor are the sampling positions to be used for mapping in Section 6.4.1.

6.2 Precision and failure analyses

A CRLB analysis specific for TDoA with *round-robin scheduling* is performed to study the **precision** of the system, whilst a *bifurcation envelope* is defined to bound the areas where it is expected to **fail**. Note that the object to be localised is a passive receiver here whereas it is often a transceiver in the literature. This is in line with the Loco Positioning system installed in our laboratory (AVAILab, *availab.org*). Nonetheless, the theoretical results are applicable to both cases, as long as receivers are sensitive and approximately omnidirectional.

6.2.1 Reference CRLB analysis

Before proceeding with the analysis, it is important to refresh some reference theory to set the background nomenclature. Considering an indoor positioning system consisting of only three anchors and using a Time-Difference-of-Arrival (TDoA) multilateration, the *pseudoranges* can be defined as the range differences between the node and each anchor:

$$\begin{aligned} \text{pseudorange:} \quad & \tau_{ij}(\mathbf{x}) = d_{\mathrm{j}}(\mathbf{x}) - d_{\mathrm{i}}(\mathbf{x}) \\ \text{range:} \quad & d_{\mathrm{i}}(\mathbf{x}) = \|\mathbf{x} - \mathbf{x}_i\| \text{ and } i, j = 1, 2, 3 \end{aligned} \tag{6.1}$$

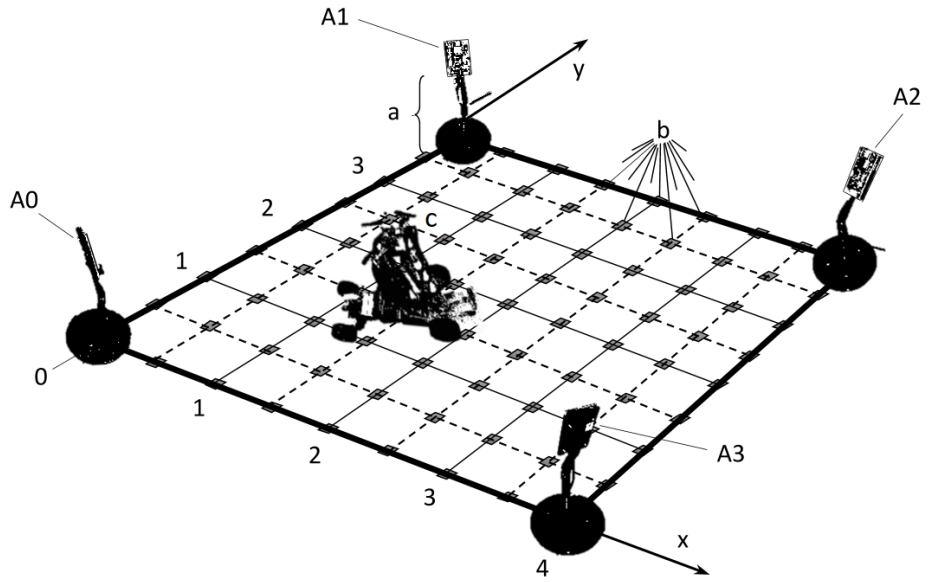


Figure 6.1: *Diagram of the setup of the studied 4×4 m^2 IPS for 2D localisation: (a) adjustable stands for transmitting anchors antennae (A0-A3), (b) measurement points regularly distributed every 50 cm in both directions, and (c) mobile stand for the object to be localised. The system's components are not in scale for better visualisation.*

where d_i is the distance between the drone ($\mathbf{x}$) and the ith anchor position ($\mathbf{x}_i$). According to [84], the speed of propagation in the medium is considered to be 1 without loss of generality since the range is linearly dependent on the Time-of-Arrival (ToA). The τ_{ij} can be assembled together by defining a TDoA mapping that transforms from the two-dimensional space of source location to a space of pseudoranges called τ-plane, as suggested in [107]:

$$\begin{array}{rccc} \boldsymbol{\tau_2}: & \mathbb{R}^2 & \rightarrow & \mathbb{R}^2 \\ & \mathbf{x} & \rightarrow & (\tau_{12}(\mathbf{x}), \tau_{13}(\mathbf{x})) \end{array} \tag{6.2}$$

Studying the TDoA map is crucial for the mathematical characterisation of the localisation problem at hand. Considering an IPS that consists of a network of N anchors and one node - e.g. a drone -, M measurements (number that depends on the localisation algorithm) are performed every time step, accordingly to the frequency at which messages are sent from anchors to node. Every measurement is modeled as a normal distribution which is a function of both the real measurements and an additive Gaussian noise. The standard deviation of the noise changes in space with the distance from any transmitting anchor. The collection of M measured pseudoranges, $\hat{\boldsymbol{\tau}}$, at time step k can be expressed as:

$$\begin{gathered}\hat{\boldsymbol{\tau}}^{(k)} \in \mathbb{R}^M, \\ \hat{\tau}_{ij} \sim \boldsymbol{\mathcal{N}}\left(\tau_{ij}(x), \bar{\sigma}_{ij}^2\right), \quad \bar{\sigma}_{ij} = f(\sigma_i, \sigma_j), \\ \tau_{ij} = \|\mathbf{x} - \mathbf{x}_i\| - \|\mathbf{x} - \mathbf{x}_j\|, \\ i, j \in \{1, \ldots, N\} \quad \text{with} \quad i \neq j,\end{gathered} \tag{6.3}$$

where $\hat{\tau}_{ij}$ is the individual pseudorange measurement, considering the two ToAs to the node from the ith and jth anchors, and τ_{ij} is the real range difference. The superscript $^{(k)}$ will be omitted, but implicitly inferred, in further analyses. Please note that $\hat{\boldsymbol{\tau}}$ is a column vector, not a matrix. The $\hat{\tau}_{ij}$ components can be as many as the binomial coefficient $\binom{N}{3} = \frac{N!}{3!(N-3)!}$. The functions evaluating the combined standard deviations, , will be presented for the specific studied IPS in section 6.2.

As early stated, the well-known Cramér–Rao Lower Bound (CRLB) analysis comes very handy in evaluating the precision of an unbiased IPS. Such analysis is based on the concept of the Fisher Information Matrix (FIM), which contains the likelihood to obtain a correct measurement. The elements of the total FIM for the general positioning problem according to [108, 109] are:

$$\begin{aligned}\mathrm{FIM}_{ij} = &\left(\frac{\partial \boldsymbol{\tau}(\mathbf{x})}{\partial x_i}\right)^{\mathrm{T}} \mathbf{F}_\tau^{-1}(\mathbf{x}) \left(\frac{\partial \boldsymbol{\tau}(\mathbf{x})}{\partial x_j}\right) \\ &+ \frac{1}{2} tr\left(\mathbf{F}_\tau^{-1}(\mathbf{x}) \frac{\partial \mathbf{F}_\tau(\mathbf{x})}{\partial x_i} \mathbf{F}_\tau^{-1}(\mathbf{x}) \frac{\partial \mathbf{F}_\tau(\mathbf{x})}{\partial x_j}\right)\end{aligned} \tag{6.4}$$

where $\mathbf{F}_\tau$ is the covariance matrix of the $\hat{\boldsymbol{\tau}}$ measurements, and $tr(\mathrm{M})$ is the trace of a matrix M. Since each standard deviation is considered to be changing in space – i.e. $\sigma_i(x, y)$ – the correction term (second row in (6.4)) is acknowledged in the following analysis. The likelihood function, $\mathcal{L}$, that describes the relative odds of obtaining the observed data $\mathbf{h}$ for all permissible values of the parameter $\mathbf{x}$ for a single measurement h is:

$$\mathcal{L}(\hat{h}|\mathbf{x}) = \frac{1}{\sqrt{2\pi}\sigma(\mathbf{x})} e^{\left(-\frac{1}{2\sigma^2(\mathbf{x})}\left(\hat{h} - h(\mathbf{x})\right)^2\right)} \tag{6.5}$$

In the positioning problem at hand, $h(\mathbf{x})$ is the range, or similarly the ToA, and the parameter $\mathbf{x}$ is the node position. The standard deviation of such distribution is again σ. Considering what Fisher Information Matrix (FIM) is in terms of likelihood [108, 109], the logarithmic likelihood has to be considered:

$$\ell(\hat{h}|\mathbf{x}) = \ln\left(\frac{1}{\sqrt{2\pi}\sigma(\mathbf{x})}\right) - \frac{1}{2}\frac{\left(\hat{h} - h(\mathbf{x})\right)^2}{\sigma^2(\mathbf{x})} \tag{6.6}$$

Hence, the total FIM for the location, considering that also each standard deviation changes in space ($\sigma_i = \xi(x, y)$) is the following:

$$\mathrm{FIM}_{ij} = \left(\frac{\partial \mathbf{h}(\mathbf{x})}{\partial x_i}\right)^{\mathrm{T}} \mathbf{F}_\tau^{-1}(\mathbf{x}) \left(\frac{\partial \mathbf{h}(\mathbf{x})}{\partial x_j}\right) + \frac{1}{2} tr\left(\mathbf{F}_\tau^{-1}(\mathbf{x}) \frac{\partial \mathbf{F}_\tau(\mathbf{x})}{\partial x_i} \mathbf{F}_\tau^{-1}(\mathbf{x}) \frac{\partial \mathbf{F}_\tau(\mathbf{x})}{\partial x_j}\right) \tag{6.7}$$

where $tr(M)$ is the trace of the matrix M.

The $\mathbf{F}_\tau$ is the information matrix of the selected TDoA measurements set. Let's suppose that the TDoA protocol requires M measurements, therefore $\mathbf{F}_\tau$ is:

$$\mathbf{F}_\tau = [F_{ij}]_{M\times M} \tag{6.8}$$

Using an efficient unbiased estimator it is proven [108] that $\mathbf{F}_\tau$ is the measurement covariance matrix:

$$\mathbf{F}_\tau = \left[E\left[(\hat{\tau}_{ij} - \bar{\tau}_{ij})(\hat{\tau}_{kp} - \bar{\tau}_{kp})\right]\right]_{M\times M} \tag{6.9}$$

where the indices i, j, k, p depends on the selected scheduling.

As an example, if all the TDoA measurements where performed keeping always the anchor 1 as reference (all the studies found in literature), all the τ_{1i} would be correlated with the standard deviation in 1 (σ_1^2). Therefore, the resulting FIM for the measurement set $\boldsymbol{\tau}_1 = \{\tau_{12}, \tau_{13}, \tau_{14}\}$ is:

$$\mathbf{F}_\tau = \begin{bmatrix} s_1 + s_2 & s_1 & s_1 \\ s_1 & s_1 + s_3 & s_1 \\ s_1 & s_1 & s_1 + s_4 \end{bmatrix} \tag{6.10}$$

with $s_i = \sigma_i^2$.

6.2.2 CRLB analysis for pseudorange multilateration with round-robin scheduling

The Cramér–Rao Lower Bound (CRLB) analysis is generally deemed suitable for evaluating the precision of an unbiased IPS. It is based on the concept of the Fisher Information Matrix (FIM) involving the likelihood of obtaining a correct measurement. For details on the theory and terminology, refer to Appendix 6.2.1. The elements of the total FIM for the general positioning problem [108, 109] are as shown in (6.11):

$$\mathrm{FIM}_{ij} = \left(\frac{\partial \mathbf{h}(\mathbf{x})}{\partial x_i}\right)^{\mathrm{T}} \mathbf{F}_\tau^{-1}(\mathbf{x}) \left(\frac{\partial \mathbf{h}(\mathbf{x})}{\partial x_j}\right) + \frac{1}{2} tr\left(\mathbf{F}_\tau^{-1}(\mathbf{x}) \frac{\partial \mathbf{F}_\tau(\mathbf{x})}{\partial x_i} \mathbf{F}_\tau^{-1}(\mathbf{x}) \frac{\partial \mathbf{F}_\tau(\mathbf{x})}{\partial x_j}\right) \tag{6.11}$$

where $\mathbf{h}(\mathbf{x})$ is the range vector (e.g. distance between receiver and anchors), $\mathbf{F}_\tau$ is the covariance matrix of the $\hat{\boldsymbol{\tau}}$ measurements, and $tr(\cdot)$ is the trace function. Eq. (6.11) considers that the standard deviations (σ_i) of the likelihood function (and hence $\mathbf{F}_\tau$) vary in space.

One column of the Jacobian matrix of $\mathbf{h}$ is defined as in (6.12).

$$\frac{\partial \mathbf{h}(\mathbf{x})}{\partial x_i} = \begin{bmatrix} \frac{\partial h_{12}(\mathbf{x})}{\partial x_i} \\ \frac{\partial h_{23}(\mathbf{x})}{\partial x_i} \\ \frac{\partial h_{34}(\mathbf{x})}{\partial x_i} \\ \frac{\partial h_{41}(\mathbf{x})}{\partial x_i} \end{bmatrix} \tag{6.12}$$

TDoA measurements for N anchors and round-robin scheduling are referred to as $\boldsymbol{\tau}_{rr} = \{\tau_{12}, \tau_{23}, ..., \tau_{N1}\}$. Thus, the divergence matrix of $\mathbf{h}$ for $\boldsymbol{\tau}_{rr} = \{\tau_{12}, \tau_{23}, \tau_{34}, \tau_{41}\}$ for the TDoA2 protocol used by `bitcraze` [110, 111] is as in (6.13).

$$\frac{\partial \mathbf{h}(\mathbf{x})}{\partial \mathbf{x}} = \begin{bmatrix} \frac{x-x_1}{\|\mathbf{x}-\mathbf{x}_1\|} - \frac{x-x_2}{\|\mathbf{x}-\mathbf{x}_2\|} & \frac{y-y_1}{\|\mathbf{x}-\mathbf{x}_1\|} - \frac{y-y_2}{\|\mathbf{x}-\mathbf{x}_2\|} \\ \frac{x-x_2}{\|\mathbf{x}-\mathbf{x}_2\|} - \frac{x-x_3}{\|\mathbf{x}-\mathbf{x}_3\|} & \frac{y-y_1}{\|\mathbf{x}-\mathbf{x}_1\|} - \frac{y-y_2}{\|\mathbf{x}-\mathbf{x}_2\|} \\ \frac{x-x_3}{\|\mathbf{x}-\mathbf{x}_3\|} - \frac{x-x_4}{\|\mathbf{x}-\mathbf{x}_4\|} & \frac{y-y_3}{\|\mathbf{x}-\mathbf{x}_3\|} - \frac{y-y_4}{\|\mathbf{x}-\mathbf{x}_4\|} \\ \frac{x-x_4}{\|\mathbf{x}-\mathbf{x}_4\|} - \frac{x-x_1}{\|\mathbf{x}-\mathbf{x}_1\|} & \frac{y-y_4}{\|\mathbf{x}-\mathbf{x}_4\|} - \frac{y-y_1}{\|\mathbf{x}-\mathbf{x}_1\|} \end{bmatrix}_{4\times 2} \tag{6.13}$$

Making use of the linear properties of the expected value, I can calculate the diagonal elements of $\mathbf{F}_\tau$ as in (6.14) and its connected elements for consecutive estimators as in (6.15), where only the j^{th} anchor is in common, the hat identifies measurements, the bar stands for the mean, and $E[\cdot]$ stands for expectation.

$$\begin{aligned} F_{ij,ij} = E\left[(\hat{\tau}_{ij} - \bar{\tau}_{ij})^2\right] = E\left[((\hat{\tau}_i - \bar{\tau}_i) - (\hat{\tau}_j - \bar{\tau}_j))^2\right] = \\ E\left[(\hat{\tau}_i - \bar{\tau}_i)^2\right] + E\left[(\hat{\tau}_j - \bar{\tau}_j)^2\right] - 2\cancel{E\left[(\hat{\tau}_i - \bar{\tau}_i)(\hat{\tau}_j - \bar{\tau}_j)\right]} = \sigma_i^2 + \sigma_j^2 \end{aligned} \tag{6.14}$$

$$\begin{aligned} F_{ij,jk} = E\left[(\hat{\tau}_{ij} - \bar{\tau}_{ij})(\hat{\tau}_{jk} - \bar{\tau}_{jk})\right] = \\ E\left[((\hat{\tau}_i - \bar{\tau}_i) - (\hat{\tau}_j - \bar{\tau}_j))((\hat{\tau}_j - \bar{\tau}_j) - (\hat{\tau}_k - \bar{\tau}_k))\right] = \\ E\left[-(\hat{\tau}_j - \bar{\tau}_j)^2\right] = -E\left[(\hat{\tau}_j - \bar{\tau}_j)^2\right] = -\sigma_j^2 \end{aligned} \tag{6.15}$$

Estimating the covariance between seemingly uncorrelated TDoA measurements $(\hat{\tau}_{ij},\ \hat{\tau}_{kp})$ is not trivial. Deriving from the Cauchy-Bunyakovsky-Schwarz inequality,

$$\begin{aligned} \left(E[\Delta\hat{\tau}_{ij} \cdot \Delta\hat{\tau}_{kp}]\right)^2 \leqslant E[\Delta\hat{\tau}_{ij}^2] \cdot E[\Delta\hat{\tau}_{kp}^2] \implies \\ -\sqrt{E[\Delta\hat{\tau}_{ij}^2] \cdot E[\Delta\hat{\tau}_{kp}^2]} \leqslant E[\Delta\hat{\tau}_{ij} \cdot \Delta\hat{\tau}_{kp}] \leqslant \sqrt{E[\Delta\hat{\tau}_{ij}^2] \cdot E[\Delta\hat{\tau}_{kp}^2]}. \end{aligned} \tag{6.16}$$

where $\Delta\hat{\tau}_{ij} = (\hat{\tau}_{ij} - \bar{\tau}_{ij})$. From (6.15) and (6.16), the covariance between $\hat{\tau}_{ij}$ and $\hat{\tau}_{kp}$ can be bounded as shown in (6.17):

$$\begin{aligned} 0 > F_{ij,kp} = E\left[(\hat{\tau}_{ij} - \bar{\tau}_{ij})(\hat{\tau}_{kp} - \bar{\tau}_{kp})\right] \geq -\sqrt{E\left[(\hat{\tau}_{ij} - \bar{\tau}_{ij})^2\right] \cdot E\left[(\hat{\tau}_{kp} - \bar{\tau}_{kp})^2\right]} = \\ -\sqrt{(\sigma_i^2 + \sigma_j^2)(\sigma_k^2 + \sigma_p^2)} \end{aligned} \tag{6.17}$$

Thus, the information matrix of the TDoA measurements set $\boldsymbol{\tau}_{rr} = \{\tau_{12}, \tau_{23}, \tau_{34}, \tau_{41}\}$ for four coplanar anchors using an efficient unbiased estimator is given by the measurement covariance matrix in (6.18), where s_i stands for σ_i^2.

$$\begin{aligned} \mathbf{F}_\tau = \begin{bmatrix} s_1 + s_2 & -s_2 & F_{12,34} & -s_1 \\ -s_2 & s_2 + s_3 & -s_2 & F_{23,41} \\ F_{12,34} & -s_3 & s_3 + s_4 & -s_4 \\ -s_1 & F_{23,41} & -s_2 & s_4 + s_1 \end{bmatrix}_{4\times 4} \\ F_{12,34} = -\sqrt{(s_1 + s_2)(s_3 + s_4)} \\ F_{23,41} = -\sqrt{(s_2 + s_3)(s_4 + s_1)} \end{aligned} \tag{6.18}$$

Kaune et al. [109] suggest that the variance for a specific source is as shown in (6.19).

$$\sigma_i^2(r) = \begin{cases} \frac{a}{\mathrm{SNR}_0} \cdot \frac{r_i^2}{r_0^2} & \text{if} \quad r_i \geq r_0 \\ \frac{a}{\mathrm{SNR}_0} & \text{if} \quad r_i < r_0 \end{cases} \\ \text{with} \quad a = \frac{c^2}{B^2} \tag{6.19}$$

where SNR_0 is the *signal-to-noise* power ratio at threshold distance r_0 from the i^{th} anchor considered; c is the signal propagation speed; and B is the bandwidth of the received signal. The SNR_0 varies with the view angle θ if the antenna has some directionality. In order to evaluate the $\mathrm{SNR}(\mathbf{x})$, I use the *Friis formula* for noise, which would give the relation between signal gain (over the noise) and distance between transmitter and receiver for different channel frequencies.

6.2.3 Signal-to-Noise-Ratio formulation

The SNR in (6.20) is the ratio between the power of the signal reaching the receiver (P_r) and the noise power (P_N):

$$\mathrm{SNR}\left(d, \theta_\mathrm{t}, \phi_\mathrm{t}, f_\mathrm{ref}, T, P_\mathrm{t}\right) = \frac{P_\mathrm{r}}{P_\mathrm{N}} \tag{6.20}$$

It can be written as a function of the distance (d) between transmitter and receiver, the representative transmission frequency (f_ref) and bandwidth of the selected channel, the temperature of the environment (T), the transmitting power (P_t) and the gains of the transmitting antenna (G_t) and the receiving antenna (G_r). Since the receiving antenna is usually very sensitive, G_r can be neglected in this analysis. The gain G_t can be a function of the azimuth (θ_t) and elevation (ϕ_t) angles with respect to the frame of reference centred on the antenna. The power at the end of the transmission line can be expressed using the contemporary Friis law, as shown in (6.21):

$$P_\mathrm{r} = \frac{P_\mathrm{t} \cdot G_\mathrm{t} \cdot G_\mathrm{r}}{L_\mathrm{t} \cdot L_\mathrm{r}} \cdot \left(\frac{c}{4\pi \cdot f_\mathrm{ref} \cdot d}\right)^2 \tag{6.21}$$

where L_t and L_r are the electric losses in the electronics of the transmitter and receiver modules, respectively, which have been embedded in the gains G_t and G_r. It is convenient to express everything in logarithmic form. Combining (6.21) and (6.19), the upper bound of the standard deviation is obtained as in (6.22), where dBm stands for dB milliwatts. Note that the noise power is expanded into a thermal noise power term, $\mathrm{k_B}TB_\mathrm{w}$, where $\mathrm{k_B}$ is the Boltzmann constant for radiation of a black body ($\approx 1.38 \times 10^{-23}\mathrm{J/K}$).

$$\begin{cases} \mathrm{SNR_{dB}} = P_\mathrm{tdBm}(T, V_i) + G_\mathrm{tdBi}(\theta_\mathrm{t}, \phi_\mathrm{t}) - 10\log_{10}(\mathrm{k_B}TB_\mathrm{w}10^3) - 20\log_{10}\left(4\pi f_\mathrm{ref} d/c\right) \\ \sigma^2 = \frac{c^2}{B_\mathrm{w}^2} \cdot 10^{-\mathrm{SNR_{dB}}/10} \end{cases} \tag{6.22}$$

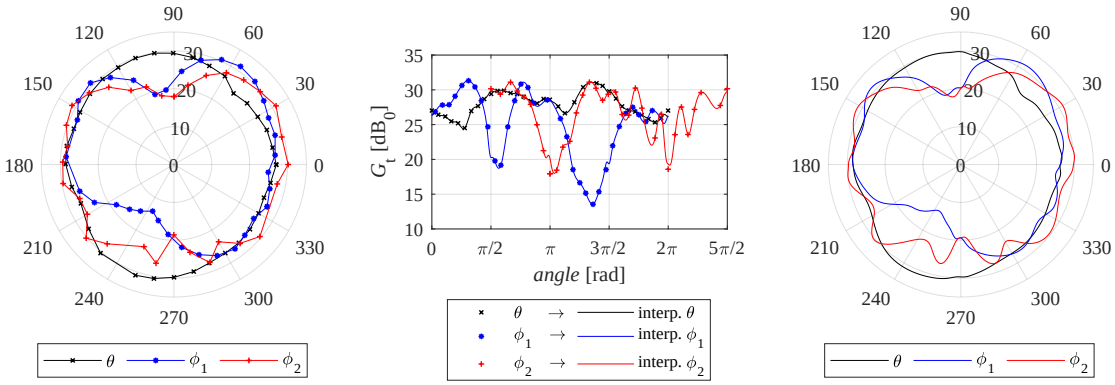
where $P_{\text{tdBm}}(T, V_i)$ is an experimental curve approximating the relationship between the transmission power, the ambient temperature, and the input voltage (V_i) as in (6.23) [112], and $G_{\text{tdBi}}(\theta_{\text{t}}, \phi_{\text{t}})$ is the measured transmitting antenna gain (with respect to an isotropic antenna), which is a three-dimensional (3D) radiation pattern function of the azimuth and elevation angles [113].

$$P_{\text{tdBm}}(T, V_i) = P_{\text{t0dBm}} + \left.\frac{\partial P_{\text{t}}}{\partial T}\right|_{T_{\text{ref}}} (T - T_{\text{ref}}) + \left.\frac{\partial P_{\text{t}}}{\partial V}\right|_{V_{\text{ref}}} (V_i - V_{\text{ref}}) \tag{6.23}$$

6.2.4 Radiation pattern of the DW1000 antenna

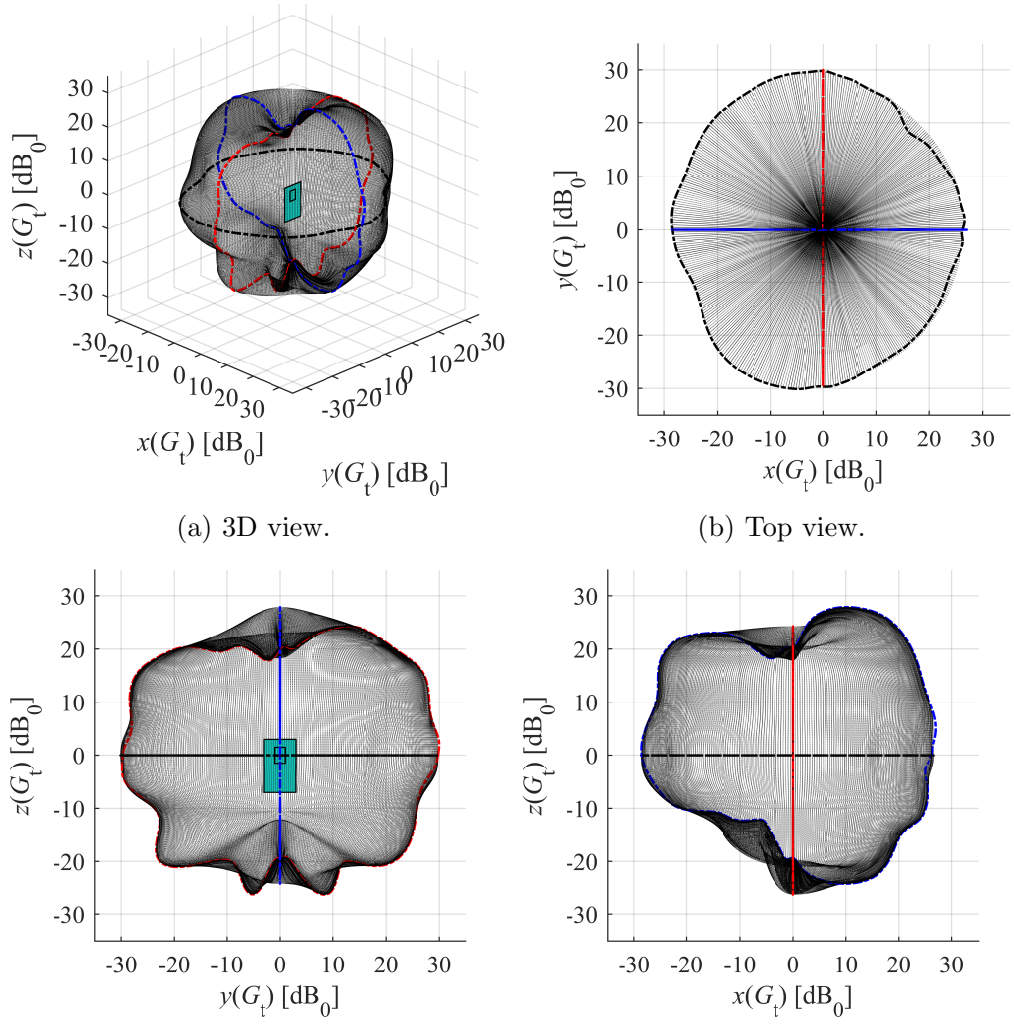
In order to reconstruct the 3D radiation pattern from the three measured sections in azimuth (θ), elevation-1 (ϕ_1) and elevation-2 (ϕ_2) planes (see Figure 6.2), I formulate a linear combination of the boundary values of the considered quadrant. Using the system of equations in (6.24), the 3D radiation pattern depicted in Figure 6.3 can be obtained.

$$\begin{cases} a_1 = cos^2(\theta) \cdot (1 - cos^{40}(\phi)) \\ a_2 = (1 - cos^2(\theta)) \cdot (1 - cos^{40}(\phi)) \\ a_3 = cos^{40}(\phi) \\ G(\theta, \phi) = a_1 \cdot G_{\phi_1} + a_2 \cdot G_{\phi_2} + a_3 \cdot G_{\theta} \end{cases} \tag{6.24}$$



(a) (b) (c)

Figure 6.2: *(a) Original experimental radiation pattern sections on θ, ϕ_1, and ϕ_2 planes; (b) approximation procedure forcing identical values on intersections; and (c) radial projection of the approximated radiation pattern sections. These are used to reconstruct the 3D radiation pattern.*



(a) 3D view.

(b) Top view.

(c) Front view.

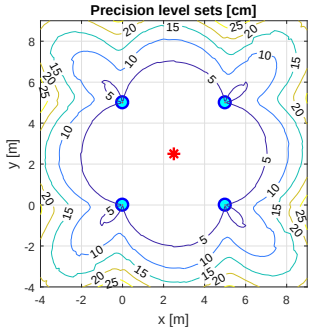
(d) Lateral view.

Figure 6.3: *Views of the reconstructed anchor's antenna radiation pattern.*

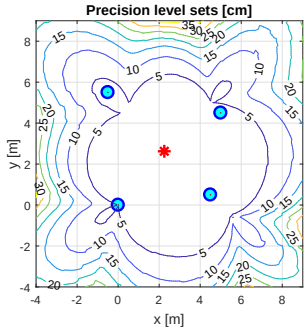
6.2.5 Analytical results of CRLB analysis

The CRLB analysis has been carried out here for two different representative distributions of four anchors: a symmetric layout and a random one, as shown in Figure 6.4. The ripples of the contour lines in Figure 6.4 are to be expected due to the anisotropy of the radiation pattern in Figure 6.3.

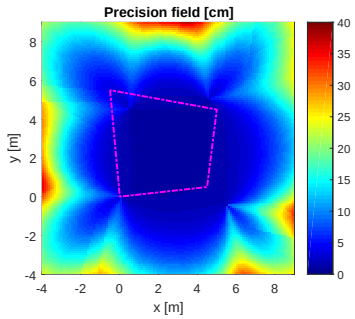
The best precision is obtained within the convex hull defined by the anchors. In this case, it is about ±5 cm with 99% confidence level (i.e. k = 2.58). A realistic non-isotropic transmitting antenna gain (DWM1000 module [113]) is also applied for the estimation of the SNR, hence the slight fluctuations in the represented values.



(a) Precision level sets, symmetric anchors.



(b) Precision level sets, random anchors.



(c) Precision colourmap, random anchors.

Figure 6.4: *Precision level sets for symmetric and random anchors, and precision colourmap for random anchors. The magenta trapezoid is the convex hull defined by four anchors.*

6.2.6 Failure analysis

Studying the TDoA map, which is a geometrical representation of the TDoA measurements, is crucial for the mathematical characterisation of the localisation problem in hand [84]. For instance, IPSs suffer from so-called *flipping uncertainty*, which is a well known problem of geometrical origin [85]. Thus, I define the *flyable area* based on a combination of the previous CRLB analysis and the bifuraction envelope below derived from a geometrical study. The aim is to bound the usable area of the IPS.

Bifurcation curve The bifurcation curve is the projection of the TDoA map boundaries from the τ-plane (pseudorange space) to the space of source localisation (2D in this case). The bifurcation curve as defined in [84] is the quintic curve $\tilde{E}(\mathbf{x})$ depicted by the roots of a polynomial $P(\mathbf{x})$ which is the representation of the TDoA map constraints. The definition of $P(\mathbf{x})$ and some examples of algebraic equations of $\tilde{E}(\mathbf{x})$ can be found in [72], whilst its rigorous derivation is presented in [84] using tools like

exterior algebra formalism and Minkowski space. This formulation is invariant under permutation of the TDoA measurements, which means that scheduling does not affect this analysis. Any TDoA-based system has a unique solution of the positioning problem if $P(\mathbf{x})$ is negative, which defines the region outside the bifurcation curves surrounding the anchors. The multilateration algorithm within the bifurcation curves (convex regions) returns either two mirrored solutions or complex solutions with no physical meaning. An example of a bifurcation curve is shown in Figure 6.5(a) for the case of three anchors $\{m_2, m_3, m_4\}$.

Bifurcation envelope For positioning systems comprising several antennae, the bifurcation curve will change dynamically depending on the pair of Times-of-Arrival (TOAs) considered in each TDoA query. As discussed earlier, the system fails to estimate the position of a vehicle within the concave regions of the bifurcation curves (containing the anchors). In order to ensure a unique solution for any possible pairing, a so-called *bifurcation envelope* is defined, which bounds all bifurcation curves on each anchor (e.g. one curve surrounding each anchor for three anchors, and four curves for four anchors). In Figure 6.5(b,c), the *flyable area* shaded in yellow is defined as the intersection of two areas:

1. The unique-solution area defined as the intersection of all concave areas outside each green bifurcation envelope (i.e. not including anchors).

2. The region with acceptable precision returned by the CRLB analysis.

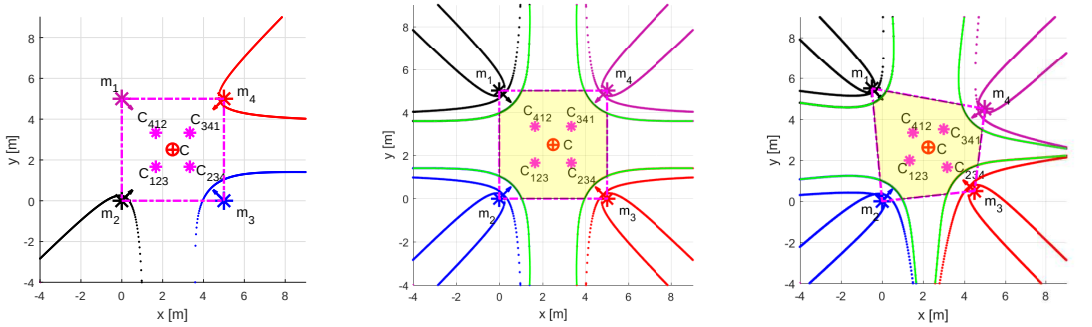
6.3 Proposed filtering process

Thus far, I have analysed precision and failure of UWB-based IPSs based on pseudorange multilateration algorithm and signal TDoA. The aim here is to develop a filtering process to improve its precision and accuracy.

6.3.1 Initial filter design

In the experimental setting, the vehicle to be positioned is a `Crazyflie 2.0` nano-quadcopter and the IPS is the `Loco Positioning System`. This is already equipped with an Extended Kalman Filter (EKF) [114, 115], which transforms raw sensor measurements into better estimates of the state of the drone. The EKF developers note that the position estimates are affected by a bias, which appears to be non-uniform in space. That is to say, the quadcopter estimates to be in a position that is shifted from the actual one.

The original idea was to use multiple filtering layers to enhance both precision and accuracy (see Figure 6.6). While the EKF is the first filter by default, the remaining ones may be applied in any order. The reason why the debiasing filter is applied last is



(a) Bifurcation curve for three anchors.

(b) Bifurcation curves, envelope and flyable area for four symmetric anchors.

(c) Bifurcation curves, envelope and flyable area for four random anchors.

Figure 6.5: *Bifurcation curves, bifurcation envelopes and flyable area for three anchors, four symmetric anchors, and four random anchors. There is a single bifurcation curve for three anchors whereas there are four bifurcation curves for four anchors (green line showing their envelope). The i^{th} transmitting anchor is represented by m_i, with m_1 being disregarded in Figure (a). The centroids of triplet (m_i,m_j,m_k) is represented by C_{ijk}, whilst C is the collective centroid. Figures (b) and (c) also show the* flyable area *(shaded in yellow) and the convex hull defined by the four anchors (dotted magenta trapezoid). The latter is taken as the region with acceptable precision.*

that the bias values can change dramatically throughout the flying area whilst precision values do not (and they also have smaller magnitudes). Hence it is better to work on a more precise estimate to avoid selecting the wrong value of the bias. The proposed filtering process is defined as follows:

1. Extended Kalman Filter.
2. Saturation (and artificial smoothing).
3. Correction of position through 4th order Adams-Moulton (AM4) correction.
4. Debiasing filter.

Unfortunately, precision was not noticeably enhanced. Therefore, only the EKF and the *Debiasing Filter* are implemented for the experiments carried out later in this chapter. Nonetheless, the mathematical formulations of the other filtering layers are included in Appendix 6.3.4 to support future work.

6.3.2 Debiasing filter

A *Debiasing Filter* (DF) is proposed and developed aiming to reduce systematic biases. The aim of the DF is to increase the accuracy of the localisation of the drone by subtracting the expected bias of the measurements. Assuming that discrete distributions

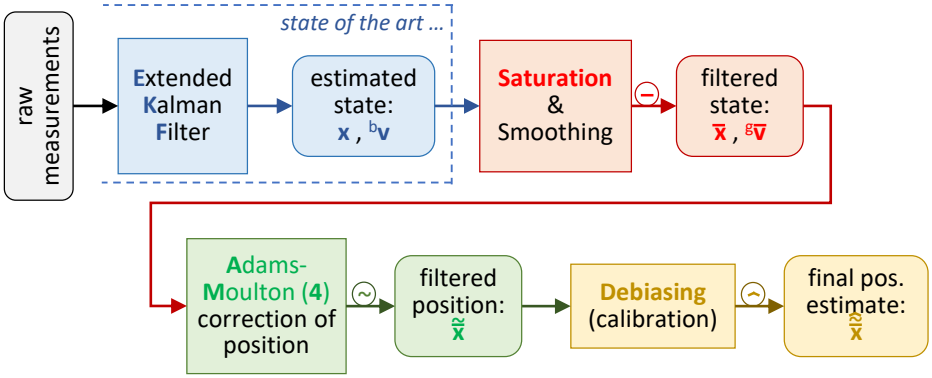


Figure 6.6: *Proposed filtering process consisting of four steps returning estimates identified with their respective filter symbols:* $\bar{\mathbf{x}}$ *stands for saturated,* $\tilde{\mathbf{x}}$ *for dynamically corrected, and* $\hat{\mathbf{x}}$ *for debiased. The superscripts* ^{b} *refers to body frame and* ^{g} *to inertial frame.*

of variances and biases (section 6.4.1) have been obtained by statistical post-processing of consecutive position measurements, two major complications arise:

1. The bias values are available only at a limited set of points, and therefore they need to be interpolated to cover the continuous domain. This interpolation is represented by a 3D surface for 2D positioning (not necessarily defined on a quadrangular grid), and by a 4D hypersurface for 3D positioning. Needless to say that the latter is considerably more complex.

2. The obtained bias distribution is a function of the real position of the drone, while the bias that needs to be subtracted from the measurement is a function of the measurement itself. This requires a change of domain from real position (100% precise and accurate) to measured position (imprecise and inaccurate).

In order to understand the debiasing process, at first a general explanation is provided starting from the desired final product - i.e. the debiased measurement. Refer to Figure 6.7 and the legend of the position variables listed in (6.25).

$$\begin{array}{cc} \mathbf{x} = (x, y) & \text{position} \\ \bar{\mathbf{x}} & \text{measured } \mathbf{x} \\ \mathbf{X}_{ij} = (X_{ij}, Y_{ij}) & \text{actual } \mathbf{x} \text{ of the } i\text{th - } j\text{th point} \\ \hat{\mathbf{x}} & \text{debiased } \mathbf{x} \end{array} \tag{6.25}$$

In Figure 6.7, every real position, surrounding the expected result of the debiasing filter, has a certain probability to be the father of the measured position. In fact, any measured position can be expressed as shown in (6.26). For instance, the measured $\bar{x}$ component of the position vector is obtained by summing to the actual (real) abscissa

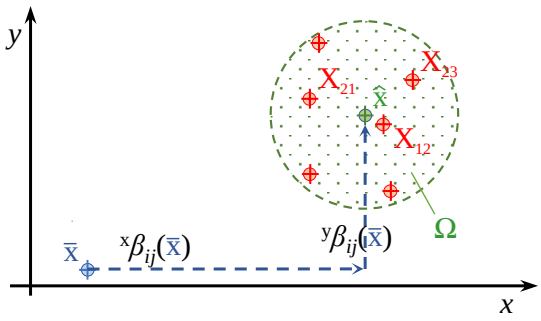


Figure 6.7: *Expected result of the debiasing filter ($\hat{\mathbf{x}}$) applied on a measured posistion ($\bar{\mathbf{x}}$) in 2D. The cloud of real positions ($\mathbf{X}_{ij}$) is constrained by the boundary $\boldsymbol{\Omega}$.*

(X) the relative bias in x-direction (${}^{\mathrm{x}}b\,(\mathbf{X})$) and a random fluctuation ($\mathcal{R}$), which is function of the evaluated variance (${}^{\mathrm{x}}p(\mathbf{X})$) in x-direction of the measurements in that specific real position.

$$\begin{aligned} \bar{x} &= X_{ij} + {}^{\mathrm{x}}b\,(\mathbf{X}_{ij}) + \mathcal{R}\,({}^{\mathrm{x}}p(\mathbf{X}_{ij}), \mathbf{x}) \\ \bar{y} &= Y_{ij} + {}^{\mathrm{y}}b\,(\mathbf{X}_{ij}) + \mathcal{R}\,({}^{\mathrm{y}}p(\mathbf{X}_{ij}), \mathbf{x}) \end{aligned} \tag{6.26}$$

Assuming that the $\mathcal{R}$ fluctuation is acceptably small, every real position (shown in Figure 6.7) can be obtained from the measurement $\bar{x}$ by subtracting the bias associated to the same real position.

Follows the derivation of the modified bias maps ${}^{\mathrm{x}}\beta\,(\bar{\mathbf{x}})$ and ${}^{\mathrm{y}}\beta\,(\bar{\mathbf{x}})$ in a 2D environment. For better understanding refer to Figures 6.8, 6.9, and 6.10. As shown in section 6.4.1 discrete mappings of the variances (${}^{\mathrm{x}}p_{ij}$ and ${}^{\mathrm{y}}p_{ij}$) and biases (${}^{\mathrm{x}}b_{ij}$ and ${}^{\mathrm{y}}b_{ij}$) in both $\hat{\mathbf{i}}$ and $\hat{\mathbf{j}}$ directions are evaluated. The real points (used to create the map) are expressed in the form $\mathbf{X}_{ij} = (X_{ij}, Y_{ij})$. For sake of simplicity, the measurements have been taken on a regular quadrangular mesh. Note that the precision of the EKF estimated positions provided by the IPS consists of the standard deviation ($\pm\sigma$) defined by the following relations between variance and standard deviation:

$$\begin{aligned} {}^{\mathrm{x}}p_{ij} &= {}^{\mathrm{x}}\sigma_{ij}^2 \quad \text{variance in } x \text{ direction} \\ {}^{\mathrm{y}}p_{ij} &= {}^{\mathrm{y}}\sigma_{ij}^2 \quad \text{variance in } y \text{ direction} \end{aligned} \tag{6.27}$$

The normal probability function (Gaussian distribution) around the $\mathbf{X}_{ij}$ position evaluated in x direction can be written as

$${}^{\mathrm{x}}\gamma_{ij} = {}^{\mathrm{x}}\mathcal{N}\,(X_{ij}, {}^{\mathrm{x}}p_{ij}) = (2\pi \cdot {}^{\mathrm{x}}p_{ij})^{-1/2} \cdot e^{-\frac{1}{2}\frac{\left(x - X_{ij}\right)^2}{{}^{\mathrm{x}}p_{ij}}} \quad | \quad \int_{-\infty}^{+\infty} {}^{\mathrm{x}}\gamma_{ij} \cdot dx = 1 \tag{6.28}$$

and similarly in y direction

$$ {}^{\mathrm{y}}\gamma_{ij} = {}^{\mathrm{y}}\mathcal{N}\left(Y_{ij}, {}^{\mathrm{y}}p_{ij}\right) = \left(2\pi \cdot {}^{\mathrm{y}}p_{ij}\right)^{-1/2} \cdot e^{-\frac{1}{2}\frac{\left(y-Y_{ij}\right)^2}{{}^{\mathrm{y}}p_{ij}}} \quad | \quad \int_{-\infty}^{+\infty} {}^{\mathrm{y}}\gamma_{ij} \cdot dy = 1 \tag{6.29} $$

Note that the infinite integral of both functions is the unity, although in the following derivations it is going to be integrated within three standard deviations, and its value is shown in (6.30).

$$ \int_{-3\cdot{}^{x}\sigma}^{+3\cdot{}^{x}\sigma} {}^{\mathrm{x}}\gamma_{ij} \cdot dx = \int_{-3\cdot{}^{y}\sigma}^{+3\cdot{}^{y}\sigma} {}^{\mathrm{y}}\gamma_{ij} \cdot dy = 0.9973 \simeq 1 \tag{6.30} $$

The shapes of ${}^{\mathrm{x}}\gamma_{ij}$ and ${}^{\mathrm{y}}\gamma_{ij}$ around a general point $\mathbf{x}_{ij}$ are shown in Figures 6.8 and 6.9.

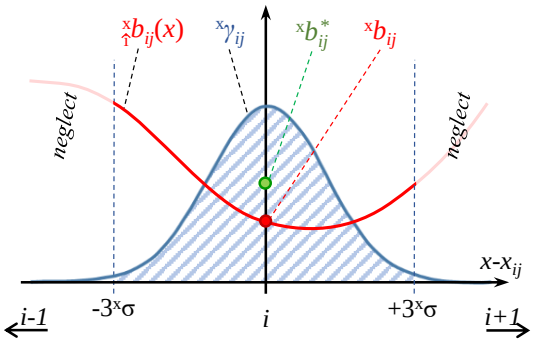


Figure 6.8: *Representation debiasing in 1D.*

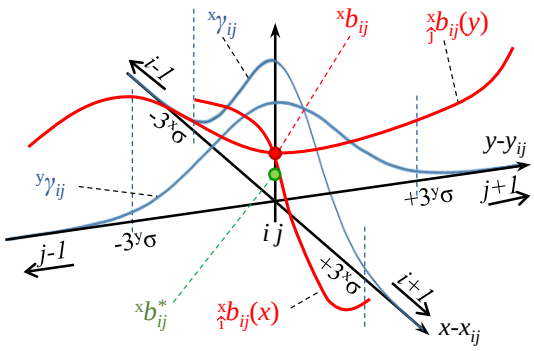


Figure 6.9: *Representation debiasing in 2D.*

For the purpose of the following derivation, continuous interpolating functions of the x and y biases both along the $\hat{\mathbf{i}}$ and $\hat{\mathbf{j}}$ axis have to be obtained (these could be cubic splines). Therefore, from the biases of x measurements around any $\mathbf{x}_{i,j}$ position, two interpolating functions can be obtained: ${}^{\mathrm{x}}_{\hat{\mathbf{i}}}b_{ij}$ and ${}^{\mathrm{x}}_{\hat{\mathbf{j}}}b_{ij}$ respectively along the $\hat{\mathbf{i}}$ and $\hat{\mathbf{j}}$ axis. Analogously, for the biases of y measurements, the ${}^{\mathrm{y}}_{\hat{\mathbf{i}}}b_{ij}$ and ${}^{\mathrm{y}}_{\hat{\mathbf{j}}}b_{ij}$ interpolating functions can be defined. The intention is to write weighted averages of the biases around the estimation position in order to estimate the expected biases. However, instead of performing a surface integral, the average of two integrals in perpendicular directions is considered. Figure 6.8 depicts the problem in $\hat{\mathbf{i}}$ direction for the bias of the measurement of the x-component of the position. The interpolated bias function ${}^{\mathrm{x}}_{\hat{\mathbf{i}}}b_{ij}$ multiplied by the weighting probability distribution ${}^{\mathrm{x}}\gamma_{ij}$ is integrated in x direction and normalised by the length of the considered interval. A fair coverage factor of $k = 3$ has been selected. This means that approximately 99% (level of confidence) of the measurements, of the real position ${}^{\mathrm{r}}x_{ij}$, will fall in the interval between $(x - 3^{\mathrm{x}}\sigma)_{ij}$ and $(x + 3^{\mathrm{x}}\sigma)_{ij}$, where ${}^{\mathrm{x}}\sigma_{ij}$ is the standard deviation of the Gaussian distribution ${}^{\mathrm{x}}\gamma_{ij}$.

The same integral can be evaluated in $\hat{\mathbf{j}}$ direction and the two integral values can be averaged in order to obtain the corrected bias value of x-component measurements, as shown in (6.31). The same process can be applied for evaluating the corrected bias of the y-component measurements (6.32).

$$ {}^{\mathrm{x}}b^*_{ij} = \frac{1}{2}\int_{-3\cdot{}^{\mathrm{x}}\sigma_{ij}}^{+3\cdot{}^{\mathrm{x}}\sigma_{ij}} {}^{\mathrm{x}}\gamma_{ij}\cdot {}^{\mathrm{x}}_{\hat{\mathbf{i}}}b_{ij}\cdot dx + \frac{1}{2}\int_{-3\cdot{}^{\mathrm{y}}\sigma_{ij}}^{+3\cdot{}^{\mathrm{y}}\sigma_{ij}} {}^{\mathrm{y}}\gamma_{ij}\cdot {}^{\mathrm{x}}_{\hat{\mathbf{j}}}b_{ij}\cdot dy \tag{6.31} $$

$$ {}^{\mathrm{y}}b^*_{ij} = \frac{1}{2}\int_{-3\cdot{}^{\mathrm{x}}\sigma_{ij}}^{+3\cdot{}^{\mathrm{x}}\sigma_{ij}} {}^{\mathrm{x}}\gamma_{ij}\cdot {}^{\mathrm{y}}_{\hat{\mathbf{i}}}b_{ij}\cdot dx + \frac{1}{2}\int_{-3\cdot{}^{\mathrm{y}}\sigma_{ij}}^{+3\cdot{}^{\mathrm{y}}\sigma_{ij}} {}^{\mathrm{y}}\gamma_{ij}\cdot {}^{\mathrm{y}}_{\hat{\mathbf{j}}}b_{ij}\cdot dy \tag{6.32} $$

For the following remaining derivations refer to Figure 6.10. Rewriting the decomposition expressed in (6.25), neglecting the fluctuating part, the measured position will be shifted from the original one approximately by the corrected weighted biases expressed in (6.31)-(6.32):

$$ \begin{aligned} \bar{X}_{ij} &= X_{ij} + {}^{\mathrm{x}}b^*_{ij} \\ \bar{Y}_{ij} &= Y_{ij} + {}^{\mathrm{y}}b^*_{ij} \end{aligned} \tag{6.33} $$

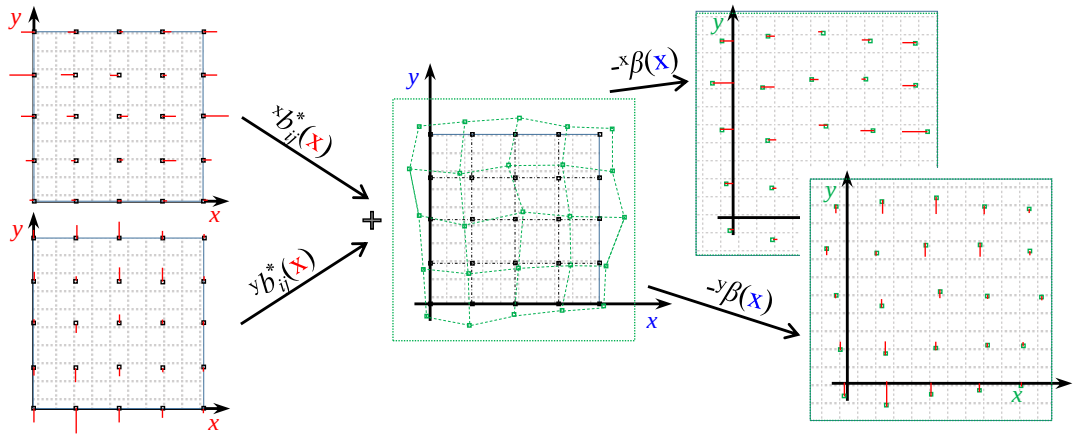


Figure 6.10: *Diagram of derivation of debiasing functions in x and y directions ${}^{x}\beta(\mathbf{x})$ and ${}^{y}\beta(\mathbf{x})$.*

Therefore, while the original experimentally-obtained biases were distributed on a regular quadrangular grid $[{}^{\mathrm{r}}x_{i,j}\ {}^{\mathrm{r}}y_{i,j}]$, the new corrected biases b^* can be distributed over a deformed grid $[\bar{x}_{i,j}\ \bar{y}_{i,j}]$. As shown in (6.34), the two new corrected bias distributions of x measurements $({}^{\mathrm{x}}m_{i,j})$ and y measurements $({}^{\mathrm{y}}m_{i,j})$ can be interpolated, obtaining bias surfaces that are function of the measured positions.

$$ \begin{aligned} {}^{\mathrm{x}}m_{ij} &= \begin{bmatrix} \bar{X}_{ij} & \bar{Y}_{ij} & {}^{\mathrm{x}}b^*_{ij} \end{bmatrix}^{\mathrm{T}} \overset{interp.}{\rightarrow} {}^{\mathrm{x}}\beta(\mathbf{x}) \\ {}^{\mathrm{y}}m_{ij} &= \begin{bmatrix} \bar{X}_{ij} & \bar{Y}_{ij} & {}^{\mathrm{y}}b^*_{ij} \end{bmatrix}^{\mathrm{T}} \overset{interp.}{\rightarrow} {}^{\mathrm{y}}\beta(\mathbf{x}) \end{aligned} \tag{6.34} $$

Finally, it is possible to subtract from the measured position these new interpolating bias functions in order to obtain a debiased measurement:

$$\begin{aligned} \hat{x} &= x - {}^{\mathrm{x}}\beta(\mathbf{x}) \\ \hat{y} &= y - {}^{\mathrm{y}}\beta(\mathbf{x}) \end{aligned} \tag{6.35}$$

Two example of calibrated debiasing function fields can be found in figures 6.21 and 6.22. The formulation of the Radial Basis Function Network (RBFN) used for the interpolation of the debiasing values is explained in the following section.

6.3.3 RBFN implementation

In this section the formulation of a Radial Basis Function Network (RBFN) is introduced for the interpolation of the debiasing distributions in both 2D (6.34) environment. In the following equations the change of variable expressed in (6.36) must be considered, so that the NN will work on strictly positive values.

$$\begin{gathered} b_{ij} = {}^{\mathrm{x}}b^*_{ij},\ {}^{\mathrm{y}}b^*_{ij},\ \text{or}\ {}^{\mathrm{z}}b^*_{ij} \quad \text{and} \\ \text{if } b_{\min} = \min\{b_{ij}\} < 0 \text{ then } b_{ij} = b^*_{ij} - b_{\min} \end{gathered} \tag{6.36}$$

The considered Gaussian activation function (f_{ij}) is defined on every marker ($\mathbf{X}_{ij}$) that was previously used for mapping purposes. Hence, every marker represents a node in the NN.

$$\begin{aligned} f_{ij}(\mathbf{x}) &= c_1 \cdot b_{ij} \cdot e^{-c_2 \cdot d^2} \\ d^2 &= \|\mathbf{x} - \mathbf{X}_{ij}\|^2 \end{aligned} \tag{6.37}$$

While the constant c_2 can be uniquely defined for each node (as follows in (6.38)-(6.41)), the constant c_1 is used for calibration of the RBFN interpolation. The value of c_1 can be chosen between 0 and 1 (by trial and error the best acceptable value is found to be $c_1 = 0.5$).

$$\begin{aligned} f_{ij}(\mathbf{X}_{ij}) &= c_1 \cdot b_{ij} \\ f_{ij}(\mathbf{X}_{mn}) &= c_1 \cdot b_{ij} \cdot e^{-c_2 \cdot d^2_{mn}} = \tfrac{1-c_1}{8} b_{mn} \\ d^2_{mn} &= \|\mathbf{X}_{mn} - \mathbf{X}_{ij}\|^2 \end{aligned} \tag{6.38}$$

$$\begin{aligned} \ln\left(c_3 \tfrac{b_{mn}}{b_{ij}}\right) &= -c_2 \cdot d^2_{mn} \\ c_3 &= \tfrac{1-c_1}{8 \cdot c_1} \end{aligned} \tag{6.39}$$

$$c_{2,mn} = -\frac{1}{d^2_{mn}} \ln\left(c_3 \frac{b_{mn}}{b_{ij}}\right) \tag{6.40}$$

The constant $c_{2,ij}(\theta)$ is a periodic function interpolating the values of $c_{2,mn}$ on a neighbourhood stencil of 8 points (in 2D).

$$\begin{gathered} c_{2,ij}(\theta) = \text{interp}\left(c_{2,mn}, \theta_{mn}\right) \ \text{ periodic in } [-\pi, +\pi] \\ mn = [(i+1, j)\,(i+1, j+1)\,(i, j+1)\,(i-1, j+1)\,(i-1, j) \\ (i-1, j-1)\,(i, j-1)\,(i+1, j-1)\,(i+1, j)] \\ \text{with} \quad \theta_{mn} = \text{atan}\left(\tfrac{Ymn - Yij}{Xmn - Xij}\right) \end{gathered} \tag{6.41}$$

Therefore, once all the $c_{2,ij}(\theta)$ constants have been defined, the resulting RBFN (shown in Figures 6.11-6.14) will be:

$$\begin{gathered} b^{(\mathrm{NN})}(\mathbf{x}) = b_{\min} + \sum_{i,j}^{N,M} f_{ij}(\mathbf{x}) \\ f_{ij}(\mathbf{x}) = c_1 \cdot b_{ij} \cdot e^{-c_{2,ij} \cdot d_{ij}^2} \\ d_{ij}^2 = \|\mathbf{x} - \mathbf{X}_{ij}\|^2 \\ c_1 = 0.5 \\ c_{2,ij} = c_{2,ij}(\theta) \text{ periodic in } [-\pi, +\pi] \\ \text{view angle } \theta_{ij} = \operatorname{atan}\left(\frac{y - Yij}{x - X_{ij}}\right) \end{gathered} \tag{6.42}$$

Since the Gaussian activation function are here defined so that they have influence only to the adjacent nodes, the debiasing filter (DF) code uploaded in the Crazyflie firmware takes account of this in order to optimise the performance in terms of real time computation. Therefore, not all $f_{ij}(\mathbf{x})$ are computed at every time step, but only those that are within a distance (d_{ij}) of 1 m from the measured position.

$$b^{(\mathrm{NN}^*)}(\mathbf{x}) = b_{\min} + \sum_{i,j|d_{ij}<1\mathrm{m}}^{N,M} f_{ij}(\mathbf{x}) \tag{6.43}$$

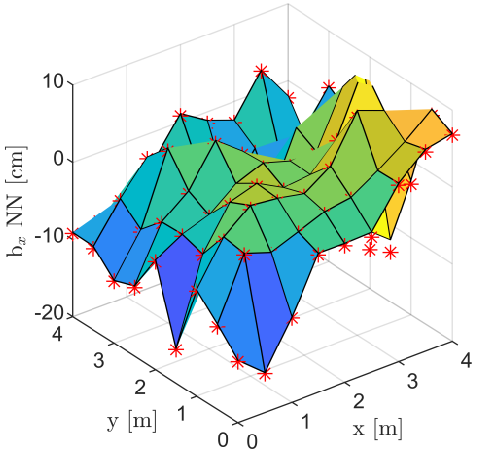


Figure 6.11: *RBFN of bias values on markers points (red stars) for estimations of the x component of the position.*

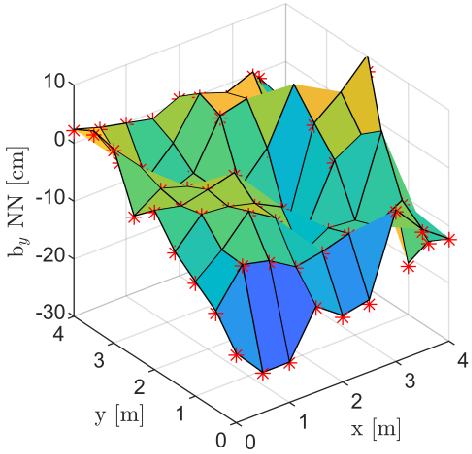


Figure 6.12: *RBFN of bias values on markers points (red stars) for estimations of the y component of the position.*

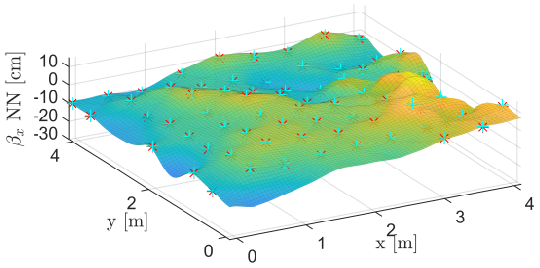


Figure 6.13: *RBFN surface interpolating bias values of estimations of the x component of the position.*

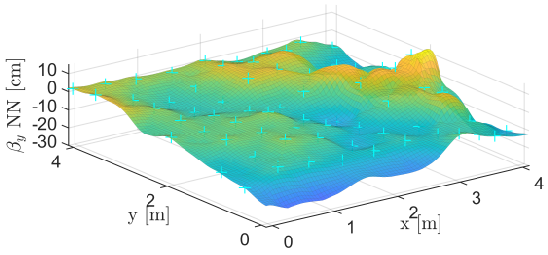


Figure 6.14: *RBFN surface interpolating bias values of estimations of the y component of the position.*

6.3.4 About other filters

In this section a formulation of the other filters is proposed: saturation, smoothing and Adams-Moulton correction. However, no experiments are carried out to test and validate such filters since the focus of the chapter is on the most crucial debiasing filter. Although, I thought that providing extra layers of filters would most likely improve the overall precision and accuracy of the positioning system. The saturation filter consists of limiting the eventual estimation overshoots of both velocity and position. Two types of velocity overshoots are considered: a maximal overestimation and a maximum allowed time derivative. For example, each component of the estimated velocity vector will be limited by a maximum velocity value that can be different in the three directions. For instance, the horizontal velocity components (in x and y direction) can be limited by the maximum cruise speed, while the vertical velocity (in z direction) could be at maximum two times the free-fall speed. Moreover, also the time derivative of the velocity is limited by the expected maximum acceleration. Also this values might be smaller horizontally and greater vertically, but for simplicity the derivatives in all directions have been limited by the same amount. The effect of the presented saturation filter on a casual sequence of velocity estimations is depicted in Figure 6.15. The filter will act analogously on a positions sequence - i.e. equations (6.48)-(6.50).

Let's start defining the scalar variation of velocity (${}^{\mathrm{g}}\Delta \mathrm{v}^{(k)}$) as the modulus of the difference between the EKF-estimated velocity at the current time ${}^{(k)}$ and the filtered velocity at the previous timestep ${}^{(k-1)}$:

$$ {}^{\mathrm{g}}\Delta \mathrm{v}^{(k)} = \left\| \Delta {}^{\mathrm{g}}\mathbf{v}^{(k)} \right\| = \left\| {}^{\mathrm{g}}\mathbf{v}^{(k)} - {}^{\mathrm{g}}\bar{\mathbf{v}}^{(k-1)} \right\| \tag{6.44} $$

This value should not exceed the maximum allowed speed variation (limited by the maximum acceleration, $a_{\max}$). Therefore, the (6.45) inequality can be used to formulate the ceiling of components of the velocity variation ($\Delta {}^{\mathrm{g}}\bar{\mathbf{v}}^{(k)}$) in (6.46).

$$ {}^{\mathrm{g}}\Delta \mathrm{v}^{(k)} \leqslant \Delta t \cdot a_{\max} \tag{6.45} $$

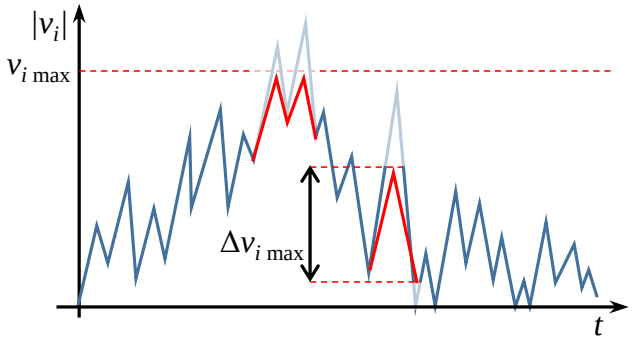


Figure 6.15: *Representation of the presented saturation filter on some velocity estimates in ith direction. In* blue *the original EKF estimate and in* red *the correction. Note that the measurements are discrete and represented by the peaks, the linear interpolation between measurements is only for visualisation purposes.*

$$\Delta^{\mathrm{g}}\bar{\mathbf{v}}^{(k)} = \min\left(\Delta t \cdot a_{\max} \cdot {}^{\mathrm{g}}\Delta \mathrm{v}^{(k)^{-1}},\, 1\right) \cdot \Delta^{\mathrm{g}}\mathbf{v}^{(k)} \tag{6.46}$$

At this point, every velocity component has to be limited by the maximum allowed speeds ($v_{i\,\max}$) as explained earlier. The components of the final filtered velocity vector (${}^{\mathrm{g}}\bar{\mathbf{v}}^{(k)}$) are defined as follows:

$$\begin{gathered} {}^{\mathrm{g}}\bar{v}_i^{(k)} = \min\left({}^{\mathrm{g}}\bar{v}_i^{(k-1)} + \Delta^{\mathrm{g}}\bar{v}_i^{(k)},\, v_{i\,\max}\right) \\ \mathbf{v}_{\max} = \begin{bmatrix} v_{1\,\max} & v_{2\,\max} & v_{3\,\max} \end{bmatrix} \end{gathered} \tag{6.47}$$

The same filtering procedure can be applied on the position estimates, but in this case only the time derivative will be limited. As before, the inequality (**??**) can be used to define the variation of each ith component ($i = 1, 2, 3$) of the filtered position $\Delta\bar{\mathbf{x}}^{(k)}$ expressed in (6.49).

$$\begin{gathered} \Delta \mathrm{x}_i^{(k)} \leqslant \Delta t \cdot v_{i\,\max} \\ \text{where} \\ \Delta \mathrm{x}^{(k)} = \left\|\Delta \mathbf{x}^{(k)}\right\| = \left\|\mathbf{x}^{(k)} - \bar{\mathbf{x}}^{(k-1)}\right\| \end{gathered} \tag{6.48}$$

$$\Delta\bar{x}_i^{(k)} = \min\left(\Delta t \cdot v_{i\,\max} \cdot \Delta \mathrm{x}^{(k)^{-1}}, 1\right) \cdot \Delta x_i^{(k)} \tag{6.49}$$

Finally the filtered estimation of the position can be defined as follows:

$$\bar{\mathbf{x}}^{(k)} = \bar{\mathbf{x}}^{(k-1)} + \Delta\bar{\mathbf{x}}^{(k)} \tag{6.50}$$

Additionally embodied in this filtering step is the smoothing filter that aims at artificially reducing the oscillations of the measurements by averaging the history of

$n+1$ measurements. The average is weighted so that the contribution of the last measurement ($\bar{\mathbf{x}}^{(k)}$) is more important.

$$\begin{aligned} \bar{\mathbf{x}}_s^{(k)} &= (1-q)^n \cdot \bar{\mathbf{x}}^{(k-n)} + \sum_{l=0}^{n-1} q \cdot (1-q)^l \cdot \bar{\mathbf{x}}^{(k-l)} \\ q &= q_{\%}/100 \end{aligned} \tag{6.51}$$

where $q_{\%}$ is the percentage of influence of the last measurement.

Thirdly, the main idea of the Adams-Moulton dynamic prediction is to correct the EKF estimations, that consider only the measurements at the current time, with a prediction that takes into account the previous history of estimations. Hence, a multistep scheme should be selected. As far as the prediction of the position is considered, an implicit scheme can be chosen since the estimation of the time derivative of the position (estimated velocity) is known. This will make the prediction more trusty since linked to the current state of the drone. Usual multistep predictor-corrector schemes consist in the combination of an explicit predictor - e.g. Adams-Bashforth (AB) of order n - and an implicit corrector - e.g. Adams-Moulton (AM) of order m - that can use the prediction of the derivative function given by AB(n). In the exception of the presented correction filter, AM can be directly used. In (6.52) an AM scheme of 4th order is defined for predicting the position $\bar{\mathbf{x}}_{\text{AM4}}^{(k)}$ by the use of the filtered (from previous sections **??** and **??**) velocity estimates at the current and the previous four time steps. A higher order AM scheme could be selected for considering a longer history of estimates.

$$\begin{aligned} \bar{\mathbf{x}}_{\text{AM4}}^{(k)} = \bar{\mathbf{x}}^{(k-1)} + \frac{\Delta t}{720} \cdot \big(& 251 \cdot {}^{\text{g}}\bar{\mathbf{v}}^{(k)} + 646 \cdot {}^{\text{g}}\bar{\mathbf{v}}^{(k-1)} \\ & -264 \cdot {}^{\text{g}}\bar{\mathbf{v}}^{(k-2)} + 106 \cdot {}^{\text{g}}\bar{\mathbf{v}}^{(k-3)} - 19 \cdot {}^{\text{g}}\bar{\mathbf{v}}^{(k-4)} \big) \end{aligned} \tag{6.52}$$

One aspect that is considered by the EKF estimates but neglected by the AM4 prediction is the effect that the control command input have on the state. For instance, the EKF takes into account the given thrust input. Intuitively, at low cruise speed the drone's dynamics are very affected by controls input so the EKF estimate is more important, while at high cruise speed the drone dynamics at short period are mostly a direct effect of the stored momentum of the flying body (that will move by inertia). Therefore, a vectorial weighting function $\boldsymbol{\alpha}$ is here defined so that the filtered correction of the estimated position follows this logic about the cruise speed. The formulation of the newly filtered position $\tilde{\bar{\mathbf{x}}}^{(k)}$ is shown in (6.53) while the shape of the ith weight component function (6.54) is shown in Figure 6.16.

$$\tilde{\bar{\mathbf{x}}}^{(k)} = \boldsymbol{\alpha} \odot \bar{\mathbf{x}}^{(k)} + (1-\boldsymbol{\alpha}) \odot \bar{\mathbf{x}}_{\text{AM4}}^{(k)} \tag{6.53}$$

where $\odot$ is the component-wise multiplication operation, so that given two vectors $\mathbf{a}$ and $\mathbf{b}$ the resulting vector components are $c_i = a_i \cdot b_i$, with $i = 1, 2, 3$.

$$\alpha_i = \exp\left(-\frac{1}{2}\left(\frac{\bar{a}_i^{(k)}}{a_{i\,\mathrm{f}}}\right)^2\right)(1-\alpha_{\min}) + \alpha_{\min} \tag{6.54}$$

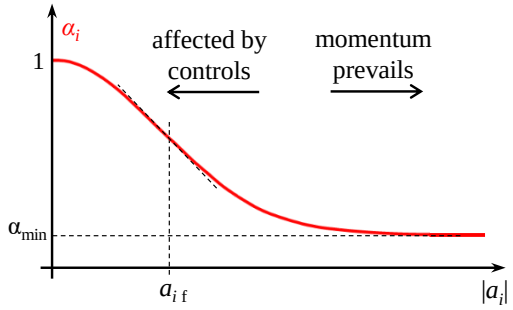


Figure 6.16: *Weight function for averaging between EKF state estimations and AM4 predictions; $v_{i\,f}$ is the flipping velocity, $v_{i\,max}$ is the maximum expected speed, and α_{min} is an additional calibration parameter.*

6.4 Design of experiments

6.4.1 IPS mapping

In order to build the maps, a large number of measurements ($N = 700$) are taken at a sampling frequency of 100 Hz while keeping the drone still for at least 30 seconds on each marker (X_{ij}). Then, the raw stream of data is post-processed omitting the transients corresponding to the movement between markers. The drone is kept aligned with the x axis and parallel to the floor, as the effect of its attitude is not being investigated. Finally, the bias (b), standard deviation (σ) and mean squared error (MSE) are computed. For instance, their values in the x direction (superscript $^{\mathrm{x}}$) are as follows:

$$\begin{aligned} {}^{\mathrm{x}}b_{ij} &= N^{-1}\textstyle\sum_{k=1}^{N} x^{(k)} - X_{ij} \\ {}^{\mathrm{x}}\sigma_{ij} &= \left({}^{\mathrm{x}}\mathrm{MSE}_{ij} - {}^{\mathrm{x}}b_{ij}^2\right)^{0.5} \\ {}^{\mathrm{x}}\mathrm{MSE}_{ij} &= N^{-1}\textstyle\sum_{k=1}^{N}\left(x^{(k)} - X_{ij}\right)^2 \end{aligned} \tag{6.55}$$

where $x^{(k)}$ is the k^{th} position measurement. The resulting maps can be found in Figure 6.17 and Figure 6.19 for x-direction measurements, and Figure 6.18 and Figure 6.20 for y-direction measurements.

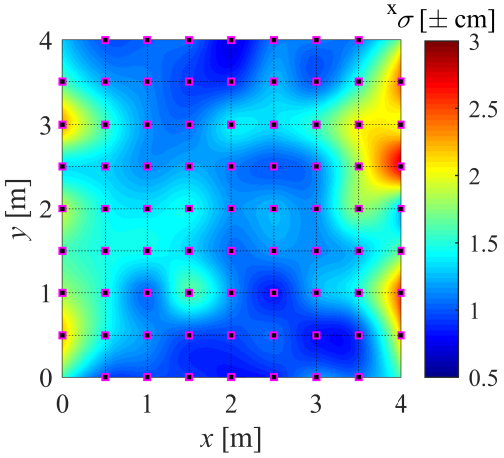


Figure 6.17: *Precision mapping of the x component of the position ($\pm{}^{x}\sigma$).*

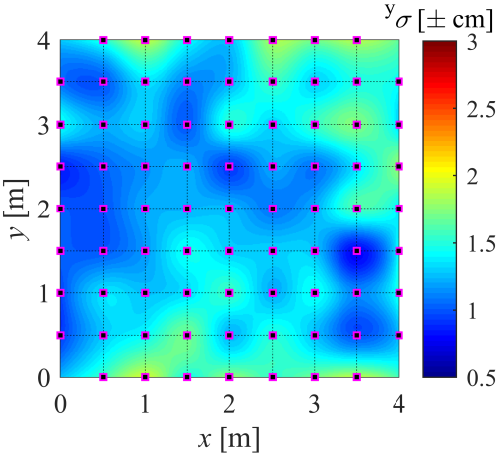


Figure 6.18: *Precision mapping of the y component of the position ($\pm{}^{y}\sigma$).*

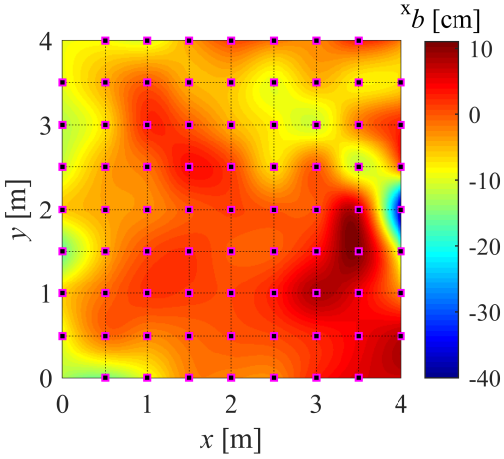


Figure 6.19: *Accuracy mapping of the x component of the position ($\pm{}^{x}b$).*

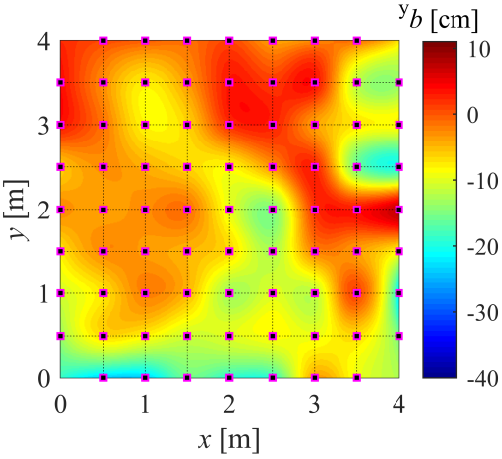


Figure 6.20: *Accuracy mapping of the y component of the position ($\pm{}^{y}b$).*

6.4.2 DF calibration and validation

After the mapping of the bias of x and y components of the position estimations with the available IPS is completed, the debiasing filter must be calibrated using these data. The calibration consists actually in interpolating using the Radial-Basis-Function-Network (RBFN) formulated in Appendix 6.3.3. The final output will be something similar to Figures 6.21-6.22, or Figures 6.13-6.14.

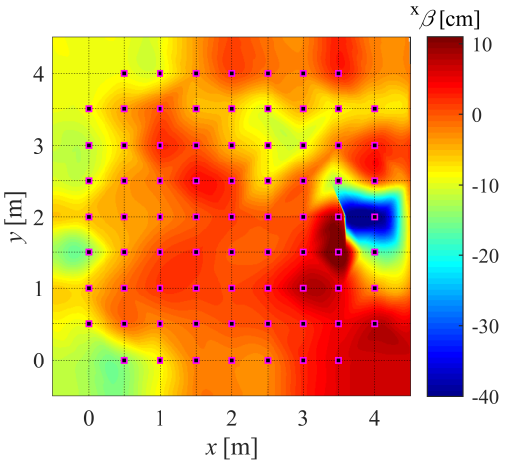


Figure 6.21: *Debiasing function for measurements of the x component of the position (${}^{x}\beta$).*

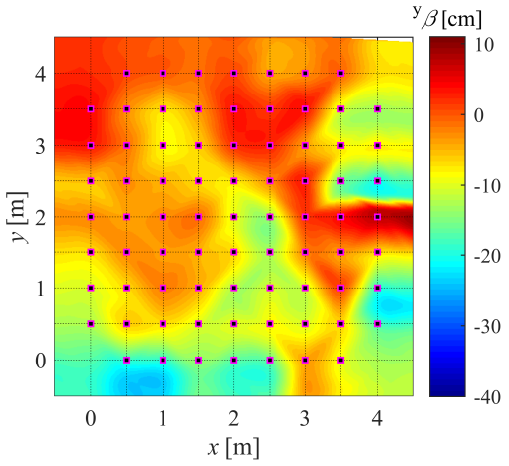


Figure 6.22: *Debiasing function for measurements of the y component of the position (${}^{y}\beta$).*

A second aim of the experiment described in this section is to validate the proposed DF. The estimations of position provided by the original IPS (which uses only EKF) are compared to those provided by the new IPS (EKF+DF) on a predefined set of staggered markers (ref. to Figure 6.1) that have not been previously used for calibration purposes. Variances and biases are evaluated to provide statistical insight on the performance of the newly developed filter.

6.4.3 DF validation in dynamic conditions

The aim of the experiment described in this section is to validate the proposed filter dynamically. Moreover, it can be used in order to calibrate the coefficients of the Adams-Moulton-4$^{\text{th}}$ (AM4), however this aspect will be investigated in further research. Let's define the positioning system to be IPS-1 when only EKF is activated, and IPS-2 when also DF is active. The aim is to compare the two with respect to dynamic (while the drone is moving) estimations of position at different cruise speed.

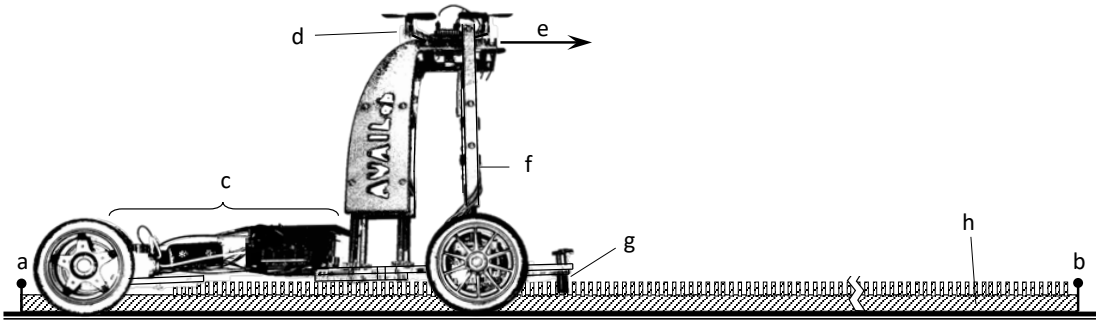


Figure 6.23: *Dynamic experiment setup: cf-stand rover on rail. (a) starting and (b) ending point of rail; (h) optical obstacles - i.e. nails; (g) optical infrared sensor; (c) power control unit, batteries and electric motors; (d) Crazyflie drone; (e) direction of movement; (f) embedded laser pointer. The optical sensor is actually aligned right underneath the drones' UWB antenna while performing the experiment.*

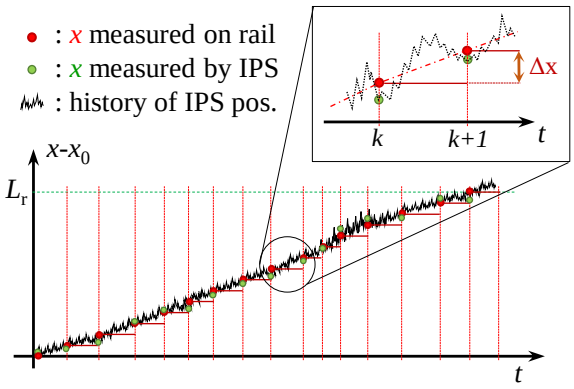


Figure 6.24: *Visualisation of rail and IPS position measurements. L_r is the total length of the rail.*

The mobile stand, on which the drone is mounted, is constrained to move along an encoder rail. In the frame of reference along the rail, the position (s) of the drone - i.e. of the optical sensor - is given by the count $n^{(k)}$ of the pins moving from the starting point (a) to the ending point (b).

$$s^{(k)} = \Delta s \cdot n^{(k)} \tag{6.56}$$

where Δs is the constant distance between each nail. Here $^{(k)}$ is the high frequency time step counting, which is the frequency at which data from all the sensors are recorded; therefore n is supposed to increment slower than k. The real position of the drone along the rail ($\mathbf{x}_r^{(k)}$) in the inertial frame of reference can be obtained by projecting $s^{(k)}$

by the angle between the rail and the x-axis, θ_r:

$$\begin{gathered}\mathbf{x}_\mathrm{r}^{(k)} = \left(x_\mathrm{r}^{(k)}, y_\mathrm{r}^{(k)}\right) \\ x_\mathrm{r}^{(k)} = s^{(k)} \cdot \cos(\theta_\mathrm{r}) \quad \text{and} \quad y_\mathrm{r}^{(k)} = s^{(k)} \cdot \sin(\theta_\mathrm{r}) \\ \theta_\mathrm{r} = \operatorname{atan}\left(\frac{y_\mathrm{b}-y_\mathrm{a}}{x_\mathrm{b}-x_\mathrm{a}}\right)\end{gathered} \tag{6.57}$$

Every time a new nail is detected, at time step $^{(p)}$, the estimated position by both IPS-1 and IPS-2 ($\mathbf{x}_1^{(k)}$,$\mathbf{x}_2^{(k)}$) and the actual position on the rail have been recorded and stored as follows in storage variables ξ and υ:

$$\begin{gathered}\text{if } n^{(k)} > n^{(k-1)} \text{ then:} \\ \xi_\mathrm{r}^{(p)} = x_\mathrm{r}^{(k)} \quad \upsilon_\mathrm{r}^{(p)} = y_\mathrm{r}^{(k)} \quad \tau^{(p)} = t^{(k)} \\ \xi_1^{(p)} = x_1^{(k)} \quad \upsilon_1^{(p)} = y_1^{(k)} \quad \xi_2^{(p)} = x_2^{(k)} \quad \upsilon_2^{(p)} = y_2^{(k)}\end{gathered} \tag{6.58}$$

The velocity estimations with IPS-1 and IPS-2 can also be compared to the discrete average velocity on the rail (6.59):

$$\begin{gathered}\nu_\mathrm{x}^{(p)} = \frac{\Delta\xi_\mathrm{r}^{(p)}}{\Delta\tau^{(p)}} \quad \text{and} \quad \nu_\mathrm{y}^{(p)} = \frac{\Delta\upsilon_\mathrm{r}^{(p)}}{\Delta\tau^{(p)}} \\ \Delta\xi_\mathrm{r}^{(p)} = \Delta s \cdot \cos(\theta_\mathrm{r}) = \text{const.} \\ \Delta\upsilon_\mathrm{r}^{(p)} = \Delta s \cdot \sin(\theta_\mathrm{r}) = \text{const.} \\ \Delta\tau^{(p)} = \tau^{(p)} - \tau^{(p-1)}\end{gathered} \tag{6.59}$$

where $\Delta\tau^{(p)}$ is the time passed between the detection of the p-th and $(p-1)$-th pins. Since the aim is to evaluate how well the IPS-2 performs with respect to IPS-1 at different cruise speed of the drone, multiple measurements are required at different speeds. The recorded position data, can be classified in groups corresponding to different speed ranges - e.g. a speed breaks vector can be defined as $[0.2, 0.4, 0.6, 0.8, 1]$. This experiment design, apart from DF, could be used also as platform to calibrate and validate the other filters presented in Appendix 6.3.4.

6.4.4 Square path experiment

The aim is to partially reproduce the experiment of a drone following a square path [114] (Figure 6.25). Since I want to investigate the performance of the only debiasing filter, it is suggested to perform an experiment that separates the IPS estimations from the automatic control dynamics of the drone. Therefore, the drone is not free to fly, but it is driven by the mobile support along the square path (which is the reference trajectory) and estimation of the position with and without the debiasing filter (DF) are recorded. The experiment is performed multiple times in order to collect some statistics on the DF's performance. The expected result is depicted in Figure 6.26. The debiased path estimation should be more accurate than the only EKF measurement, aiming towards the reference square path of the rail.

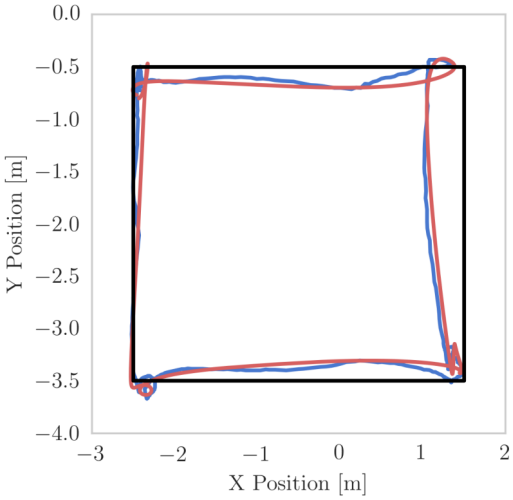


Figure 6.25: *Experiment performed by [114]. The drone was not constrained, free to move following through auto-pilot the square path multiple times. It compares the EKF estimate (blue line) with the actual position of the drone (red line). Both trajectories are not accurate since shifted from the actual wanted reference square path (black line).*

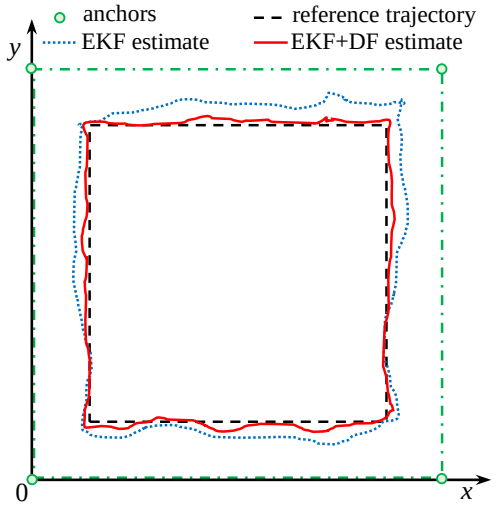


Figure 6.26: *Proposed experiment following a square path. It is expected the EKF+DF estimation (continuous red line) to be more accurate than the EKF only estimation (dotted blue line). The drone is forced to move linearly on four rails (reference black square).*

6.5 Results and Discussion

6.5.1 Proof of accuracy improvement

The improvement to the accuracy of the positioning system given by the use of the proposed debiasing filter is proven by performing the validation experiment described in Section 6.4.2, which results are depicted in Figures 6.27-6.30. In order to highlight the overall accuracy gain using the DF, the absolute value of the bias is represented. Please note that the calibration points are on the main grid spaced 50 cm, while the validation mesh is staggered by 25 cm from the main calibration points.

Accordingly to the colour legend used in Figures 6.27-6.30, the whiter the area the more accurate it is. Hence it is really evident the contribution of the proposed DF. Although, it is noticeable that in very few points the DF failed to improve the accuracy in the validation positions. This means that the sampling points used for the mapping did not capture that gradient of the bias. A finer sampling mesh most likely would have discovered that trend, however a compromise between mapping refinement and complexity of the interpolating RBFN (which would result in longer computational time, detrimental for real time application) must be made. Moreover, another interest-

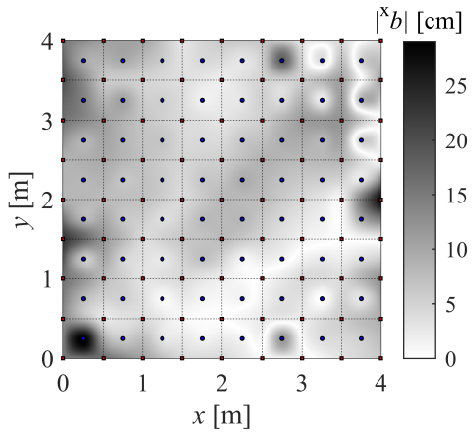


Figure 6.27: *Original (only EKF) absolute bias for x-direction measurements.*

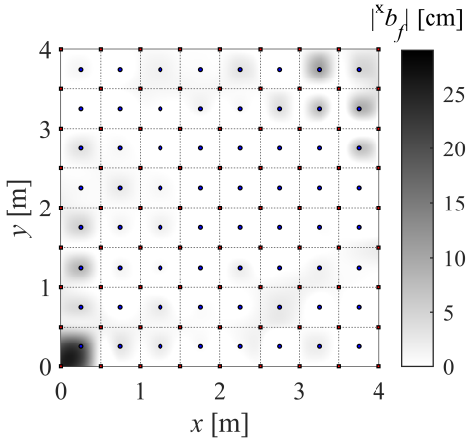


Figure 6.28: *Absolute bias for x-direction measurements after applying debiasing filter (DF).*

ing aspect of the previously carried theoretical analysis is manifest around the anchor positioned at (0,0) m. In fact, the points within approximately 50 cm radius around this anchor are undefined because those locations fall within the bifurcation envelope, explained in Section 6.2.6. Therefore, no position can be measured in this area and the DF is justified to fail.

6.5.2 Debiasing dynamic validation

In this section all the results related to the dynamic experiments explained in Sections 6.4.3 and 6.4.4 are collected and investigated. I expected a reduction in the performance of the DF, because of dynamic effects intrinsic of the used positioning algorithm that are not addressed by the DF. Still, even if slightly, the formulated DF allows more accurate measurement of the position, as depicted in Tables 6.1 and 6.2. More precisely the first table refers to the dynamic validation of the DF explained in Section 6.4.3, while the statistics of the results of the square path experiment (Section 6.4.4) are collected in Table 6.2.

The following Figure 6.31 is an example of a single dynamic experiment in x-direction. Such partial experiment has been repeated 10 times for each rail position in order to obtain a general trend of the IPS measurements.

Notice in Figure 6.31 the dynamic 'misbehaviour' of the IPS at around 22 s which cannot be addressed by the presented DF but which could be solved by the use of other filters presented in section 6.3.

Follows in Figure 6.32 a graphical representation of the overall results of the square-path experiment, and in Table 6.2 a quantitative summary.

The root-mean-square-errors (RMSEs) in Table 6.2 are computed comparing the trend lines for each edge with the actual position of the drone on the rail at every

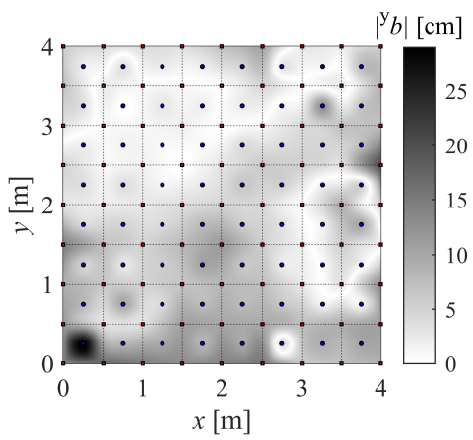


Figure 6.29: *Original (only EKF) absolute bias for y-direction measurements.*

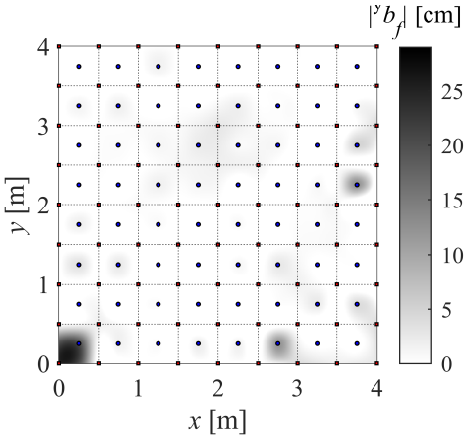


Figure 6.30: *Absolute bias for y-direction measurements after applying debiasing filter (DF).*

time step. For instance, the RMSE_{EKF} on the bot edge is obtained considering the cloud of data (cyan colour in Figure 6.32) of ten experiments on the edge going from (0.5,0.5) to (3.5,0.5). The same dynamic issues that where pointed out in the validation experiment in Figure 6.31 persist in the square experiment (yellow regions in Figure 6.32). Hence, believing that DF is not responsible to address this intrinsic problems of the studied IPS, I thought it would be useful to isolate this misbehaviour and give statistics of a selected subdomain of data that were not affected by this. Hence the *raw* and *sel.* column headings in Table 6.2.

6.6 Conclusion

Considering the recent drive of robotics towards autonomy and self-organisation, the precision and accuracy of Indoor Positioning Systems (IPS) are crucial for performing indoor experiments efficiently and safely. While the precision of the system is generally well studied, reasonably estimated, and provided by the manufacturer, accuracy tends to be assessed poorly if not plainly disregarded. This may not pose a major problem when flying one small inexpensive drone – or even a few of them in formation flight – but would certainly render the system not operational for a swarm of frenetic self-organising drones which may end up bumping into one another. In this chapter about IPS, beside a comprehensive study of precision, accuracy and failure of IPSs, a debiasing filter was developed aiming to improve the accuracy. The proposed analysis is generalised and repeatable for any other IPS that uses UWB technology and multilateration algorithm based on the TDoA signal property. The numerical values that follow in this section correspond to the specific IPS used for the validation and testing experiments.

The measured distribution and magnitude of the precision are coherent with their

			$\text{RMSE}_{x,\text{avg}}$ [cm]			$\text{RMSE}_{y,\text{avg}}$ [cm]			
dir.	x [m]	y [m]	IPS-1	IPS-2	$\boldsymbol{\Delta x}$	IPS-1	IPS-2	$\boldsymbol{\Delta y}$	v_{avg}
hor.	[0, 4]	1	12.7	6.8	**5.9**	10.0	7.9	**2.1**	0.58
hor.	[0, 4]	2	12.0	8.1	**3.9**	6.7	4.3	**2.4**	0.44
hor.	[0, 4]	3	12.6	8.0	**4.6**	9.3	8.0	**1.3**	0.43
ver.	1	[4, 0]	15.6	10.3	**5.4**	9.4	6.8	**2.7**	0.58
ver.	2	[4, 0]	10.3	8.0	**2.3**	15.8	10.1	**5.7**	0.51
ver.	3	[4, 0]	11.4	9.4	**2.0**	15.3	12.2	**3.1**	0.42

Table 6.1: *Representative results of the dynamic on-rail validation. In the column headed 'dir.' is highlighted if the rail was oriented horizontally (hor.) or vertically (ver.). The path was always cross-cutting the overall flying domain left-to-right or top-to-bottom. The RMSEs of an IPS with (IPS-2) and without (IPS-1) debiasing are compared in order to depict the improvement given by the DF, which average value is stored in the columns with Δx and Δy headings. The average cruise velocity is also registered in the last column.*

				$\text{RMSE}_{\text{IPS-1}}$ [cm]		$\text{RMSE}_{\text{IPS-2}}$ [cm]		
edge	dir.	x [m]	y [m]	raw	sel.	raw	sel.	$\boldsymbol{\Delta}$ [cm]
bot	hor.	[0.5, 3.5]	0.5	9.2	9.5	7.5	4.7	**4.8**
right	ver.	3.5	[0.5, 3.5]	12.6	12.5	9.0	8.3	**4.2**
top	hor.	[3.5, 0.5]	3.5	6.0	5.5	5.7	4.6	**0.9**
left	ver.	0.5	[3.5, 0.5]	15.8	15.2	8.8	6.7	**8.5**

Table 6.2: *Representative results of the square-path experiment. The RMSEs of an IPS with and without debiasing are compared in order to depict the improvement given by the filter. The columns with 'raw' heading refer to the overall stream of data, while the 'sel.' heading refers to the selection of the undamaged stream of data - e.g. neglecting intervals where uncontrolled misbehaviour of the IPS are detected. Δ gives an idea of the average improvement of the accuracy given by the use DF.*

theoretical estimations – i.e. using Cramér–Rao Lower Bound analysis, the best precision values are in the centre of the flying area and the maximum standard deviation is lower than ±3 cm (figures 6.17-6.18). The IPS's failure study – i.e. the formulation of the bifurcation envelope – reflects in the experimental measurements as undefined positioning areas in the anchor's proximity (Figures 6.27-6.30).

As far as the formulated debiasing filter is considered, its static use – e.g. when the drone is hovering – shows a dramatic improvement of the accuracy: from an original worst case of ±20 cm to a corrected one of about ±5 cm. Testing such debiasing filter dynamically – i.e. moving the drone – still shows a substantial improvement of the accuracy, but not as much as expected: a maximum accuracy gain of about 5.5 cm (tables 6.1-6.2). Other dynamic issues occur, that should be addressed with additional filters complementarily to the proposed one. A possible filtering process

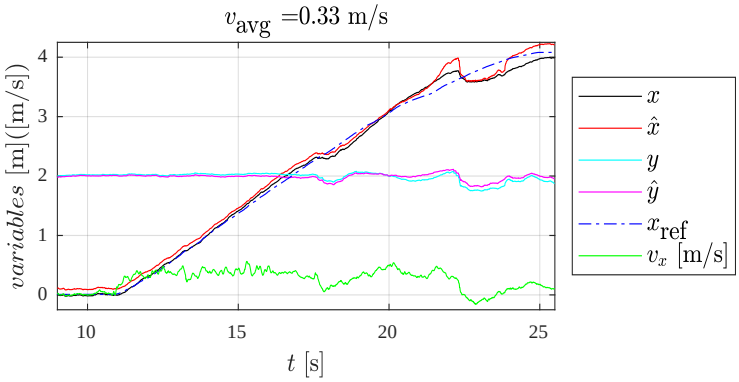


Figure 6.31: *Example of horizontal dynamic experiment performed at average cruise velocity of 0.33 m/s with x spanning from 0 m to 4 m at constant $y = 2$ m. The variables are: the poistion estimate with only EKF (x, y); the debiased position $(\hat{x}, \hat{y})$; the actual position on the rail (x_{ref}); and the estimated instantaneous velocity in x-direction (v_x).*

has been suggested consisting of the following steps to be taken after obtaining a first raw estimation of the position given by the Extended-Kalman-Filter: saturation and smoothing, correction by the use of a prediction model (such as Adams-Moulton), and finally the proposed debiasing filter. The reproduction of a previous reference experiment [114], consisting of flying a drone along a square path, proves the global improvement of the accuracy given by the use of the proposed debiasing filter also for dynamic case (figure 6.32).

Further extension of the presented research might involve: the generalisation of the introduced rigorous precision, accuracy, and failure analysis for a three-dimensional IPS; the formulation and validation of the proposed debiasing filter in 3D and automation of the process; the formulation and testing of further filtering algorithms to cope with the dynamic effects on the position estimation; and the optimisation of the positioning and orientation of the anchors in 3D availing of the slight anisotropy of the radiation pattern of the UWB module.

6.7 Summary

Scope: Contribute, with theoretical and experimental research, to the development of a Swarm-in-the-Loop system for testing self-organisation algorithms for swarms. The focus of this Chapter was on Indoor Positioning Systems (IPS), which is a crucial technology for indoor experiments in swarm robotics.
The developed IPS ought to have the following properties:

- *precise* and *accurate*, with also good understanding of its *failure* condition areas;
- *passing through solid obstacles*: the chosen technology is UWB-based (Ultra

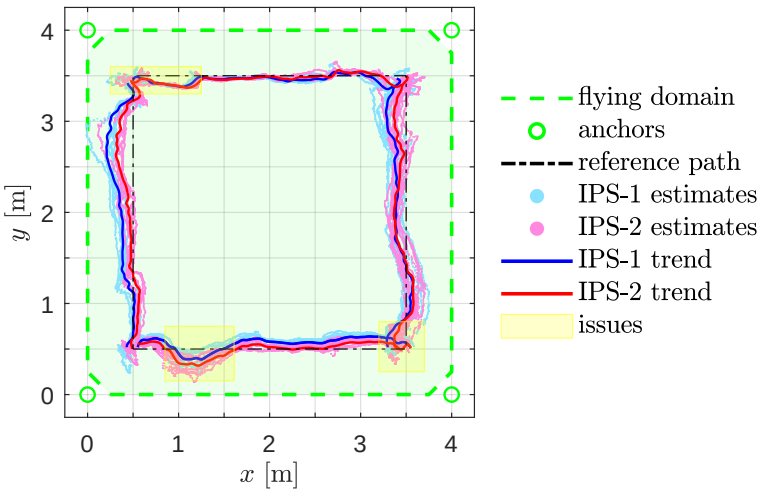


Figure 6.32: *Visualisation of the square-path experiment results. The flying domain is delimited by the four anchors while the reference path is the black dotted square. The drone starts moving from the corner located at (0.5,0.5) and moves towards the positive x-axis direction then following the other edges of the reference path. Two clouds of data with their respective trend lines are shown: IPS-1 which only uses EKF and the IPS-2 embedding also the DF. The yellow transparent areas highlight the problematic phases. The overal experiment shows a time-to-time considerable improvement given by the DF.*

Wideband);

- indoor system *analogous to GNSSs* (such as GPS): considered to use a Time Difference of Arrival (TDoA) algorithm, possibly asynchronous.

Method, studies, and experiments:

- *Cramér–Rao Lower Bound analysis* (CRLB) to estimate numerically the precision of the system at hand.
- Define geometrically the *bifurcation envelope* which consists in the areas where the IPS fails to provide an unambiguous position measurement.
- Rapid prototyping and *assembly* of the IPS components. *Writing the software* in the drones firmware for three purposes: the filter, the live debugging, and the data collection for the experiments.

- *Calibration*: map the original *accuracy of the IPS measurements* and *formulate the debiasing filter* as modified (by the local standard deviation) de-bias values on a deformed grid.
- Perform a *static test* of the debiasing filter on a staggered grid (i.e. not on the points used for calibration).
- Perform *dynamic experiments* to assess the performance of the developed filter when the drone is moving.

Main outcomes and findings:

- A rigorous and comprehensive *study of precision, accuracy* and *failure* of 2D UWB-based IPS using TDoA algorithm. This could be used in future for optimisation of the positioning and orientation of the anchors (i.e. the IPS antennae) availing of the slight anisotropy of the radiation pattern of their UWB module.
- The measured IPS precision is in agreement with the theoretical CRLB analysis
- A *debiasing filter* aimed at improving the accuracy of the used IPS. This was placed right after the Extended Kalman Filter, which was already part of the firmware of the used drones. It performs outstandingly statically, while its improvement on accuracy deteriorates during dynamic experiments.
- The IPS study and the debiasing filter should be generalised to 3D during further future research.
- An in-house IPS to be used for SwiL simulations.
- The the TDoA-based multilateration can be used to create a "local positioning system" (LPS) outdoor. Its scope would be to improve the GPS estimations in case of obstruction of the signal by large solid objects.

Chapter 7

Conclusion and Final Remarks

In this chapter, concluding remarks are provided about what has been achieved for the four main topics of this thesis: wildfire propagation modelling and simulation, self-organising firefighting drones modelling and simulation, stigmergic collision-avoidance algorithm, and analysis and improvement of indoor positioning systems (IPSs). Moreover, in Figure 7.1 a summarising diagram highlights the various research topics that I carried out, along with other intended necessary sections, in order to explain how they relate to the grand firefighting scheme: I believe this could help in clarifying the links between studies, presented in the previous chapters, that might seem so distant at a first look but that all come together in this picture. Two main projects represent the steps to prove the employability of swarms of drones to fight wildfires: the first consists in the development of a *Swarm-in-the-Loop* (SwiL) *system* to test self-organisation algorithms and firefighting strategies in a safe and reasonably affordable indoor environment; while the second is the *"real world" test platform* for validating the swarm algorithm over a multitude of firefighting missions on *prescribed fires*. It is important to note that, even though the swarm's main mission is to suppress wildfires, their self-organisation can be leveraged in order to perform other missions, for instance: wildfire mapping and monitoring, fire perimetre containment, and aiding evacuation of endangered population. Moreover, even though this research is all about firefighting swarms, it should not be misinterpreted as an attempt to suppress any wildfire on our planet and prevent any new ones from happening: if this would be even possible, it would have a detrimental impact on the worldwide ecosystem. In fact, wildfires act as oxygen-level controllers as supported by a fairly recent study on the relation between terrestrial ecosystems, fires and oxygen levels [116]; therefore, not all wildfires are intrinsically bad. Hence, the proposed technology should be used sparingly and thoughtfully evaluating the importance of the eventual ecosystem the fire is about to destroy and the impact on the local population's lives.

The SwiL system (Figure 7.1) consists of three main subsystems: *wildfire dynamics*, *swarm dynamics*, and *communication and monitoring*. Within these, for each component, I highlighted the major wanted properties and the main outcomes of the related research. It is important to notice that, only four main blocks communicate in order

to form the *wildfire suppression model* which is the firefighting solution that ought to be tested in the SwiL. Such components work in a loop:

1. the *wildfire propagation* model (FireProM-F) initialises the simulated fire in a number of predefined or random points and the fire starts spreading;

2. a *fire detection* model senses the wildfire and provides information to the swarm accordingly to the selected technology (in this research case, the technology was infrared camera, so the information transmitted was simply a temperature field);

3. each drone in the *firefighting swarm* acts on their memory of the received information (e.g. the best temperature experience could be an attractor for the decision of the next targeted position);

4. the swarm acts on the simulated fire through a *fire extinction* model, that for simplicity in this research has been considered to be water dropping (hence the two main payloads of the drones were fuel and water);

5. the *wildfire propagation* model evolves in time considering the impact of the extinction model (then restarts from point 2).

The virtual and physical interfaces between these sub-models are handled by the server, which covers also a monitoring function. The idea was to set up a Gazebo/ROS platform that would be used for both SiL simulations, in which the drones are simulated, and SwiL simulations, where the drones are real hardware in the loop. In case of the latter, such platform would be used only for two purposes: a virtual representation of what is really happening in the laboratory (e.g. visualising the position of each drone in the flying domain), and communication between the FireProM-F simulation and the real swarm dynamics. Other three blocks can be seen in the swarm dynamics subsystem in the diagram 7.1: multi sub-swarms, collision-avoidance algorithm, and indoor positioning system (IPS). The first two are improvements of the core firefighting algorithm for the swarm, while the IPS is a necessary tool for indoor swarm robotics. During the presented research I managed to contribute on the last two components which were quite critical for the advancement of the research towards SwiL simulations. I find that it is important to mention that the developed multi-agent collision-avoidance algorithm is stigmergic, which means it relies on indirect communication between the drones by means of modifying some properties of the surrounding environment - e.g. in this case transmitting a signal at constant strength. As far as the IPS is considered, I managed to develop a debiasing filter which managed to considerably improve the accuracy of the position estimations. Even though the IPS is an indoor technology, the same principle can be used to create outdoor a "local positioning system" (LPS) to improve the precision and accuracy of the GPS estimation in case of signal obstruction - e.g. in proximity of a building or a hill-side, or when flying in canyons. In such LPS some reference drones could actually stay stationary and act as reference antennae for the TDoA-based multilateration.

Concluding, the research presented in this thesis is just the "tip of the iceberg" of a larger multidisciplinary applied research which will eventually enable an exciting wildfire suppression technology. I passionately believe this will be part of our highly technological future where autonomy is the engine of most of our daily lives.

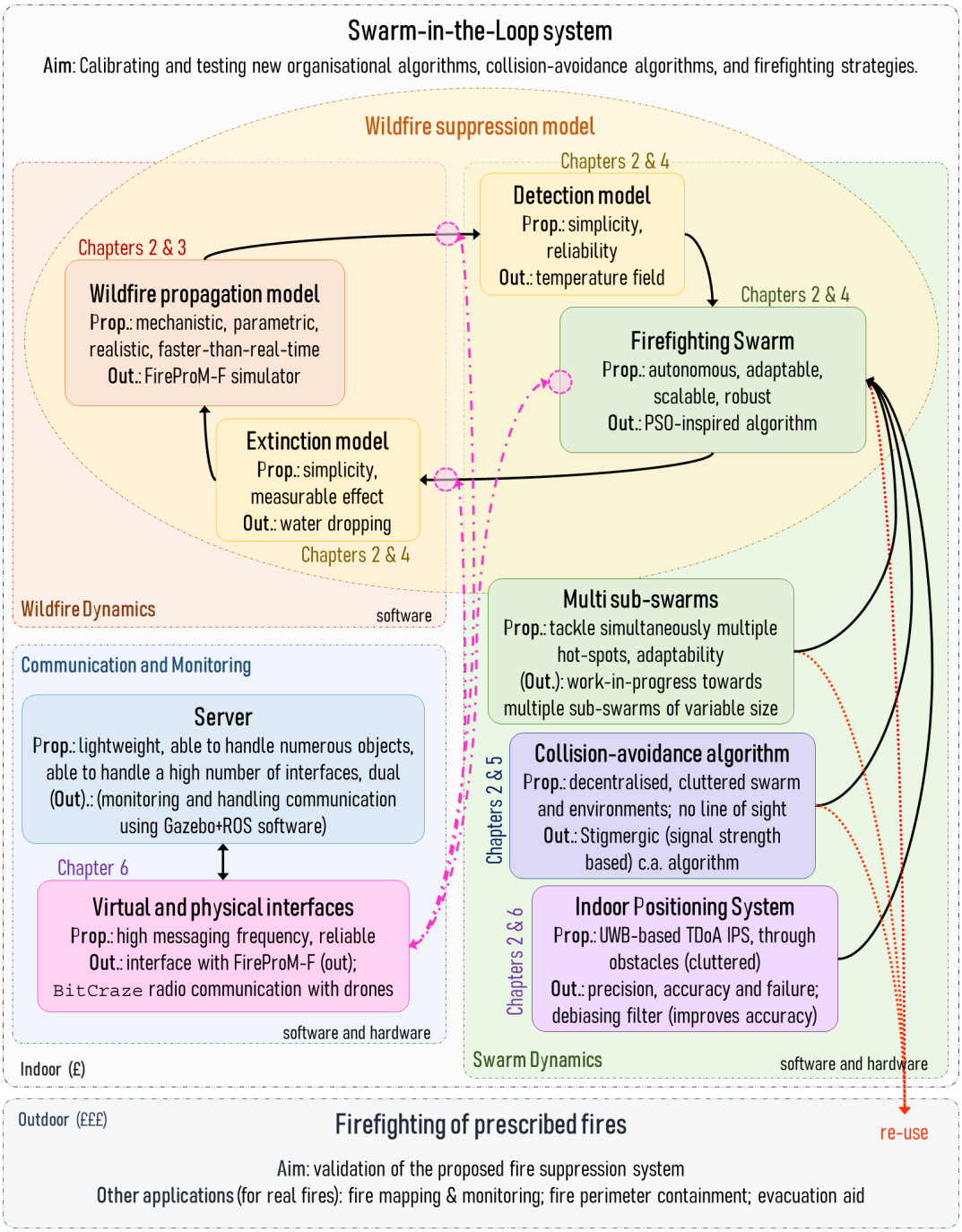


Figure 7.1: *Summarising diagram of the undergone research and how its various topics relate to the grand scheme. "**Prop.**" stands for a list of desirable and attained properties; "**Out.**" for delivered outcomes; "**(Out.)**" for desired but not/partially delivered outcomes; and (£) means reasonably affordable while (£££) is very costly.*

Bibliography

[1] M. S. Innocente and P. Grasso, "Self-organising swarms of firefighting drones: Harnessing the power of collective intelligence in decentralised multi-robot systems," *Journal of Computational Science*, vol. 34, pp. 80–101, may 2019.

[2] *Building wildfire resilience into forest management planning.* Edinburgh: Forestry Commission, 2014.

[3] G. Perry, "Current approaches to modelling the spread of wildland fire: a review," *Progress in Physical Geography*, vol. 22, no. 2, pp. 222–245, 1998.

[4] A. L. Sullivan, "Wildland surface fire spread modelling, 1990–2007. 1: Physical and quasi-physical models," *International Journal of Wildland Fire*, vol. 18, no. 4, p. 349, 2009.

[5] ——, "Wildland surface fire spread modelling, 1990–2007. 2: Empirical and quasi-empirical models," *International Journal of Wildland Fire*, vol. 18, no. 4, p. 369, 2009.

[6] ——, "Wildland surface fire spread modelling, 1990–2007. 3: Simulation and mathematical analogue models," *International Journal of Wildland Fire*, vol. 18, no. 4, p. 387, 2009.

[7] P. Grasso and M. S. Innocente, "Physics-based model of wildfire propagation towards faster-than-real-time simulations," *Computers & Mathematics with Applications*, vol. 80, no. 5, pp. 790–808, sep 2020.

[8] R. C. Rothermel, "A mathematical model for predicting fire spread in wildland fuels." 1972.

[9] P. L. Andrews, "Current status and future needs of the BehavePlus fire modeling system," *International Journal of Wildland Fire*, vol. 23, no. 1, p. 21, 2014.

[10] M. A. Finney, "Farsite: Fire area simulator-model development and evaluation," 1998.

[11] P. L. Andrews, "The rothermel surface fire spread model and associated developments: A comprehensive explanation," *Gen. Tech. Rep. RMRS-GTR-371. Fort*

Collins, CO: US Department of Agriculture, Forest Service, Rocky Mountain Research Station. 121 p., vol. 371, 2018.

[12] R. Linn, "A transport model for prediction of wildfire behavior," Tech. Rep., jul 1997.

[13] R. Linn, J. Reisner, J. J. Colman, and J. Winterkamp, "Studying wildfire behavior using FIRETEC," *International Journal of Wildland Fire*, vol. 11, no. 4, p. 233, 2002.

[14] O. Séro-Guillaume and J. Margerit, "Modelling forest fires. part I: a complete set of equations derived by extended irreversible thermodynamics," *International Journal of Heat and Mass Transfer*, vol. 45, no. 8, pp. 1705–1722, apr 2002.

[15] K. B. McGrattan, "Fire dynamics simulator technical reference guide volume 1: Mathematical model [6th ed.]," Tech. Rep., 2006.

[16] W. Mell, M. A. Jenkins, J. Gould, and P. Cheney, "A physics-based approach to modelling grassland fires," *International Journal of Wildland Fire*, vol. 16, no. 1, p. 1, 2007.

[17] L. Ferragut, M. Asensio, and S. Monedero, "A numerical method for solving convection–reaction–diffusion multivalued equations in fire spread modelling," *Advances in Engineering Software*, vol. 38, no. 6, pp. 366–371, jun 2007.

[18] J. Margerit and O. Séro-Guillaume, "Modelling forest fires. part II: reduction to two-dimensional models and simulation of propagation," *International Journal of Heat and Mass Transfer*, vol. 45, no. 8, pp. 1723–1737, apr 2002.

[19] P. Grasso and M. S. Innocente, "A two-dimensional reaction-advection-diffusion model of the spread of fire in wildlands," in *Advances in forest fire research 2018*. Imprensa da Universidade de Coimbra, 2018, pp. 334–342.

[20] O. Séro-Guillaume, S. Ramezani, J. Margerit, and D. Calogine, "On large scale forest fires propagation models," *International Journal of Thermal Sciences*, vol. 47, no. 6, pp. 680–694, jun 2008.

[21] V. G. Ntinas, B. E. Moutafis, G. A. Trunfio, and G. C. Sirakoulis, "Parallel fuzzy cellular automata for data-driven simulation of wildfire spreading," *Journal of Computational Science*, vol. 21, pp. 469–485, jul 2017.

[22] L. H. Encinas, S. H. White, A. M. del Rey, and G. R. Sánchez, "Modelling forest fire spread using hexagonal cellular automata," *Applied Mathematical Modelling*, vol. 31, no. 6, pp. 1213–1227, jun 2007.

[23] C. Li, J. Li, L. Hu, and D. Hou, "Visualization and simulation model of underground mine fire disaster based on cellular automata," *Applied Mathematical Modelling*, vol. 39, no. 15, pp. 4351–4364, aug 2015.

[24] X. Rui, S. Hui, X. Yu, G. Zhang, and B. Wu, "Forest fire spread simulation algorithm based on cellular automata," *Natural Hazards*, vol. 91, no. 1, pp. 309–319, nov 2017.

[25] N. Fernandez-Anez, K. Christensen, and G. Rein, "Two-dimensional model of smouldering combustion using multi-layer cellular automaton: The role of ignition location and direction of airflow," *Fire Safety Journal*, vol. 91, pp. 243–251, jul 2017.

[26] R. M. Almeida and E. E. N. Macau, "Stochastic cellular automata model for wildland fire spread dynamics," *Journal of Physics: Conference Series*, vol. 285, p. 012038, mar 2011.

[27] H. Gazmeh, A. Alesheikh, and M. K. 1, "A new methodology in modeling forest fire spread using cellular automata," *Journal of Advanced Science and Engineering Research*, 2012.

[28] K. Yamamoto, N. Takada, and M. Misawa, "Combustion simulation with lattice boltzmann method in a three-dimensional porous structure," *Proceedings of the Combustion Institute*, vol. 30, no. 1, pp. 1509–1515, jan 2005.

[29] B. Porterie, N. Zekri, J.-P. Clerc, and J.-C. Loraud, "Modeling forest fire spread and spotting process with small world networks," *Combustion and Flame*, vol. 149, no. 1-2, pp. 63–78, apr 2007.

[30] A. Kacem, C. Lallemand, N. Giraud, M. Mense, M. D. Gennaro, Y. Pizzo, J.-C. Loraud, P. Boulet, and B. Porterie, "A small-world network model for the simulation of fire spread onboard naval vessels," *Fire Safety Journal*, vol. 91, pp. 441–450, jul 2017.

[31] S. Osher and J. A. Sethian, "Fronts propagating with curvature-dependent speed: Algorithms based on hamilton-jacobi formulations," *Journal of Computational Physics*, vol. 79, no. 1, pp. 12–49, nov 1988.

[32] V. Mallet, D. Keyes, and F. Fendell, "Modeling wildland fire propagation with level set methods," *Computers & Mathematics with Applications*, vol. 57, no. 7, pp. 1089–1101, apr 2009.

[33] A. S. Bova, W. E. Mell, and C. M. Hoffman, "A comparison of level set and marker methods for the simulation of wildland fire front propagation," *International Journal of Wildland Fire*, vol. 25, no. 2, p. 229, 2016.

[34] C. Tymstra, "Development and structure of prometheus : the canadian wildland fire growth simulation model," Information report NOR-X-417, 2010.

[35] K. Tolhurst, B. Shields, and D. Chong, "Phoenix: development and application of a bushfire risk management tool." *The Australian Journal of Emergency Management*, 2008.

[36] K. B. McGrattan and G. P. Forney, "Fire dynamics simulator user's guide (version 4)," Tech. Rep., 2004.

[37] D. Casbeer, S.-M. Li, R. Beard, R. Mehra, and T. McLain, "Forest fire monitoring with multiple small UAVs," in *Proceedings of the 2005, American Control Conference, 2005.* IEEE, 2005.

[38] C. Torresan, A. Berton, F. Carotenuto, S. F. D. Gennaro, B. Gioli, A. Matese, F. Miglietta, C. Vagnoli, A. Zaldei, and L. Wallace, "Forestry applications of UAVs in europe: a review," *International Journal of Remote Sensing*, vol. 38, no. 8-10, pp. 2427–2447, nov 2016.

[39] R. Graml and G. Wigley, "Bushfire hotspot detection through uninhabited aerial vehicles and reconfigurable computing," in *2008 IEEE Aerospace Conference.* IEEE, mar 2008.

[40] C. Yuan, Y. Zhang, and Z. 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, pp. 783–792, jul 2015.

[41] E. Şahin, S. Girgin, L. Bayindir, and A. E. Turgut, "Swarm robotics," in *Natural Computing Series.* Springer Berlin Heidelberg, pp. 87–100.

[42] E. Bonabeau, "Agent-based modeling: Methods and techniques for simulating human systems," *Proceedings of the National Academy of Sciences*, vol. 99, no. Supplement 3, pp. 7280–7287, may 2002.

[43] E. Şahin, "Swarm robotics: From sources of inspiration to domains of application," in *Swarm Robotics.* Springer Berlin Heidelberg, 2005, pp. 10–20.

[44] C. W. Reynolds, "Flocks, herds and schools: A distributed behavioral model," *ACM SIGGRAPH Computer Graphics*, vol. 21, no. 4, pp. 25–34, aug 1987.

[45] H. F., "A stochastic nonlinear model for coordinated bird flocks," *The Ubiquity of Chaos, AAAS Publications*, no. 10018343671, 1990. [Online]. Available: https://ci.nii.ac.jp/naid/10018343671/en/

[46] J. Kennedy, "Swarm intelligence," in *Handbook of Nature-Inspired and Innovative Computing.* Kluwer Academic Publishers, 2006, pp. 187–219.

[47] B. M. Albaker and N. A. Rahim, "Unmanned aircraft collision detection and resolution: Concept and survey," in *2010 5th IEEE Conference on Industrial Electronics and Applications*. IEEE, jun 2010.

[48] J. N. Yasin, S. A. S. Mohamed, M.-H. Haghbayan, J. Heikkonen, H. Tenhunen, and J. Plosila, "Navigation of autonomous swarm of drones using translational coordinates," in *Advances in Practical Applications of Agents, Multi-Agent Systems, and Trustworthiness. The PAAMS Collection*. Springer International Publishing, 2020, pp. 353–362. [Online]. Available: 10.1007/978-3-030-49778-1_-28

[49] J. van den Berg, M. Lin, and D. Manocha, "Reciprocal velocity obstacles for real-time multi-agent navigation," in *2008 IEEE International Conference on Robotics and Automation*. IEEE, may 2008.

[50] R. Regele and P. Levi, "Cooperative multi-robot path planning by heuristic priority adjustment," in *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, oct 2006.

[51] C. Blesing, D. Luensch, J. Stenzel, and B. Korth, "Concept of a multi-agent based decentralized production system for the automotive industry," in *Lecture Notes in Computer Science*. Springer International Publishing, 2017, pp. 19–30. [Online]. Available: 10.1007/978-3-319-59930-4_2

[52] A. W. ter Mors, "Conflict-free route planning in dynamic environments," in *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, sep 2011.

[53] F. B. von der Osten, M. Kirley, and T. Miller, "Anticipatory stigmergic collision avoidance under noise," in *Proceedings of the 2014 Annual Conference on Genetic and Evolutionary Computation*. ACM, jul 2014.

[54] X. Zhang, A. Liniger, and F. Borrelli, "Optimization-based collision avoidance," Nov. 2017.

[55] O. Khatib, "Real-time obstacle avoidance for manipulators and mobile robots," in *Proceedings. 1985 IEEE International Conference on Robotics and Automation*. Institute of Electrical and Electronics Engineers.

[56] J.-O. Kim and P. Khosla, "Real-time obstacle avoidance using harmonic potential functions," *IEEE Transactions on Robotics and Automation*, vol. 8, no. 3, pp. 338–349, jun 1992.

[57] D. E. Chang, S. Shadden, J. Marsden, and R. Olfati-Saber, "Collision avoidance for multiple agent systems," in *42nd IEEE International Conference on Decision and Control (IEEE Cat. No.03CH37475)*. IEEE.

[58] A. Soriano, E. J. Bernabeu, A. Valera, and M. Vallés, "Multi-agent systems platform for mobile robots collision avoidance," in *Advances on Practical Applications of Agents and Multi-Agent Systems*. Springer Berlin Heidelberg, 2013, pp. 320–323. [Online]. Available: 10.1007/978-3-642-38073-0_37

[59] D. Wang, T. Fan, T. Han, and J. Pan, "A two-stage reinforcement learning approach for multi-UAV collision avoidance under imperfect sensing," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 3098–3105, apr 2020.

[60] W. Sakpere, M. A. Oshin, and N. B. Mlitwa, "A state-of-the-art survey of indoor positioning and navigation systems and technologies," *South African Computer Journal*, vol. 29, no. 3, dec 2017.

[61] D. Chen, K. Neusypin, M. Selezneva, and Z. Mu, "New algorithms for autonomous inertial navigation systems correction with precession angle sensors in aircrafts," *Sensors*, vol. 19, no. 22, p. 5016, nov 2019.

[62] H. Rong, Y. Gao, L. Guan, Q. Zhang, F. Zhang, and N. Li, "GAM-based mooring alignment for SINS based on an improved CEEMD denoising method," *Sensors*, vol. 19, no. 16, p. 3564, aug 2019.

[63] S. Widodo, T. Shiigi, N. Hayashi, H. Kikuchi, K. Yanagida, Y. Nakatsuchi, Y. Ogawa, and N. Kondo, "Moving object localization using sound-based positioning system with doppler shift compensation," *Robotics*, vol. 2, no. 2, pp. 36–53, apr 2013.

[64] D. J. Schott, A. Saphala, G. Fischer, W. Xiong, A. Gabbrielli, J. Bordoy, F. Höflinger, K. Fischer, C. Schindelhauer, and S. J. Rupitsch, "Comparison of direct intersection and sonogram methods for acoustic indoor localization of persons," *Sensors*, vol. 21, no. 13, p. 4465, jun 2021.

[65] D. Arbula and S. Ljubic, "Indoor localization based on infrared angle of arrival sensor network," *Sensors*, vol. 20, no. 21, p. 6278, nov 2020.

[66] A. A. Mahmoud, Z. U. Ahmad, O. C. Haas, and S. Rajbhandari, "Precision indoor three-dimensional visible light positioning using receiver diversity and multi-layer perceptron neural network," *IET Optoelectronics*, vol. 14, no. 6, pp. 440–446, dec 2020.

[67] A. Alarifi, A. Al-Salman, M. Alsaleh, A. Alnafessah, S. Al-Hadhrami, M. Al-Ammar, and H. Al-Khalifa, "Ultra wideband indoor positioning technologies: Analysis and recent advances," *Sensors*, vol. 16, no. 5, p. 707, may 2016.

[68] Y. P. Tsang, C.-H. Wu, W. Ip, G. Ho, and M. Tse, "A bluetooth-based indoor positioning system: a simple and rapid approach," *Annual Journal IIE (HK)*, vol. 35, pp. 11–26, 10 2015.

[69] B. Ezhumalai, M. Song, and K. Park, "An efficient indoor positioning method based on wi-fi RSS fingerprint and classification algorithm," *Sensors*, vol. 21, no. 10, p. 3418, may 2021.

[70] A. Abusara, M. S. Hassan, and M. H. Ismail, "Reduced-complexity fingerprinting in WLAN-based indoor positioning," *Telecommunication Systems*, vol. 65, no. 3, pp. 407–417, oct 2016.

[71] H. Sahota and R. Kumar, "Sensor localization using time of arrival measurements in a multi-media and multi-path application of in-situ wireless soil sensing," *Inventions*, vol. 6, no. 1, p. 16, feb 2021.

[72] M. Compagnoni and R. Notari, "Tdoa-based localization in two dimensions: the bifurcation curve," *Fundamenta Informaticae*, vol. 135, no. 1-2, pp. 199–210, 2014.

[73] L. A. M. Hernandez, S. P. Arteaga, G. S. Perez, A. L. S. Orozco, and L. J. G. Villalba, "Outdoor location of mobile devices using trilateration algorithms for emergency services," *IEEE Access*, vol. 7, pp. 52 052–52 059, 2019.

[74] M. F. Mosleh, M. J. Zaiter, and A. H. Hashim, "Position estimation using trilateration based on ToA/RSS and AoA measurement," *Journal of Physics: Conference Series*, vol. 1773, no. 1, p. 012002, feb 2021.

[75] D. Neirynck, E. Luk, and M. McLaughlin, "An alternative double-sided two-way ranging method," in *2016 13th Workshop on Positioning, Navigation and Communications (WPNC)*. IEEE, oct 2016.

[76] F. Jamil, N. Iqbal, S. Ahmad, and D.-H. Kim, "Toward accurate position estimation using learning to prediction algorithm in indoor navigation," *Sensors*, vol. 20, no. 16, p. 4410, aug 2020.

[77] P. Mahida, S. Shahrestani, and H. Cheung, "Deep learning-based positioning of visually impaired people in indoor environments," *Sensors*, vol. 20, no. 21, p. 6238, oct 2020.

[78] *Galileo Initial Services*, European Union Agency for the Space Programme, May 2021. [Online]. Available: https://www.euspa.europa.eu/european-space/galileo/services/initial-services

[79] S.-C. Yeh, W.-H. Hsu, M.-Y. Su, C.-H. Chen, and K.-H. Liu, "A study on outdoor positioning technology using GPS and WiFi networks," in *2009 International Conference on Networking, Sensing and Control*. IEEE, mar 2009.

[80] M. Ghavami, L. Michael, and R. Kohno, *Ultra Wideband Signals and Systems in Communication Engineering*. John Wiley & Sons, Feb. 2007.

[81] F. Orjales, J. Losada-Pita, A. Paz-Lopez, and Á. Deibe, "Towards precise positioning and movement of UAVs for near-wall tasks in GNSS-denied environments," *Sensors*, vol. 21, no. 6, p. 2194, mar 2021.

[82] H. Cramér, *Mathematical Methods of Statistics (PMS-9), Volume 9*. Princeton University Press, Apr. 1999. [Online]. Available: https://www.ebook.de/de/product/3646411/harald_cramer_mathematical_methods_of_statistics_pms_9_volume_9.html

[83] C.-S. Chen, Y.-J. Chiu, C.-T. Lee, and J.-M. Lin, "Calculation of weighted geometric dilution of precision," *Journal of Applied Mathematics*, vol. 2013, pp. 1–10, 2013.

[84] M. Compagnoni, R. Notari, F. Antonacci, and A. Sarti, "A comprehensive analysis of the geometry of TDOA maps in localization problems," *Inverse Problems*, vol. 30, no. 3, p. 035004, feb 2014.

[85] S. O. Dulman, A. Baggio, P. J. Havinga, and K. G. Langendoen, "A geometrical perspective on localization," in *Proceedings of the first ACM international workshop on Mobile entity localization and tracking in GPS-less environments - MELT '08*. ACM Press, 2008.

[86] B. J. Mcbride, S. Gordon, and M. A. Reno, "Coefficients for calculating thermodynamic and transport properties of individual species," Tech. Rep., 1993.

[87] K. Ragland, D. Aerts, and A. Baker, "Properties of wood for combustion analysis," *Bioresource Technology*, vol. 37, no. 2, pp. 161–168, jan 1991.

[88] A. W. Cook, "Enthalpy diffusion in multicomponent flows," *Physics of Fluids*, vol. 21, no. 5, p. 055109, may 2009.

[89] S. Mathur and S. C. Saxena, "Relations between thermal conductivity and diffusion coefficients of pure and mixed polyatomic gases," *Proceedings of the Physical Society*, vol. 89, no. 3, p. 753, 1966.

[90] J. R. Howell, R. Siegel, and M. P. Menguc, *Thermal Radiation Heat Transfer*. Taylor & Francis Inc, Jul. 2015.

[91] M. E. Houssami, A. Lamorlette, D. Morvan, R. M. Hadden, and A. Simeoni, "Framework for submodel improvement in wildfire modeling," *Combustion and Flame*, vol. 190, pp. 12–24, apr 2018.

[92] C. Bruch, B. Peters, and T. Nussbaumer, "Modelling wood combustion under fixed bed conditions," *Fuel*, vol. 82, no. 6, pp. 729–738, apr 2003.

[93] W. Mell, A. Maranghides, R. McDermott, and S. L. Manzello, "Numerical simulation and experiments of burning douglas fir trees," *Combustion and Flame*, vol. 156, no. 10, pp. 2023–2041, oct 2009.

[94] H. E. Anderson, "Aids to determining fuel models for estimating fire behavior," Tech. Rep., 1982.

[95] J. Anderson, *Fundamentals of Aerodynamics (Mcgraw-hill Series in Aeronautical And Aerospace Engineering)*. McGraw-Hill Science/Engineering/Math, 2005.

[96] R. Martin, "Fire rate of spread calculation for two fuels," *Western Journal of Applied Forestry*, vol. 3, no. 2, pp. 54–55, 1988.

[97] D. X. Viegas, J. R. Raposo, D. A. Davim, and C. G. Rossa, "Study of the jump fire produced by the interaction of two oblique fire fronts. part 1. analytical model and validation with no-slope laboratory experiments," *International Journal of Wildland Fire*, vol. 21, no. 7, p. 843, 2012.

[98] N. Cheney, J. Gould, and W. Catchpole, "The influence of fuel, weather and fire shape variables on fire-spread in grasslands," *International Journal of Wildland Fire*, vol. 3, no. 1, p. 31, 1993.

[99] N. Cheney and J. Gould, "Fire growth in grassland fuels," *International Journal of Wildland Fire*, vol. 5, no. 4, p. 237, 1995.

[100] M. S. Innocente, S. M. B. Afonso, J. Sienz, and H. M. Davies, "Particle swarm algorithm with adaptive constraint handling and integrated surrogate model for the management of petroleum fields," *Applied Soft Computing*, vol. 34, pp. 463–484, sep 2015.

[101] B. J. Leonard and A. P. Engelbrecht, "Scalability study of particle swarm optimizers in dynamic environments," in *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 2012, pp. 121–132.

[102] M. S. Innocente and P. Grasso, "Swarms of autonomous drones self-organised to fight the spread of wildfires," in *Proceedings of the GEOSAFE Workshop on Robust Solutions for Fire Fighting*, 2018.

[103] T. M. Blackwell and P. J. Bentley, "Dynamic search with charged swarms," in *Proceedings of the 4th Annual Conference on Genetic and Evolutionary Computation*, ser. GECCO'02. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2002, p. 19–26.

[104] C. W. Reynolds, "Flocks, herds and schools: A distributed behavioral model," in *Proceedings of the 14th annual conference on Computer graphics and interactive techniques - SIGGRAPH '87*. ACM Press, 1987.

[105] R. G. Braga, R. C. da Silva, A. C. B. Ramos, and F. Mora-Camino, "Collision avoidance based on reynolds rules: A case study using quadrotors," in *Information Technology - New Generations. Advances in Intelligent Systems and Computing*, S. Latifi, Ed., vol. 558. Cham: Springer International Publishing, 2018, pp. 773–780.

[106] M. S. Innocente, "Particle swarms reformulated towards a unified and flexible framework," in *Advances in Swarm Intelligence. ICSI 2021. Lecture Notes in Computer Science*, Y. Tan and Y. Shi, Eds., vol. 12689. Springer Nature, 2021, pp. 275–286.

[107] M. Compagnoni, A. Pini, A. Canclini, P. Bestagini, F. Antonacci, S. Tubaro, and A. Sarti, "A geometrical–statistical approach to outlier removal for TDOA measurements," *IEEE Transactions on Signal Processing*, vol. 65, no. 15, pp. 3960–3975, aug 2017.

[108] S. M. Kay, *Fundamentals of Statistical Processing, Volume I.* Prentice Hall, Mar. 1993. [Online]. Available: https://www.ebook.de/de/product/3640711/steven_m_kay_fundamentals_of_statistical_processing_volume_i.html

[109] R. Kaune, J. Horst, and W. Koch, "Accuracy analysis for tdoa localization in sensor networks," in *14th International Conference on Information Fusion*, 2011. [Online]. Available: http://fusion.isif.org/proceedings/Fusion_2011/data/papers/217.pdf?

[110] Bitcraze, "Loco positioning system: Tdoa principles," 2020. [Online]. Available: http://www.bitcraze.io/documentation/repository/lps-node-firmware/2020.09/functional-areas/tdoa_principles/

[111] ——, "Loco positioning system: Tdoa2 vs tdoa3," 2020. [Online]. Available: http://www.bitcraze.io/documentation/repository/lps-node-firmware/2020.09/functional-areas/tdoa2-vs-tdoa3/

[112] D. L. 2015, "Dw1000 ieee802.15.4-2011 uwb transceiver - datasheet v2.09," 2015. [Online]. Available: http://www.decawave.com/sites/default/files/resources/dw1000-datasheet-v2.09.pdf

[113] ——, "Dwm1000 ieee 802.15.4-2011 uwb transceiver module - datasheet v1.3," 2015. [Online]. Available: http://www.decawave.com/sites/default/files/resources/dwm1000-datasheet-v1.3.pdf

[114] M. W. Mueller, M. Hamer, and R. D'Andrea, "Fusing ultra-wideband range measurements with accelerometers and rate gyroscopes for quadrocopter state

estimation," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, may 2015.

[115] M. W. Mueller, M. Hehn, and R. D'Andrea, "Covariance correction step for kalman filtering with an attitude," *Journal of Guidance, Control, and Dynamics*, vol. 40, no. 9, pp. 2301–2306, sep 2017.

[116] S. M. Rimmer, S. J. Hawkins, A. C. Scott, and W. L. Cressler, "The rise of fire: Fossil charcoal in late devonian marine shales as an indicator of expanding terrestrial ecosystems, fire, and atmospheric change," *American Journal of Science*, vol. 315, no. 8, pp. 713–733, oct 2015.