INSA Toulouse
Dept. of Mathematical Engineering and Modeling

FINAL INTERNSHIP REPORT

Regional-scale Wildfire Spread Simulations with a Complex Topography : Data Assimilation for Model State Estimation

Charlotte EMERY

February 4th, 2013 - August 2nd, 2013

At CERFACS and University Of Maryland

Supervisors:

Mélanie Rochoux
Ecole Centrale Paris
Grande voie des vignes
92295 Châtenay-Malabry
France

Sophie Ricci
GLOBC Research Team
CERFACS - URA 1875
42, avenue Gaspard Coriolis
31057 Toulouse cedex 01

Arnaud Trouve
Fire Protection Engineering
University of Maryland
College Park, MD 20742
United State of America

Toulouse, September 10th, 2013

Abstract

Computer-based wildfire spread modeling has emerged during the past two decades as a powerful tool for applications in both fire risk management and fire emergency response. However, because the underlying dynamics feature complex multi-physics occurring at multiple scales, our ability to accurately simulate the behavior of wildfires remains limited. The dynamics of wildfires are determined by interactions between pyrolysis, combustion, flow dynamics as well as atmospheric dynamics. These interactions occur at: vegetation scales that characterize the biomass fuel; topographical scales that characterize the terrain and vegetation boundary layer; and meteorological micro/meso-scales that characterize atmospheric conditions. The mathematical models proposed to simulate wildfire spread are limited because of their inability to cover the entire range of relevant scales, because also of knowledge gaps and/or inaccuracies in the description of the physics as well as knowledge gaps and/or inaccuracies in the description of the controlling input parameters (i.e., the vegetation, topographical and meteorological properties).

Even though relevant insight into wildfire dynamics have been recently made via numeical simulations at flame scales; those studies remain restricted to research projects and are not compatible with real-time forecast operational objectives useful in fire emergency response. The mathematical models proposed to simulate wildfire spread are limited because of their inability to cover the entire range of relevant scales, because also of knowledge gaps and/or inaccuracies in the description of the physics as well as knowledge gaps and/or inaccuracies in the description of the controlling input parameters (i.e., the vegetation, topographical and meteorological properties).Thus, cost-effective approaches imply numerous modeling simplifications and approximations that make the prediction of wildfire spread complicated and uncertain.

A data assimilation approach appears as a new paradigm to overcome some of the current limitations of regional-scale wildfire modeling. In the area of fire research, the idea of data assimilation is particularly attractive given the large uncertainties associated with many of the input variables, in particular in the representation of fuel sources. While still original in the field of fire and combustion, data assimilation is an established approach in several scientific areas, for instance in the field of numerical weather predictions.

This approach has been developed since 2010 at CERFACS and University of Maryland during three previous internships and is also part of a PhD thesis ending by the end of 2013. The previous studies were focused on developing a preliminary wildfire spread simulator limited

to flat terrains and an ensemble-based data assimilation algorithm whose objective was to correct modeling errors by reducing uncertainties in the input parameters of the wildfire spread-rate model

This internship was divided into two steps:

- During the first period of the intership, based at CERFACS in Toulouse, the current data assimilation framework has been modified to directly estimate the location of the fire front instead of the model input parameters. Several synthetic-data experiments have then been carried out to validate the new framework and test its performances and limitations
- During the second period, spent at the University of Maryland in the USA, the wildfire spread model has been updated to take into account the contribution of the topography in the determination of the fire front spread-rate and to be able to spread the fire over any topographical distribution.

Acknowledgments

To begin with, I would like to warmly thank all my supervisors Mélanie Rochoux, Sophie Ricci and Arnaud Trouve for considering my application in the first place and giving me the opportunity to work on this project. They were of a great help and support and always available during my internship. It was a huge pleasure to work with them. I would like to particularly thank Mélanie Rochoux for all the advice she gave me while reviewing my slides and papers for my weekly updates and skype meetings. Also, I am indebted to Sophie Ricci who gave me the opportunity to pursue my journey in a PhD thesis.

I wish to give a general thank to the entire GlobC Team at CERFACS for being so nice with me and allow me to feel comfortable during the first months of my internship.

I am grateful to all my fellow countrymen from University Of Maryland: Arthur, Sebastien, Vivien and Bertrand to allow me not to forget my mother tongue during work days and all the people I met at the Fire Protection Engineering Department; Christine, Zhoreh, Steffen, Andrew, Pat ... for the very nice time spent together and their welcome when I first arrived in the US.

My last thoughts go to my american housmates Ally, Joe, Danielle and Colin whose kindness and friendlyness made me feel like at home for three months; and I will never forget their support when I was experiencing my first tornado alert.

Internship framework

This internship is the continuity of the project entitled "Forest Fire Modeling using Data Assimilation", carried out since 2010 within the collaboration between CERFACS and UMD. Both institutions bring their own experience in scientific computing, CERFACS in CFD and data assimilation, UMD in fire physics and fire modeling. Three previous interns and a PhD student, Mèlanie Rochoux, Blaise Delmotte and Jean-Michel Bart, have worked on the project, developing tools used as basis in this internship and a PhD thesis on the same subject is being completed in 2013.

CERFACS, Toulouse, France

Since the end of the 1980s, CERFACS is a well-known international institute for high performance computing, which aims at developing advanced methods for the algorithmic resolution and numerical simulation of large scale scientific and technological problems. These problems are of interest to both research and industry, and require the most powerful supercomputers currently available.

CERFACS hosts interdisciplinary teams involved in several different research fields as parallel algorithms, code coupling, data assimilation, aerodynamics, combustion, climate and environment, and also electromagnetism and acoustics. The employees are physicists, applied mathematicians, numerical analysts and software engineers and come from 10 different countries. The multidisciplinarity of this wildfire spread modeling project is then eased by the presence of such a wide range of knowledge and expertise.

Although, two research teams intervine and mainly participate in this project:

- The **GlobC** team (Global Change and Climate modeling) focuses on understanding the climate variability on regional to global scales. It aims to improving climate forecasts as well as impact studies at seasonnal-to-decadal time scales. Notably, part of this team is highly qualified in data assimilation. Four main areas of applications are presently explored (atmospheric chemistry, oceanography, hydrolics), and one objective at CERFACS is to widen the scope of applications, for instance to fire propagation.
- The **CFD** team (Computational Fluid Dynamics) and more precisely the Combustion group is involved in all problems related to combustion with high performance computing,

so as to apply LES (Large Eddy Simulation) techniques to model turbulent flows and turbulent combustion. The AVBP (A Virtual Burning Project) solver has been jointly developed by CERFACS and IFP, the French Institute of Petroleum, to perform parallel compressible LES computations, and is specifically designed to simulate industrial gas turbines, aero gas turbines, rocket engines, and laboratory burners.

Fire Protection Engineering Department, University Of Maryland, USA

Located at the border with Washington D.C. in College Park, in the state of Maryland, the University of Maryland was founded in 1856 and was dedicated originally to agricultural studies. With 112 graduate degrees spread over 13 different colleges and schools, among whom the A.J. Clark School of Engineering, it enrols about 37,000 students (including 10,000 postgraduates) from more than 130 countries and is nowadays recognized as one of the top public North-American universities.

Since 1956, the Fire Protection Engineering Department belongs to the School of Engineering and offers the only full ABET-accredited (Accreditation Board for Engineering and Technology) undergraduate program and one of the two graduate degree programs in the US. The department researches focus on recent advances in material testing practices, fire detection, performance-based design and modeling techniques to predict fire growth, smoke movement or the response of building systems in design and fire investigation applications.

Through the BRE (Build Reseach Establishment) Centre for Fire Safety Engineering, its current mission is to develop a real-time emergency response system for enclosure fires. This is a challenging task as the most advanced technologies shall be used so as to limit the computational cost to real-time.

Outline

The content of this report is organized as follow: first, we present the wildfire spread simulator designed to simulate the propagation of surface wind-aided fire on flat terrains. In a second chapter, we present concepts and methods of data assimilation and a brief state-of-the-art of data assimilation applied to wildfire spread modeling. Then, the third chapter is dedicated to the implementation of the state estimation approach for the data assimilation algorithm. With this approach, data assimilation is used to correct the position and the shape of the fire front. Finally, the last chapter details the incorporation of the topography contribution in the wildfire spread model.

The data assimilation part has been carried out in first place for several reasons. First, the three first months of the internship were located at CERFACS in Toulouse where my supervisors were specialized in data assimilation methods. Besides, results from previous internships are limited to flat terrains simulations. Then, testing and validating the state estimation approach on flat terrains allowed us to compare both results. Moreover, it was decided that the incorporation of the topography was to be done when the data assimilation part was complete. Finally, the modification of the model could more easily be done at University of Maryland where the labaratory is highly-skilled in fire modeling.

Contents

Chapter 1

The Wildfire Spread simulator FIREFLY

The dynamics of wildfires are determined by multi-scales interactions between pyrolysis, combustion, flow dynamics as well as atmospheric dynamics. These interactions occur at: vegetation scales that characterize the biomass fuel; topographical scales that characterize the terrain and vegetation boundary layer; and meteorological micro/meso-scales that characterize atmospheric conditions. The understanding of the contribution of those environmental conditions in the wildfire spread allow us to determine the local velocity of the fire, commonly named the rate of spread (ROS). But the interactions of all those multi-physical processes over such a wide range of temporal and spatial scales turn the modeling of wildfire behavior into a challenging problem [1].

Real-time predictions of the direction and speed of a propagating wildfire has been identified as a valuable research objective with direct applications in fire emergency management. A wide range of models has been developed either for academic research projects or for operational applications. These models represent a continuous spectrum of possible modeling approaches, from the purely physical model (based on the fundamental understanding of the fire physics) to purely empirical model (based on phenomenological description of the fire and on parameterization of the rate of spread).

This part aims at first providing a background of the processes involved in wildland fires to understand the main components of wildfire spread mechanism. Then, a short review of wildfire models developed for regional-scale applications will be presented. Finally, the last section will be dedicated to the wildfire spread solver FIREFLY used in the context of this internship: this Eulerian front-tracking solver has been developed at CERFACS since 2010 (see [2, 3, 4]) in order to simulate the propagation of surface wildland fires at regional scales and at a low computational cost.

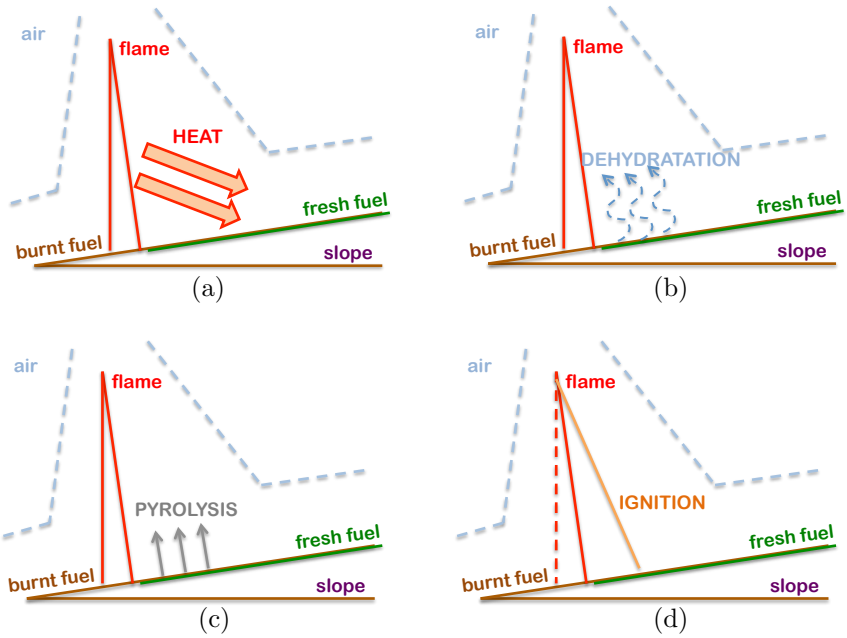


Figure 1.1: Schematic of wildfire spread mechanism [3].

1.1 Fundamentals of fire and combustion

The combustion-related processes involved in wildfires are complex, due to the heterogeneous nature of the vegetation (also referred to as vegetal fuel) and to the numerous physical processes implied (chemical reactions in the gas and solid phases of the vegetation, heat transfers, buoyancy-driven floaw, etc). Wildland vegetal fuel is composed of live and dead plant material consisting primarly of leaf litter, twigs, bark, wood, grasses and shrubs, with a considerable range of physical structures, chemical components, age and level of biological decomposition [5]. Nevertheless, the primary chemical component of biomass fuel is cellulose (of chemical form $(C_6O_5H_{10})_n$). Figure (1.1) illustrates the main steps of the thermal degradation of cellulose leading to the ignition of the vegetal fuel and therefore the spread of a wildfire.

When heat from a close flame or from sun radiation is applied to cellulose (see Fig. (1.1a)), fresh vegetal fuel undergoes a series of chemical reactions called pyrolosis. First, the fuel dehydrates (see Fig. (1.1b)) with the increase of the temperature. The higher the fuel moisture content, the longer the dehydrating and the slower the wildfire spread. Then, when the heating continues, oxydable gases are released (see Fig. (1.1c)) and driven near to the flame by convection effects. Finally, the spread or the start of the wildfire (also called ignition, see Fig. (1.1d)) is induced by the rapid oxidation of thoses flammable gases.

From this quick description, we see that the predominant process involved in wildfire spread is heat transfer. The physical processes allowing heat transfer are numerous: convection, radiation, conduction [6]. Under low wind conditions, radiation from the close flame is the dominating

process [7] while advection becomes predominant with significant positive wind conditions [8]. By "positive", it is implied that the wind blows in the direction of the propagation. The heat is then carried by advection further towards the unburnt vegetal fuel and the thermal degradation of far fuel is initialized earlier resulting in higher values of ROS. Added to these phenomena, the heat released from the chemical reactions results in heated gases in the form of combustion products or heated ambient air. Those heated gases have a reduced density compared to ambient air and so rise by convection. The convection leads to the formation of smoke plume (made of hot gases) which changes the local state of the atmosphere and locally pertubs the wind which, in return, affects the wildfire spread. Therefore, while the fire impacts the local atmosphere conditions by transferring heat and energy through convection and turbulence, changes in the ambient meteorological conditions, such as wind speed, direction, air moisture, have also significant impact on the state of the fuel and so on the behavior of the fire. Finally, the topography of the terrain in which the fire is burning also plays an important part in the fire spread. When a fire spreads in the upslope direction, the transfer of radiant heat to the fuel is increased because the fuel is closer to the flame, resulting in a faster spread.

Relevant insight into wildfire dynamics has been obtained in recent years via detailed numerical simulations performed at flame scales (i.e., with a spatial resolution on the order of 1 m). For instance, FIRETEC [9] or WFDS [10] combine advanced physical modeling and classical methods of Computational Fluid Dynamics (CFD) to accurately describe the combustion-related processes that control the fire spread. Note that because of its high computational cost, flame-scale CFD is currently restricted to research projects and is not compatible with operational applications. The running of CFD simulations for such large fires is too expensive numerically and especially the execution time is too long for real-time simulations in order to make predictions. In contrast, we adopt in the following a regional-scale viewpoint. At regional-scale, wildfires size varies from the square meters scale to several square kilometers.

1.2 An overview of fire spread model at regional scale

1.2.1 Variables of interest at regional scale

For simulations of systems involving flames, several strategies for computing the flame are available. At regional scale, the most common approach is to model the flame as a discontinuity. This approach relies on the assumption that a wildfire exhibits, at a macroscopic scale, a topology that is similar to premixed flames [11]. A premixed flame is a flame in which the oxidizer has been mixed with the fuel before it reaches the flame front. This creates a thin flame front with burnt gases on one side, fresh gas reactants on the other side, and the front propagates towards the fresh gases. Therefore, as illustrated in Fig. (1.2), the fire is considered as a propagating interface between burnt area and non-burnt area. The main variable to determine here is the local Rate Of Spread (ROS) of the fire front, called Γ. This variable is the velocity (defined with respect to a fixed observer) at which the fire spreads in the normal direction to the fire front. Besides, as briefly described in the previous section, the rate of spread depends on local conditions of vegetation, weather and topography.

Figure 1.2: Snapshots of wildland fires.

1.2.2 Classes of fire spread model

Three types of fire spread models can be distinguished:

- **Physical model:** A physical model is one that is based upon fundamental principles of fluid mechanics (mass, momentum, heat conservation equations) and account for the physics (advection, radiation, convection) and chemistry (dehydratation, pyrolysis, ignition) of wildland behavior. Many are based on the same principles and differ only in the methodology of implementation or the purpose of use. Many are implemented only in one or two dimensions in order to improve computational or analytical feasibility. Main models in this category are FIRETEC [9] and WFDS [10]. A detailed overview of physical models since 1990 is presented in [12].

- **Semi-empirical model:** A semi-empirical model is one that is based on a limited physical framework (heat conservation equation only) whose parameters are calibrated using experimental observations and results. Because they rely on physical description (principle of conservation of energy or heat balance), they can be extended from labaratory experiments to field scale experiments (e.g. Australian field experiments of grassfires in a range of wind speed conditions). Semi-empirical models are simple enough to be efficiently implemented for computational applications. One of the most well-known semi-empirical model is due to Rothermel [13].

- **empirical model:** An empirical model is one that is based upon observation and experiment only and not on theory. The focus of empirical modeling of wildland fire is the determination of the key characteristics used to describe the behavior of the fire like the rate of spread of the heading fire (wind spreading with the wind). A detailed overview of empirical and semi-empirical models since 1990 is presented in [14].

Semi-empirical models are the dominant approach used in current operational wildfire spread models, see for instance BehavePlus [15] and FARSITE [16] in the United States, the MacArthur Fire Danger Meters [17] in Australia and the Forest Fire Behavior Prediction System (FBPS) [18] in Canada. BehavePlus and FARSITE use a model due to Rothermel [13] to determine

the ROS. Our wildfire spread simulator FIREFLY also relies on the semi-empirical Rothermel's model.

1.3 A front-tracking solver: FIREFLY

The Eulerian front-tracking solver, called FIREFLY, simulates the propagation of surface wildland fires at regional scales as illustrated in Fig. (1.3). The solver is currently limited to flat terrains and problems with complex topography are investigated in chapter 4 of this report. FIREFLY tracks the evolution of the fire front location using the following three components:

1. The Rothermel's model to determine the ROS with respect to local environmental conditions,
2. A Level-Set-based solver [19] to propagate the surface fire in 2D materialized as a progress variable noted $c = c(x, y, t)$. This flame marker is calculated as a solution of a 2D PDE:
   - $c = 1^-$: the vegetation is burnt,
   - $0^+ \leq c \leq 1^-$: the front zone,
   - $c = 0^+$, the vegetation is fresh.
3. An isocontour algorithm to extract from the 2D propagated field the fire front defined as the isocontour $c = 0.5$.

1.3.1 Sub-model for the rate of spread: the Rothermel's model

The quasi-empirical model of Rothermel is a reference in the fire community as it provides a general expression to determine the ROS of the head of a fire as a function of vegetation (fuel) properties associated with a pre-defined fuel category (i.e., the vertical thickness of the fuel layer, the fuel moisture content, the fuel particle surface-to-volume ratio, the fuel loading and the fuel particle mass density), topographical properties (i.e., the terrain slope) and meteorological properties (i.e., the wind velocity at mid-flame height):

$$\Gamma = \Gamma(x, y, t) = \Gamma\Big(\delta_v, M_v, M_{v,ext}, \Sigma_v, m_v^{''}, \rho_p, \Delta h_c, \mathbf{u}_w\Big) \tag{1.1}$$

where input parameters are summarized in Table 1.1. Any of these parameters can be uniformly or spatially distributed over the computational domain and also time-dependent or not. The values of these parameters directly impact the velocity at which the fire spread.

1D original formulation

The Rothermel's model is based on an energy (as heat) balance model in a unite volume of the fuel layer, first proposed by Fransden [20]. It is based on data obtained from wind tunnel experiments in artificial fuel beds of varying characteristics to correlate fire behavior with measured input variables. For a control volume in the unburnt region, the ROS along the normal direction

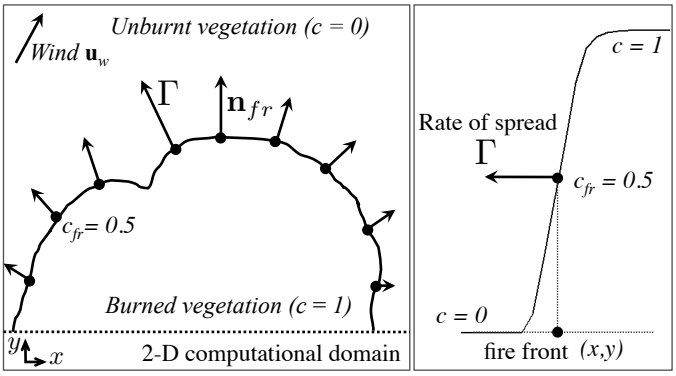


Figure 1.3: FIREFLY front-tracking solver. Left: the fire front is the isocontour $c_{fr} = 0.5$; Γ measures the local rate-of-spread of the fire along the normal direction $\mathbf{n}_{fr}$. Right: profile of the spatial variations of the progress variable c across the fire front.

Name	Symbol	Unit
Fuel depth (vertical thickness of the vegetation layer)	δ_v	m
Fuel moisture (mass of water divided by mass of dry vegetation)	M_v	-
Fuel moisture at extinction	$M_{v,ext}$	-
Fuel particle surface-to-volume ratio	Σ_v	1/m
Fuel loading	m_v''	kg/m^2
Fuel particle mass density	ρ_p	kg/m^3
Fuel heat of combustion	Δh_c	J/kg
Wind velocity vector at mid-flame height (projected into horizontal plane)	$\mathbf{u}_w$	m/s

Table 1.1: Input parameters in the Rothermel-based ROS model.

to the fire front is given by the ratio between the received energy (due to radiation/advection/-conduction) I_p and the necessary energy to ignite the fuel $\rho_b \epsilon Q_{ig}$:

$$\Gamma = \frac{I_p}{\rho_b \epsilon Q_{ig}} \tag{1.2}$$

with:

- I_p, the propagating flux in J.m^{-2}.s^{-1},
- Q_{ig}, the heat of preignition in J.kg^{-1},
- ϵ, the effective heating number (amount of fuel involved in the ignition process) and
- ρ_b the bulk density in kg.m^{-3}. The bulk density is the density of the porous medium, not the density of the solid phase of the medium.

The propagating flux I_p can be expressed as a proportion of the reaction intensity I_R, i.e. the burning gases energy released from the organic matter in the fuels:

$$I_p = \xi I_R \tag{1.3}$$

with ξ the propagating flux ratio. It is the percentage of energy released by the pyrolysis which is involved in the ignition of the fresh fuel. The ROS with no wind and no slope, denoted as Γ_0 therefore reads:

$$\Gamma_0 = \frac{\xi I_R}{\rho_b \epsilon Q_{ig}}. \tag{1.4}$$

Wind and slope contribution to the ROS have been determined in the Rothermel's model in the 1D case of heading and upslope fire. Heading fire implies that the wind is blowing in the same direction as the propagation and upslope fire similarly implies that the fire is propagating in the direct uphill direction. Therefore, wind and slope positively change the propagating heat flux by exposing the potential fuel to additionnal convective and radiant heat. The dimensionless coefficients ϕ_w^* and ϕ_s^* represent the additional propagating flux produced respectively by the wind and the slope. They have been evaluated from experimental data as function of the fuel properties, wind velocity (for ϕ_w^*) and slope stepness (for ϕ_s^*). Inserting these terms in the expression of the ROS (1.4) then gives the general expression of the ROS for heading upslope fire:

$$\Gamma = \Gamma_0(1 + \phi_w^* + \phi_s^*) = \frac{\xi I_R(1 + \phi_w^* + \phi_s^*)}{\rho_b \epsilon Q_{ig}}. \tag{1.5}$$

Model adaptation for surface wildland fire spread

The original configuration of the Rothermel's model is too restrictive for 2D wildland fire spread. The following presents the 2D extrapolation adopted in FIREFLY in order to determine a wind-dependent ROS in 2D direction. For that purpose, we need to define some characteristic angles in the horizontal plane to represent the direction of the wind and the propagation of the fire front: the angles are defined from the "North" direction (see Fig. (1.4)), namely the positive y-coordinates and increasing in the clockwise direction. We thus define:

- The "flame" angle $\theta(x, y, t)$ indicates the outgoing normal direction to the front. As the fire front propagates in time and space, this angle is varying in time and space as well. This angle is associated with the normal vector to the front $\mathbf{n_{flame}}$ defined as:

$$\mathbf{n_{flame}}(x, y, t) = \begin{pmatrix} n_{x,flame}(x, y, t) \\ n_{y,flame}(x, y, t) \end{pmatrix} = \begin{pmatrix} \sin(\theta(x, y, t)) \\ \cos(\theta(x, y, t)) \end{pmatrix} \tag{1.6}$$

- The wind angle θ_w indicates the direction in which the wind is blowing. For a given wind velocity u_w^*, the wind vector is defined as:

$$\mathbf{u_w^*} = \begin{pmatrix} u_w^* \sin(\theta_w) \\ u_w^* \cos(\theta_w) \end{pmatrix} \tag{1.7}$$

As illustrated in Figure (1.4), the ROS highly depends on the wind condition but also on the direction of spread: when the wind blows in the same direction as the propagation direction,

Figure 1.4: Wind-dependent topology of the front

the wind constribution is optimal but at the opposite, when the wind blows against or across the propagation, its contribution is minimal. A possible way of modeling this behavior is to consider, in any spread direction θ, the projected wind in that direction. In other terms, the wind velocity u_w in a direction θ is the general wind velocity u^*_w projected along this direction. It is obtained by calculating the scalar product between the wind vector (1.7) and the ROS vector (1.6). Nevertheless, to make sure that the velocity remains positive, the wind contribution is also forced to 0 when the scalar product becomes negative (back of the fire):

$$u_w(x,y,t) = \begin{cases} \mathbf{n_{flame}}(x,y,t).\mathbf{u^*_w} & , \text{ If } \mathbf{n_{flame}}(x,y,t).\mathbf{u^*_w} > 0 \\ 0 & , \text{ If } \mathbf{n_{flame}}(x,y,t).\mathbf{u^*_w} \leq 0 \end{cases} \tag{1.8}$$

1.3.2 Level-set-based fire spread simulator

A great number of techniques dealing with infinitely thin interfaces is available in the literature. Among them are front tracking methods, known for providing better accuracy such as Level-Set methods. The method works as follows: starting with the current position of the interface and its speed of propagation, the Level-Set solver computes the position at a later time (see Fig. (1.3)).

The Level-Set is a front tracking method commonly used to model the evolution of a moving interface. The central idea of this method has been developed by [21]. Nowadays widely used in combustion [22] (especially for flame capturing), computer vision [23], or more generally front tracking problems, the Levet Set method has been experimented on wildland fire propagation by [19, 24] with promising results. By considering only advection, the interface $c = 0.5$ is guaranteed, thanks to an accurate numerical integration, to propagate at the prescribed speed Γ (see Fig. (1.5)). All the physics of the fire is contained in the ROS model, Γ.

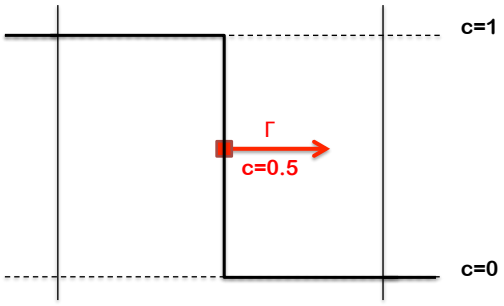


Figure 1.5: 1D Level-Set interface

Formulation

The Level-Set equation describes the front propagation:

$$\begin{cases} \frac{\partial c}{\partial t} + \mathbf{\Gamma}.\nabla c = 0 & , \quad \forall (x,y) \in \Omega, \ t \in [0;T] \\ c(x,y,0) = c_0(x,y) & , \quad \forall (x,y) \in \Omega \\ \nabla c(x,y,t).\mathbf{n}_{\partial\Omega}(x,y) = 0 & , \quad \forall (x,y) \in \partial\Omega \end{cases} \tag{1.9}$$

with

$$\mathbf{\Gamma} = \begin{pmatrix} \Gamma(x,y,t) n_{x,\text{flame}}(x,y,t) \\ \Gamma(x,y,t) n_{y,\text{flame}}(x,y,t) \end{pmatrix}$$

the local ROS in the normal direction to the front.

Implementation

Propagating an interface avoiding numerical diffusion or ensuring numerical stability requires a specific treatment using high order numerical schemes. This section presents the method for solving the Level-Set equation (1.9) as in [19]. The overall algorithm is described as follow:

- The gradient of the progress variable c is calculated using a central finite difference scheme,
- Using these differences, the fire front normal vector $\mathbf{n_{flame}}$ and then $\mathbf{\Gamma}$ are computed,
- The monotonicity preserving the gradient ∇c is calculated with a limiter scheme (Superbee slope limiter [25]),
- Finally, the model is integrated in time using a second order Runge Kutta scheme.

For further details, the reader should refer to [3].

The summary of all the elements of the FIREFLY solver is presented in Fig. (1.6)

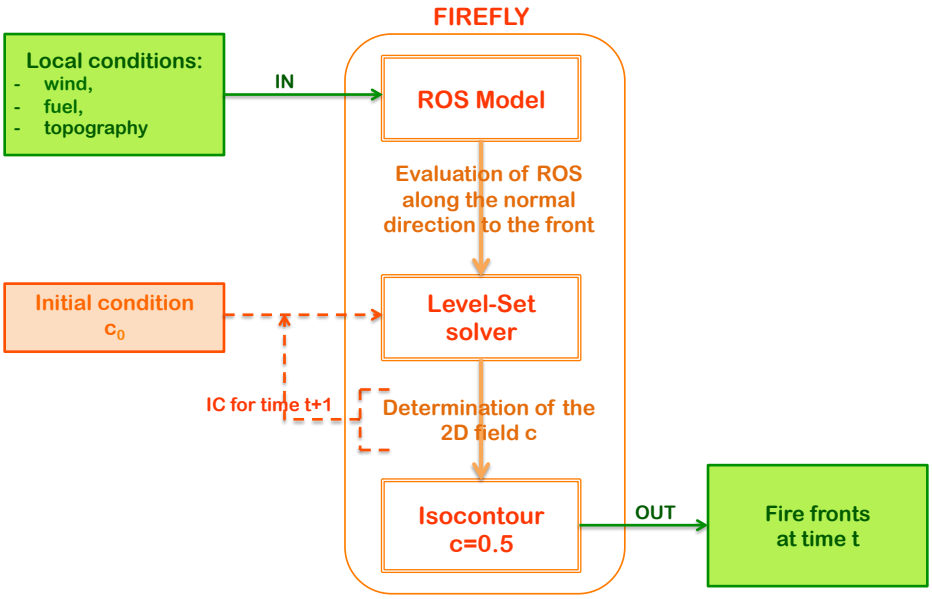


Figure 1.6: Summary of the elements in FIREFLY solver.

1.4 Conclusion

Mathematical models proposed to simulate wildfire spread at a cost consistent with operational framework remain limited because of their inability to cover the entire range of relevant scales. A first limitation in those models is that the ROS model does not explicitly account for the interaction of the fire and the atmosphere and has a domain of validity that is limited to the conditions of the calibration experiments used during its original development. A second limitation is that the input parameters that determine the ROS are often unknown or are only known with limited accuracy.

To be able to develop an efficient wildfire modeling tool capable of forecasting the fire spread dynamics, the wildfire spread model has to be combined to a mathematical algorithm whose objective is to minimize the uncertainties either in the input parameters or in the state of the model and thus, provide more accurate predictions of wildfire behavior. Data assimilation methodologies offer such possibilities by using a set of observations to correct the parameters and/or the state and therefore, allow to overcome some current limitations of the wildfire spread models.

Chapter 2

Data assimilation: concepts, methods and applications

Computational wildfire spread models can be used for two purposes: in re-analysis mode, to analyse the chemical and physical processes underlying physics in past events, and in forecast mode; to predict future behavior of the fire at different lead-times. Within the framework of this internship, both aspects are investigated. In both cases, the model is subject to uncertainties that translate into errors in the model outputs. The use of real-time measurements to constraint the model is a way to reduce these uncertainties. The reduction of the uncertainties is achieved through a data assimilation (DA) process.

The benefits of DA has already been greatly demonstrated in meteorology [26, 27, 28], oceanography [29, 30]; more recently in hydrodynamics, oil and gaz pipeline models [31], wind-turbine prediction and in our case, in regional-scale wildland fire spread modeling [32, 33, 34, 35]. A dynamical model describes the evolution in time of a physical phenomenon. Computer models are used to predict a quantity of interest (temperature, pressure) in the future. However, multi-scale and multi-physics phenomena (eg. fire flame propagation) are quite difficult to model. Also, the model is written using simplifications and numerical discretizations that generate other sources of uncertainties to add up to those in the numerical model initial condition, boundary condition and parameters. In parallel, the observations are imperfect due to instrumental errors, and sparse in time and/or space and they may be undirectly related to the model variables. DA allows to combine information from the model (called the background) with the observations as they become available thus allowing to constrain the model and to reduce the uncertainty in the simulated and forecast state. This constitutes an assimilation cycle. The variables that the DA algorithms aim at correcting is called the control vector, it can gather model variables also called state variables and/or some model parameters. In the fields of application for data assimilation, the natures of the control variables can be for instance:

- In weather forecasting and oceanography, these variables are generally the model state (temperature or pressure field). The optimized state is then used as the initial condition for the model for the next assimilation cycle,

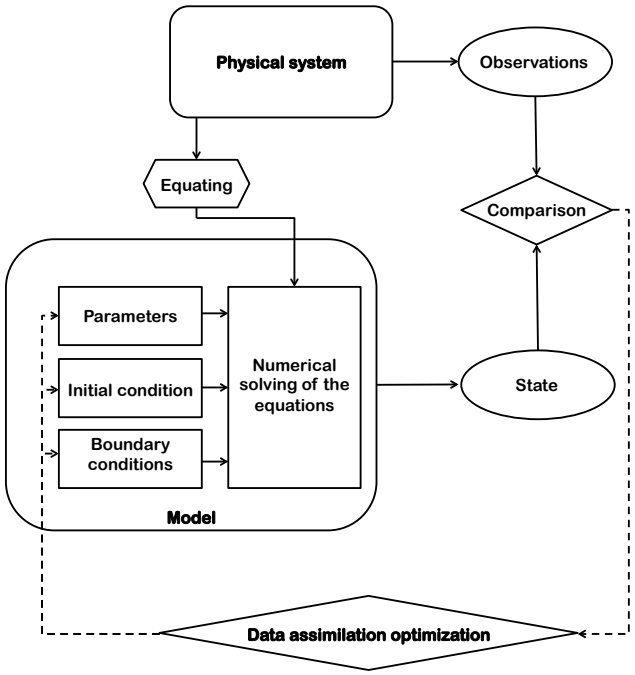


Figure 2.1: General framework of data assimilation

- In hydrodynamics, the control vector gathers the state vector (water level and discharge) as well as boundary conditions (upstream and lateral inflows) and model parameters (bathymetry).

As illustrated in Fig. (2.1), the idea of DA is to use a set of observations (the measurements) obtained by in-situ sensors, spaceborne and airborne images to correct an a priori value (the background) of a set of control variables to produce an optimal description of the control vector (the analysis). The resolution of this inverse problem consists in the minimization of a cost function combining the different sources of information and their respective uncertainties. When the physical model is linear and the cost function quadratic, an optimal solution can be determined (4DVAR,[36, 37], Kalman filter,[38]). These algorithms can be extended for non linear problems: Extended Kalman filter [39, 40], Ensemble Kalman filter, 4DVar-Inc [28, 41].

First, the random variables that are commonly used in DA formalism are presented in section 1[1]. Then, the classical algorithm are presented in section 2. For further details, the reader should refer to [2, 3, 4, 42, 43]. Finally, section 3 aims at presenting a state-of-the-art for DA applied to wildfire modeling and the research works this study is based on.

[1]The denotation with capital letters is for random variables while lower case letters are for realization of those random variables and determinist ones.

2.1 Data assimilation formalism

2.1.1 The data assimilation variables

The model operator and the state vector

The model operator $\mathcal{M}$ is the generally non linear operator representing the numerical algorithm used to solve the set of equations describing the dynamics of the physical system. The state vector gathers the model outputs, e.g the simulated field(s) defined at the grid points of the computational domain.

The control vector

The control vector $\mathbf{X}$ contains all the variables that shoud be corrected by the DA algorithm. The control vector can contain the state vector and/or the model parameters. Therefore, this vector is not systematically in the same space as the state vector. The control vector gathers the variables and parameters to which the model is the most sensitive to and that are most respondible for uncertainty. The control vector defines the control space of size n. It is an approximation of the true control vector $\mathbf{x}^t$, which represents the true values of these n variables which are, in reality, unknown. The background vector is the a priori knowledge of the control vector. It can results from a previous integration of the model (this provides the state) or from the a priori knowledge of the mmodel parameters. DA aims at determining an optimal value of the control vector, called the analysis and denoted by $\mathbf{X}^a$ which is closer to $\mathbf{x}^t$ than the background $\mathbf{X}^b$.

The observation vector and the innovation vector

The observations give an incomplete description in space and time of the dynamical system. In the most general case, the observations, gathered in the vector $\mathbf{Y}^o$, are of different nature from the state variables. The Observing System Simulation Experiment (OSSE) framework is used to validate DA platforms and observations are synthetically generated.

As information from the background and the observations must be compared in the course of DA, an operator $\mathcal{H}$ mapping the control space onto the observation space is defined:

$$\mathcal{H}(\mathbf{X}) = \mathbf{Y} \tag{2.1}$$

with $\mathbf{X}$ the control vector and $\mathbf{Y}$ the equivalent of $\mathbf{X}$ in the observation space. The difference $\mathbf{Y} - \mathbf{Y}^o$ can thus be computed.

In the framework of state estimation (when the control vector is composed of the model variables) and if the observations are of the same nature of the model variables, $\mathcal{H}$ comes down to a linear interpolation operator from the model grid points to the observation grid points. In the framework of parameter estimation, the observation operator $\mathcal{H}$ is the composition of the model operator $\mathcal{M}$:

$$H = \mathcal{S} \circ \mathcal{M} \tag{2.2}$$

mapping the control space to the state space with the selection operator $\mathcal{S}$ mapping from the state space to the osbervation space. In this case, the observation operator is usually non linear as the model is usually non linear.

In order to apply the classical DA algorithm to non linear cases, the observation operator is generally linearized around a reference state or parameters $\mathbf{X}^g$ (usually chosen as the background $\mathbf{X}^b$). The non linear operator is noted[2] $\mathcal{H}$ while it is noted H when it is linear. Its jacobian is denoted by $\mathbf{H}$:

$$\mathcal{H}(\mathbf{X}^g + \delta\mathbf{X}^g) = \mathcal{H}(\mathbf{X}^g) + \left.\frac{\partial\mathcal{H}}{\partial\mathbf{X}}\right|_{\mathbf{X}^g} \delta\mathbf{X}^g + O((\delta\mathbf{X}^g)^2) \tag{2.3}$$

with $\lim_{\delta\mathbf{X}^g\to 0} O((\delta\mathbf{X}^g)^2)(\delta\mathbf{X}^g)^{-2} = 0$ Hence,

$$\mathbf{H} = \left.\frac{\partial\mathcal{H}}{\partial\mathbf{X}}\right|_{\mathbf{X}^g} \tag{2.4}$$

$\mathbf{H}$ is called the linear tangent to $\mathcal{H}$.

The innovation vector displays the difference between the observation vector and the background projection in the observation space:

$$\mathbf{d} = \mathbf{Y}^o - \mathcal{H}(\mathbf{X}^b). \tag{2.5}$$

2.1.2 Definition and modeling of the errors

The background and observations vector only give an approximate description of the true control vector:

- The background error $\boldsymbol{\epsilon^b}$ represents the difference between the true control $\mathbf{x}^t$ and the background control variables:
$$\boldsymbol{\epsilon^b} = \mathbf{X}^b - \mathbf{x}^t, \tag{2.6}$$
- while the observation error $\boldsymbol{\epsilon^o}$ represents the difference between the observations and the true state $\mathcal{H}(\mathbf{x}^t)$:
$$\boldsymbol{\epsilon^o} = \mathbf{Y}^o - \mathcal{H}(\mathbf{x}^t). \tag{2.7}$$
$\boldsymbol{\epsilon^o}$ is the sum of the measurement error $\boldsymbol{\epsilon^m}$ and the representativeness error $\boldsymbol{\epsilon^r}$ that is linked to the incapacity of the model to represent the signal as it is observed (spatial, temporal resolution as well as simplification of the physics).

The statistical description of these errors is given by the Probability Density Functions (PDF). It describes the relative likelihood for these errors to occur at a given point in the control or observation space. For some DA algorithms, in particular those used in the context of this study, those PDFs are assumed Gaussian (see section 2) and therefore, can be fully described by two characteristic variables, namely the expectation $\mathbb{E}$ and the variance Var.

[2]The same notation system is applied to the model operator $\mathcal{M}$ and its linear tangent $\mathbf{M}$.

In DA applications, the vectors are composed of n one-dimensional continuous random variables: $\mathbf{X} = \begin{pmatrix} X^{(1)} & \dots & X^{(n)} \end{pmatrix}^T$. The error $\boldsymbol{\epsilon}$ on those variables are also vectors of n one-dimentional continuous random variables $\boldsymbol{\epsilon} = \begin{pmatrix} \epsilon^{(1)} & \dots & \epsilon^{(n)} \end{pmatrix}^T$. Their expectation and variance are then defined as

$$\mathbb{E}[\boldsymbol{\epsilon}] = \begin{pmatrix} \mathbb{E}[\epsilon^{(1)}] \\ \vdots \\ \mathbb{E}[\epsilon^{(n)}] \end{pmatrix}, \quad \text{Var}[\boldsymbol{\epsilon}] = \mathbb{E}[(\boldsymbol{\epsilon} - \mathbb{E}[\boldsymbol{\epsilon}])(\boldsymbol{\epsilon} - \mathbb{E}[\boldsymbol{\epsilon}])^T]. \tag{2.8}$$

In multi-dimensional case, the second-order moment is a matrix characterized as the covariance matrix of size $n \times n$. Indeed:

$$\begin{aligned} \text{Var}[\boldsymbol{\epsilon}]_{ii} &= \mathbb{E}\left[(\epsilon^{(i)} - \mathbb{E}[\epsilon^{(i)}])^2\right] = \text{Var}[\epsilon^{(i)}], \\ \text{Var}[\boldsymbol{\epsilon}]_{ij} &= \mathbb{E}\left[(\epsilon^{(i)} - \mathbb{E}[\epsilon^{(i)}])(\epsilon^{(j)} - \mathbb{E}[\epsilon^{(j)}])\right] = \text{Cov}[\epsilon^{(i)}, \epsilon^{(j)}]. \end{aligned} \tag{2.9}$$

To compare two dependant random variables, the covariance is defined as a measure of how much two random variables change together. The interpretation of the error covariance matrix is not straigthforward as the correlation and variance information are gathered in the covariance. It is more adapted to work with the correlation matrix associated with it. The correlation between two random variables $\epsilon^{(i)}$ and $\epsilon^{(j)}$ is defined as

$$\rho(\epsilon^{(i)}, \epsilon^{(j)}) = \frac{\text{Cov}[\epsilon^{(i)}, \epsilon^{(j)}]}{\sqrt{\text{Var}[\epsilon^{(i)}]}\sqrt{\text{Var}[\epsilon^{(j)}]}}. \tag{2.10}$$

It is a measure, between 1 and -1, of how linked the errors on two variables are. The correlations can be interpreted as how a variable is affected by the variation of the others:

- A correlation equal to 1 means that it exists an increasing linear relashionship between the two variables. If a correlation between two variables is equal to 1, it implies that when one variable increases (respect. decreases), the other also increases (respect. decreases) with the same amplitude.
- When the correlation is still positive but inferior to 1, the variation of one of the variable will less affect the second variable but they will both vary in the same sense.
- A correlation equal to 0 means that the two variables are completely independant.
- In contrast, a correlation equal to -1 means that the linear relashionship between the two variables is decreasing. It implies that when one variable increases (respect. decreases), the other decreases (respect. increases). So when the correlation is negative but higher then -1, they vary in opposite direction but with different amplitude.

Therefore, by applying the definition of the variance/covariance in Eq. (2.9) to the error vectors $\boldsymbol{\epsilon^b}$ and $\boldsymbol{\epsilon^o}$, we can define the background and observation error covariance matrix $\mathbf{B}$ and $\mathbf{R}$. Also, by applying the definition of correlation in Eq. (2.10) to the error covariance matrix $\mathbf{B}$, we can define the associated correlation matrix $\mathbf{C}$.

Since the true control vector $\mathbf{x}^t$ is unknown, the errors and their statistics are also unknown ans should be estimated. One way to estimate these statistics is to use a sample of N_e independant realizations. For one-dimensional random variable $X_e = \begin{bmatrix} x^{[1]} & \dots & x^{[N_e]} \end{bmatrix}$, the sample expectation and variance are calculated as

$$\overline{X_e} = \frac{1}{N_e} \sum_{l=1}^{N_e} x^{[l]}, \quad \mathrm{Var}[X_e] = \frac{1}{N_e - 1} \sum_{l=1}^{N_e} (x^{[l]} - \overline{X_e})^2. \tag{2.11}$$

Identically, for random vector $\mathbf{X}_e = \begin{pmatrix} X_e^{(1)} & \dots & X_e^{(n)} \end{pmatrix}^T$, the sample moments can be defined as

$$\overline{\mathbf{X}_e} = \begin{pmatrix} \overline{X_e^{(1)}} \\ \vdots \\ \overline{X_e^{(n)}} \end{pmatrix}, \quad \mathrm{Var}[\mathbf{X}_e] = \frac{1}{N_e - 1} \sum_{l=1}^{N_e} (\mathbf{X}_e - \overline{\mathbf{X}_e})(\mathbf{X}_e - \overline{\mathbf{X}_e})^T. \tag{2.12}$$

This method will be exploited in this study when applying the EnKF.

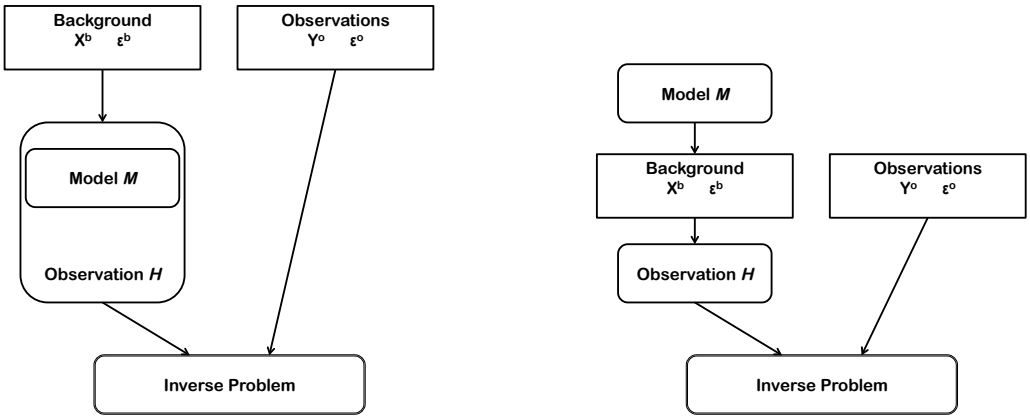
In the framework of DA, the errors are assumed to be unbiased giving that their expectation is equal to zero so the only quantities that remain to describe are the covariance matrices. These matrices are respectively denoted by $\mathbf{B}$, $\mathbf{R}$ and $\mathbf{A}$ for the background error covariance matrix, the observations error covariance matrix and analysis error covariance matrix. Therefore, we will consider in the following of the study that:

$$\boldsymbol{\epsilon^b} \sim \mathcal{N}(0, \mathbf{B}), \quad \boldsymbol{\epsilon^o} \sim \mathcal{N}(0, \mathbf{R}). \tag{2.13}$$

2.1.3 Synthesis on the general data assimilation inverse problem

In the following, the statistics of the errors are described with covariance models and also, the errors on the observations and prior information are assumed to be uncorrelated and to follow a Gaussian distribution. The resolution of the data assimilation problem requires therefore four components:

- observation vector of the physical system $\mathbf{Y}^o$ and their associated errors statistics $(\boldsymbol{\epsilon^o}, \mathbf{R})$,
- a background estimation of the control vector $\mathbf{X}^b$ and its associated errors statistics $(\boldsymbol{\epsilon^b}, \mathbf{B})$,
- a model $\mathcal{M}$ describing the system dynamics,
- an observation operator $\mathcal{H}$ and its associated linear tangent operator with respect to the control vector $\mathbf{H}$ if required.



(a) parameter estimation (b) State estimation

Figure 2.2: Interweaving of the different components of the DA inverse problem in the case of parameter estimation (a) and state estimation (b): the difference comes from the use of the model operator M. For parameter estimation, the M is used within the observation operator and for state estimation, the control is obtained from the integration of the model.

The DA problem can be formulated as an inverse problem: given $\mathbf{Y}^o$ and $\mathbf{X}^b$, the goal is to approximate in an optimal sense the true control vector $\mathbf{x}^t$ satisfying:

$$\begin{aligned} \mathbf{X}^b &= \mathbf{x}^t + \boldsymbol{\epsilon}^{\boldsymbol{b}}, \\ \mathbf{Y}^o &= \mathcal{H}(\mathbf{x}^t) + \boldsymbol{\epsilon}^{\boldsymbol{o}}. \end{aligned} \tag{2.14}$$

Figure (2.2) presents how each of these four components are imbricated for parameter and state estimation. The main difference between the two approaches is when the model operator $\mathcal{M}$ is used. In parameter estimation, the model operator is a part of the observation operator to map from the control space (the parameters) to the state state (state variables). In state estimation, it is used to evolve the control variables in time.

2.2 Data assimilation algorithms

2.2.1 Optimal estimation

To find an estimate of the true state, the idea is to determine the probability distribution of the current control $\mathbf{X}$ conditionned by the measurements $\mathbf{Y}^o$. Under the assumption of linear observation and model operator H and M, the Bayesian theory on probabilities give us that:

$$p(\mathbf{X}|\mathbf{Y}^o)p(\mathbf{Y}^o) = p(\mathbf{Y}^o|\mathbf{X})p_b(\mathbf{X}) \tag{2.15}$$

with p_b the forecast PDF. In the data assimilation procedure, the measurements have been made so the values are known giving $p(\mathbf{Y}^o) = 1$. The equation (2.15) becomes then

$$p_a(\mathbf{X}) = p(\mathbf{X}|\mathbf{Y}^o) = p(\mathbf{Y}^o|\mathbf{X})p_b(\mathbf{X}) \tag{2.16}$$

with p_a the analysis PDF.

Assuming that the PDFs of the background and observations errors are Gaussian, the probability density function of the current control can be written as:

$$p_b(\mathbf{X}) \propto \exp\left(-\frac{1}{2}(\mathbf{X}-\mathbf{X}^b)^T\mathbf{B}^{-1}(\mathbf{X}-\mathbf{X}^b)\right). \tag{2.17}$$

In the same way, the data $\mathbf{Y}^o$ likelihood given the control $\mathbf{X}$ is defined by:

$$p(\mathbf{Y}^o|\mathbf{X}) \propto \exp\left(-\frac{1}{2}(\mathbf{Y}^o-H(\mathbf{X}))^T\mathbf{R}^{-1}(\mathbf{Y}^o-H(\mathbf{X}))\right), \tag{2.18}$$

with $H(\mathbf{X})$, the representing what would the data be if the control were exact. From Eq. (2.16), we find that the posterior PDF is also Gaussian:

$$p_a(\mathbf{X}) \propto \exp\left(-\frac{1}{2}(\mathbf{X}-\mathbf{X}^b)^T\mathbf{B}^{-1}(\mathbf{X}-\mathbf{X}^b) - \frac{1}{2}(\mathbf{Y}^o-H(\mathbf{X}))^T\mathbf{R}^{-1}(\mathbf{Y}^o-H(\mathbf{X}))\right). \tag{2.19}$$

We want to find the control $\mathbf{X}$ that maximizes the probability p_a, so:

$$\max_{\mathbf{X}} p_a(\mathbf{X}) \Longleftrightarrow \min_{\mathbf{X}} -\ln\left(p_a(\mathbf{X})\right). \tag{2.20}$$

Leading to the minimization problem defined as

$$\min_{\mathbf{X}} J(\mathbf{X}) = \min_{\mathbf{X}}\left\{\frac{1}{2}\left(\mathbf{X}-\mathbf{X}^b\right)^T\mathbf{B}^{-1}\left(\mathbf{X}-\mathbf{X}^b\right) + \frac{1}{2}\left(\mathbf{Y}^o-H(\mathbf{X})\right)^T\mathbf{R}^{-1}\left(\mathbf{Y}^o-H(\mathbf{X})\right)\right\}. \tag{2.21}$$

Equivalently:

$$\nabla J(\mathbf{X}^a) = \mathbf{B}^{-1}(\mathbf{X}^a-\mathbf{X}^b) - \mathbf{H}^T\mathbf{R}^{-1}(\mathbf{Y}^o-H(\mathbf{X}^a)) = 0 \tag{2.22}$$

with $\mathbf{X}^a$ the analysis control vector. DA methods that focuses on solving Eq. (2.22) by using a minimizer are commonly named variational methods.

However, another approach is possible by directly defining the analysis from Eq. (2.22) as:

$$\mathbf{X}^a = \mathbf{X}^b + \mathbf{K}(\mathbf{Y}^o - H(\mathbf{X}^b)). \tag{2.23}$$

Those methods are named filtering methods. The Bayesian update Eq. (2.23) is then the basis of many filtering methods such as the Kalman filter (presented in the next section). The correction term also called the analysis increment satisfies: $\delta\mathbf{X}^a = \mathbf{K}(\mathbf{Y}^o - H(\mathbf{X}^b)) = \mathbf{K}\mathbf{d}$, with $\mathbf{K}$: the gain matrix:

$$\mathbf{K} = \left(\mathbf{B}^{-1} + \mathbf{H}^T\mathbf{R}^{-1}\mathbf{H}\right)^{-1}\mathbf{H}^T\mathbf{R}^{-1} = \mathbf{B}\mathbf{H}^T\left(\mathbf{H}\mathbf{B}\mathbf{H}^T + \mathbf{R}\right)^{-1}. \tag{2.24}$$

Therefore, variationnal and filtering approaches differs from how equation (2.22) is solved. If there are only a few variables to correct, e.g. as in parameters calibration, the filtering approach

is more efficient than the variationnal one. Calculating an analytic solution using a linear approximation is indeed less costly than the evaluation of the cost function and its gradient at each iteration. On the other hand, when the number of control variables increases and/or the observation operator $\mathcal{H}$ is highly nonlinear, the cost function direct minimization may be less costly and/or more precise to find a more optimal/realistic solution. However, if the observation operator H is linear, the two approaches are completely equivalent. For more informations on the formalism, please refer to [42, 43].

The application of Eq. (2.23) can lead to different scenarii:

- In the most general case, the analysis is a linear combination of the background and the observations. The contribution of each part is then weighted by the confidence, given by the inverse of the errors, we have in each source of information.

- A perfect confidence in the backgound $\mathbf{X}^b$ leads to a zero matrix $\mathbf{B}$ and therefore a zero matrix $\mathbf{K}$. In this context the assimilation correction is zero and

$$\mathbf{X}^a = \mathbf{X}^b. \tag{2.25}$$

- Conversely, if the confidence is total in the observations, $\mathbf{R}$ tends to the zero matrix. Then, if $\mathbf{H}$ is invertible, $\mathbf{K} = \mathbf{H}^{-1}$. Thus Eq. (2.23) becomes

$$\mathbf{X}^a = \mathbf{H}^{-1}\mathbf{Y}^o. \tag{2.26}$$

2.2.2 The Kalman filter (KF)

The Kalman filter was first proposed by R.E. Kalman in 1960 [44] and is the sequential version of the least-squares analysis seen in previous sections under linear assumptions for the operator H and M. In real-time data assimilation, the Kalman filter and its extensions are widely used because they are adapted to sequential data assimilation where the observations are obtained frequently in time (the green spot in Fig. (2.3)). In the Kalman filter community, the notation differs from those used in the first part of this report. The subscript "b" for the background value of the control variables is replaced by the subscript "f" for forecast as it is obtained by propagating in time the optimal estimate from the previous assimilation step. Moreover, the background (respec. analysis) error covariance matrix $\mathbf{B}$ (respec. $\mathbf{A}$) is renamed $\mathbf{P}^f_{(k)}$ (respec. $\mathbf{P}^a_{(k)}$).

The two steps of the standard Kalman filter equations are defined as a prediction or forecast step and an update or analysis step, as illustrated in Fig. (2.3).

1. The prediction step consists in, at time k, advancing the model in time using the analysis from the previous step time as initial condition. In the same time, the background error covariance matrix is calculated using the analysis error covariance matrix from the previous time step:

$$\mathbf{X}^f_{(k)} = M_{(k-1)\to(k)}(\mathbf{X}^a_{(k-1)}), \qquad \mathbf{P}^f_{(k)} = \mathbf{M}_{(k-1)\to(k)}\mathbf{P}^a_{(k-1)}\mathbf{M}^T_{(k-1)\to(k)}. \tag{2.27}$$

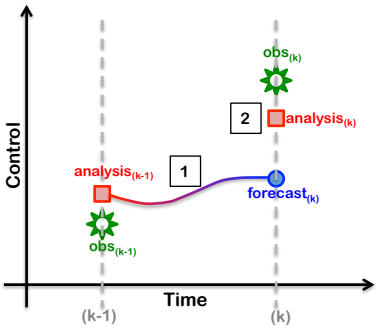


Figure 2.3: The Kalman filter is a two-steps algorihtm: 1. A prediction step during which the model is forecasted in time, 2. An updating step where the analysis is estimated.

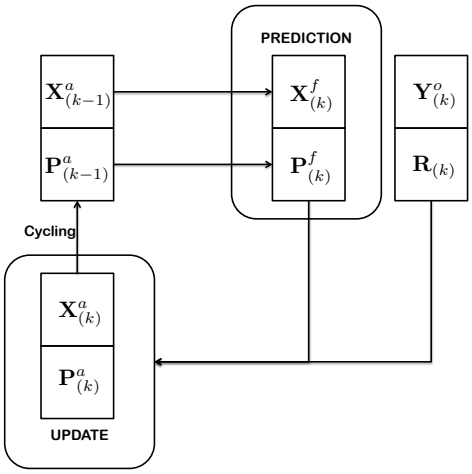


Figure 2.4: Kalman Filter algorithm

Note that in the present study, we consider that the model is perfect. Usually, the definition of $\mathbf{P}^f_{(k)}$ in Eq. (2.27) is completed by the addition of a matrix $\mathbf{Q}$ defining the error covariance matrix of the model itself, uncorrelated with from the error brought from the previous iteration.

2. Then, the update step defines an optimized estimate of the true state $\mathbf{X}^a_{(k)}$ and the associated error covariance matrix $\mathbf{P}^a_{(k)}$ based on equation (2.23) at time k :

$$\mathbf{X}^a_{(k)} = \mathbf{X}^f_{(k)} + \mathbf{K}_{(k)}(\mathbf{Y}^o_{(k)} - \mathbf{H}\mathbf{X}^f_{(k)}). \tag{2.28}$$

Figure (2.4) shows how the KF variables are cycled within the two KF steps. Note that for the KF algorithm, the model and observation operators are linear. Then, the linear tangent operators, denoted by $\mathbf{M}$ and $\mathbf{H}$ and the original operators M and H are equal.

The extensions or equivalent of the Kalman filter are numerous [45]. Among them, some are

based on the updating equations but get close from "smoother methods", referred to as Kalman smoothing, see [46, 47]. Moreover, the first important limitation of the Kalman filter is that the operator $\mathcal{M}$ and $\mathcal{H}$ are supposed linear (giving M and H) but, in most cases in DA, the model is nonlinear and so are the operators. To overcome this problem, the Extended Kalman filter (EKF) was introduced as an extention of the Kalman filter.

2.2.3 The Extended Kalman filter (EKF)

In the case of a nonlinear model $\mathcal{M}$ and of a nonlinear observation operator $\mathcal{H}$, it is still possible to apply the Kalman filter equations by taking the linear operators $\mathbf{M}$ and $\mathbf{H}$ as Jacobian matrix of $\mathcal{M}$ and $\mathcal{H}$ respectively. Both operators are called the linear tangent:

$$\mathbf{M} = \left.\frac{\partial \mathcal{M}}{\partial \mathbf{X}}\right|_{\mathbf{X}^g} \qquad \mathbf{H} = \left.\frac{\partial \mathcal{H}}{\partial \mathbf{X}}\right|_{\mathbf{X}^g}. \tag{2.29}$$

Commonly, the prior estimate $\mathbf{X}^f$ is chosen as the reference $\mathbf{X}^g$.

The equations of the EKF are the same as the Kalman filter's ones except that the nonlinear form of the operators are used in the forecast step and to calculate the innovation. The linear tangent versions are only used to determine the error covariance matrices such that:

- Prediction/Forecast step:

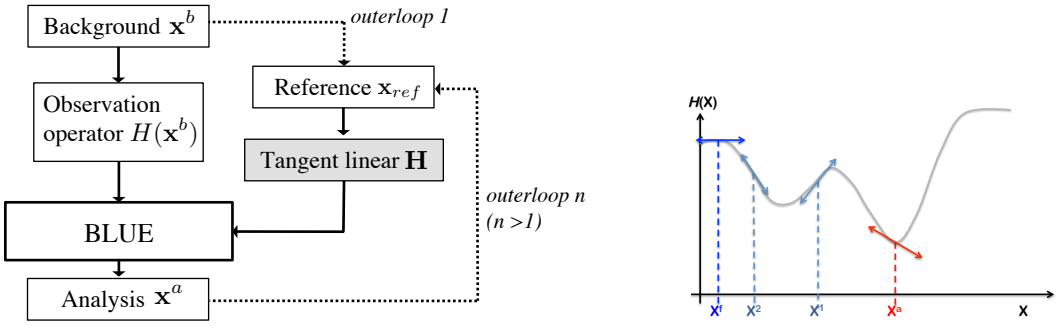
$$\mathbf{X}^f_{(k)} = \mathcal{M}_{(k-1)\to(k)}(\mathbf{X}^a_{(k-1)}), \qquad \mathbf{P}^f_{(k)} = \mathbf{M}_{(k-1)\to(k)}\mathbf{P}^a_{(k-1)}\mathbf{M}^T_{(k-1)\to(k)}, \tag{2.30}$$

- Update/Analysis step:

$$\mathbf{X}^a_{(k)} = \mathbf{X}^f_{(k)} + \mathbf{P}^f_{(k)}\mathbf{H}^T\left(\mathbf{H}\mathbf{P}^f_{(k)}\mathbf{H}^T + \mathbf{R}_{(k)}\right)^{-1}\left(\mathbf{Y}^o_{(k)} - \mathcal{H}(\mathbf{X}^f_{(k)})\right), \qquad \mathbf{P}^a_{(k)} = \left(\mathbf{I} - \mathbf{K}_{(k)}\mathbf{H}\right)\mathbf{P}^f_{(k)}. \tag{2.31}$$

Unlike the standard KF, the EKF is not an optimal estimator due to the use of the linear tangent models. In addition, the EKF may quickly diverge if the prior estimate is far away from the true state and generally, the estimated covariance matrices are underestimated. The linear tangent is equivalent to a linear approximation of $\mathcal{H}$ (or $\mathcal{M}$) in the vicinity of a given reference, here $\mathbf{X}^g$. The choice of $\mathbf{X}^g$ is then crucial to obtain a correct analysis control. An idea to find the best reference control was used in [32]: as illustrated in Fig. (2.5a), an external loop over the reference control $\mathbf{X}^g$ is introduced in the EKF algorithm (called as the BLUE box in the figure). That allows then to correct the control several times at the same assimilation cycle by updating the value of $\mathbf{X}^g$ at each iterations. The reference is initialized at $\mathbf{X}^f$ and updated with the analysis $\mathbf{X}^a$ on the previous iteration in the loop until a convergence is reached.

As illustrated in Fig. (2.5b), an "optimal" value of the linear tangent is progressively estimated using several iterations. Then, the analysis is found thanks to these successive quadratic minimizations. If the cost function to minimize is not quadratic and no strictly decreasing, the



(a) External loop (b) Convergence Issue

Figure 2.5: Data Assimilation algorithm formulated as a BLUE iterative [32]: principle of the converging loop for the EKF process.

external loop may encounter local minima or even diverges. The EKF linearizes, sometimes unrealistically, the observation $\mathcal{H}$ (and also model $\mathcal{M}$) operator based on local derivatives which are often difficult to compute reliably(because of difficult analytical determination or expensive finite differences approximations of the linear tangent). So, to better account for nonlinearities, an ensemble-based DA approach can be used: ensemble forecasts stochastically characterize the nonlinear response of models to variations in the input data. The Ensemble Kalman filter offers then an ensemble-based version of the KF. Besides, apart from the nonlinearity limits of the EKF, difficulties may be encountered to model the error covariance matrix because of storage capacities. Numerous works have been done in order to determine the background error covariance matrix: [48, 49] used wavelets transforms and their properties towards affine transforms to represent covariances and correlation while [50] used a spectral diagonal approach. Many other techniques can be used, with some described in [51]. The Ensemble Kalman filter, even if it remains based on the Gaussian distribution assumption, suggests then an alternative as to use Monte-Carlo-based approach to stochastically characterize the covariance matrix. The stochastic approach allows to better take into account the nonlinearities of the physics while also avoiding the storage of the matrix from one assimilation cycle to another.

2.2.4 The Ensemble Kalman filter (EnKF)

The Ensemble Kalman filter (EnKF) was first proposed by Evensen: in a prior study in [52] and rigorously formulated in [53] as a Monte-Carlo implementation of the Bayesian update used previously in the standard Kalman filter. The EnKF algorithm stochastically approximates the forecast PDF of the control vector $p_f(\mathbf{X})$ by performing a series (an ensemble) of N_e independent simulations, thereby providing N_e forecast estimates of the control vector. The ensemble containing all the prior members is called the forecast ensemble:

$$\left\{\mathbf{x}^{f,[1]} \quad \mathbf{x}^{f,[2]} \quad \dots \quad \mathbf{x}^{f,[N_e]}\right\}. \tag{2.32}$$

The step in which the prior members are generated is the most important because the ensemble needs to represent the control likelihood the best way possible giving sampling errors and

approximations.

To be coherent with the stocastic approach, we need to consider the only observation vector as a random variable. This is allowed by the use of the process of randomization of data [54]: from the original measurement, N_e observation vectors are generated by randomly pertubing it. Using the estimated standard deviation on the observations σ^o, the observed data for the j-th member is calculated by:

$$\forall j = 1 \dots N_e,\ \mathbf{y}^{o,[j]} = \mathbf{y}^o + \epsilon^o_j, \qquad \epsilon^o_j \sim \mathcal{N}(0, \sigma^o). \tag{2.33}$$

The observation ensemble is then defined as:

$$\left\{ \mathbf{y}^{o,[1]} \quad \mathbf{y}^{o,[2]} \quad \dots \quad \mathbf{y}^{o,[N_e]} \right\}. \tag{2.34}$$

The unknown background error covariance matrix $\mathbf{P}^f$ is now replaced by its stochastic approximation over the N_e members $\mathbf{P}^f_e$. If there are n control variables, the size of the matrix is $n \times n$. Is it calculated using the N_e control realizations $\mathbf{x}^{f,[.]}$ centered by their sample expectations, based on Eq. (2.12):

$$\mathbf{P}^f_e = \frac{1}{N_e - 1}\mathbf{N}\mathbf{N}^T \text{ with } \mathbf{N} = \left[\mathbf{x}^{f,[1]} - \overline{\mathbf{x}^f},\ \mathbf{x}^{f,[2]} - \overline{\mathbf{x}^f},\ \dots,\ \mathbf{x}^{f,[N_e]} - \overline{\mathbf{x}^f} \right]. \tag{2.35}$$

The observation error covariance matrix $\mathbf{R}$ could either be modeled by its sample version $\mathbf{R}_e$ or either be modeled using the observation error standard deviation σ^o. According to KF equations, $\mathbf{R}$ has to be invertible. By defining the matrix by its sample version using the randomized data $\mathbf{Y}^o_e$, this matrix will not necessarily be diagonal, meaning that the observations are not independent and besides, the matrix can be singular and more expensive to invert. So, the covariance matrix is based on uncorrelated observation errors: the matrix is diagonal, positive, definite with each diagonal element equal to the observation error variance $(\sigma^o)^2$. For p observations, the matrix is of size $p \times p$:

$$\mathbf{R} = (\sigma^o)^2 \times \mathbf{I} \tag{2.36}$$

with $\mathbf{I}$ the identity matrix.

When all the required ensembles are built, the Ensemble Kalman filter can be applied to produce an ensemble of analysis vectors $\left\{ \mathbf{x}^{a,[1]} \quad \mathbf{x}^{a,[2]} \quad \dots \quad \mathbf{x}^{a,[N_e]} \right\}$ such that the KF update is applied to each member independently:

$$\forall l = 1 \dots N_e,\ \mathbf{x}^{a,[l]}_{(k)} = \mathbf{x}^{f,[l]}_{(k)} + \mathbf{P}^f_{e,(k)}\mathbf{H}^T \left(\mathbf{H}\mathbf{P}^f_{e,(k)}\mathbf{H}^T + \mathbf{R} \right)^{-1} \left(\mathbf{y}^{o,[l]}_{(k)} - \mathcal{H}(\mathbf{x}^{f,[l]}_{(k)}) \right). \tag{2.37}$$

Ensemble-based methods like the EnKF offers more flexibility in the kind of computational models and physical systems to associate with in DA applications. They have already been used in the wildfire modeling community for temperature state correction. Before presenting some existing research works about DA applied to wildfire modeling, it is important to quickly introduce the notion of cycling, necessary for sequential DA.

2.2.5 Cycling of the analysis

The description of the above algorithms is given at one particular observation time. Yet, those algorithms and all DA algorithms in general are applied on several assimilation cycles. Therefore, some particular treatments of the analysis may be required. Those treatments allow to generate the model state corresponding to the current analysis control to be used as initial condition for the next model integration:

- For parameter estimation, the analysis control vector at the time (k) is directly used as background control vector at the time (k+1). But also, the model is re-integrated from the time (k-1) to (k) using the analysis control vector. The model state at time (k) obtained from this integration is then used as initial condition for the next model integration from time (k) to time (k+1).
- For state estimation, the corrected state is directly used as initial condition for the next model integration. The control vector may also be a subspace of the state vector. Thus, some reconstruction algorithm are required to reconstruct the state vector from the control vector.

Now that general DA algorithms have been introduced, the following presents an overview of DA techniques applied in wildfire spread modeling.

2.3 Applying data assimilation to wildfire modeling

In wildfire spread modeling, given the large uncertainties in the input variables, the control vector can be composed of either the Rothermel's model parameters or the fuel distribution, the temperature field or the fire front position or even all of them together. A first option, namely parameter estimation, allows to improve the fire modeling performances by using feedback on the physics of the problem to improve the calibration of the model parameters. The second option, namely state estimation, is based on the fact that the model cannot be perfectly corrected due to the multiple sources of uncertainties in the model parameters but also in the chaotic nature of the system and the model formulation itself. Then, a direct correction on the model state variables can provide insights in the magnitude of the uncertainties and especially on the model error.

2.3.1 Parameter estimation

DA has been widely applied for model parameter estimation. In particular in hydrology [55] by estimating the model input as run-off coefficient or characteristics of conceptual reservoirs; and also in hydrodynamics [56, 57] by estimating river bathymetry or friction coefficient. For wildfire modeling, researches have been led in the framework of a collaboration between CERFACS and UMD [2, 3, 4] and the results were published in [32, 33, 34].

In this work, the parameters that are estimated describe weather conditions eg. wind magnitude and direction $\mathbf{u}_w$, topography information (slope) and fuel layer properties eg. moisture

content M_f, surface-to-volume ratio Σ. Thus, the control vector $\mathbf{X}$ is composed of a set of two to eight parameters used in the calculation of the fire local rate of spread Γ, based on the Rothermel's model [13]:

$$\mathbf{X} = \left[\delta_v(x,y), M_v, M_{v,ext}, \Sigma_v, m_v^{''}, \rho_p, \Delta h_c, \mathbf{u}_w(t)\right], \tag{2.38}$$

$$\Gamma = \Gamma(x,y,t) = P\Big(\delta_v(x,y), M_v, M_{v,ext}, \Sigma_v, m_v^{''}, \rho_p, \Delta h_c, \mathbf{u}_w(t)\Big)\, \delta_v(x,y), \tag{2.39}$$

where input parameters are summarized in Tab. (1.1). In the following, the wind velocity $\mathbf{u}_w$ is treated as a time-dependent, spatially-uniform quantity; the fuel depth δ_v is treated as a time-independent, spatially-varying quantity; all other parameters are treated as constant and uniform. The choice was made to consider uniformly-distributed parameters in order to limit the size of the control vector and allow the use of a filtering algorithm where the correction is explicitely calculated using Eq. (2.28). For a full description of the fire model FIREFLY, the reader should refer to Chapter 1.

The DA algorithm uses a discretization of observed fire fronts as a set of N_{fr}^o markers. The observation vector at time t $\mathbf{Y}_t^o$ is defined as $\mathbf{Y}_t^o = [(x_1^o, y_1^o), \ldots, (x_{N_{fr}}^o, y_{N_{fr}}^o)]$. To map the control space onto the osbervation space, the construction of the observation operator $\mathcal{H}$ is a two-step operation. $\mathcal{H}(\mathbf{X})$ is the composition of the numerical model FIREFLY $\mathcal{M}_{[t-1,t]}(\mathbf{X})$ and of a selection operator $\mathcal{S}(c)$:

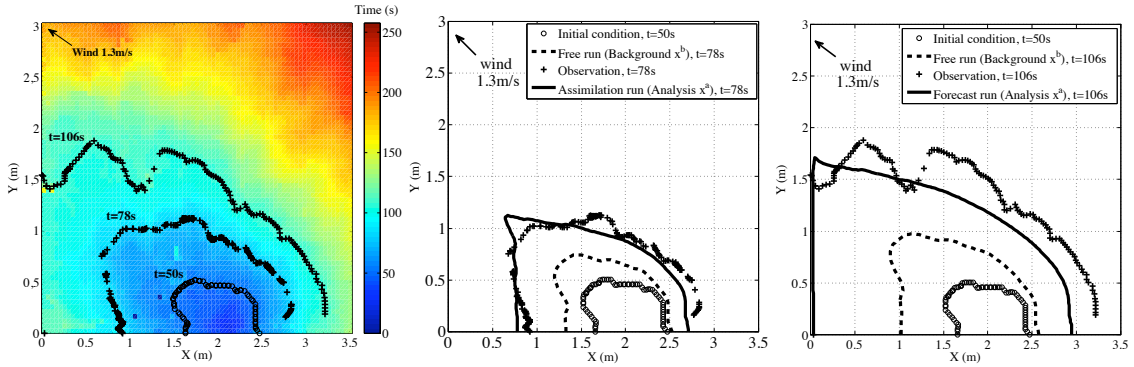
$$\mathcal{H}(\mathbf{X}) = (\mathcal{S} \circ \mathcal{M})(\mathbf{X}) \tag{2.40}$$

with:

- The numerical model $\mathcal{M}_{[t-1,t]}(\mathbf{X})$ that integrates the physics on the assimilation time window $[t-1, t]$ using the control vector parameters $\mathbf{X}$. It provides the simulated field of the progress variable c associated to this particular control vetor.
- The selection operator $\mathcal{S}(c)$ that extract from c the isocontour $c = 0.5$ as a discretization of the simulated front with N_{fr}^o to match the observed front.

This observation operator is nonlinear and has to be estimated using a linear tangent model. In the framework of EKF implementation in [33], the operator $\mathcal{H}$ is linearly estimated using finite differences. In more recents studies based on the EnKF [32, 34], the use of a large number of members allows good estimations but remains limited by the computationnal cost. In both approaches, the results have to be interpreted with parsimony because the nonlinearities are only treated locally.

This approach gives good estimates of the parameters and, at a large-scale, the simulated front is found to properly represent the forecast trajectory. But, somes cases where the fire front is locally deformed (due to unknown local fuel properties different from the uniform description used in the model), showed some limits. As illustrated in Fig. (2.6b-c) (EKF applied to two parameters) and Fig. (2.7) (EnKF applied to four parameters) and from the same set of observed data of a controlled grassland fire experiment in both simulations; applying a spatially uniform



(a) Arrival times of the fire front (b) Assimilation at $t = 78$ s (c) Assimilation at $t = 106$ s

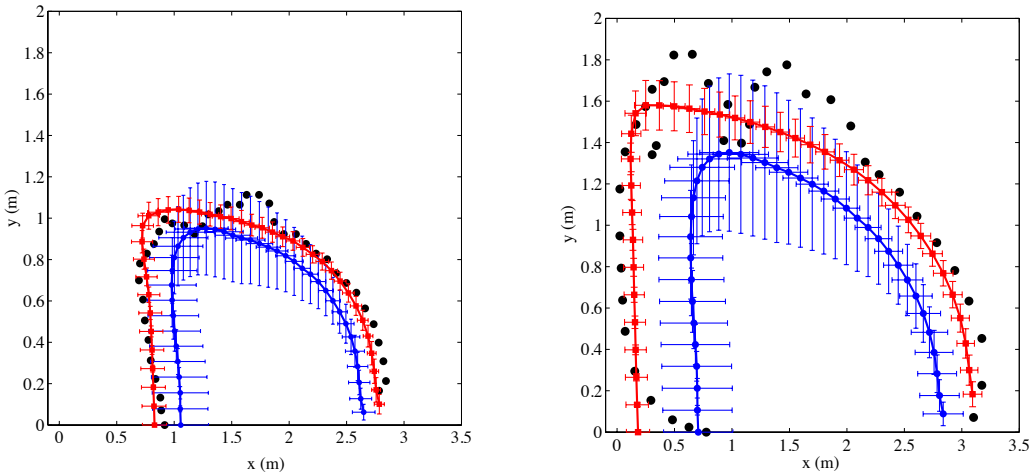
Figure 2.6: Comparison between simulated and measured fire front positions [32]. EKF applied on two control parameters: the fuel moisture content M_v and the fuel surface-to-volume ratio Σ_v.

correction of the parameter only allows a global correction of the fire front. The topology of the front cannot be changed, meaning that local changes of the fire line are out of reach with this approach.

In these case of parameter estimation, although the mean distance between the analysis fire front and the observations is reduced compared to the mean distance to the prior fire front, the front curvature at small-scales in not retrieved. To improve this correction, we have two options:

- Consider Rothermel's model parameters as function of the two spatial coordinates as the fuel layer thickness δ and the rate of spread Γ. With this approach, the size of the control vector would be dramatically increased, causing very expensive resolution. Besides, we would be correcting parameters values at some mesh nodes far away from the front or within the burnt zone where there is no required correction. In a way, we would be able to limit the size of the problem by applying DA algorithm only considering on mesh nodes close to the fire. Still, spatially-distributed parameters are difficult to validate: in real cases, such high-resolution parameter distribution is not available and therfore there is no mean of assessing if the analysis values are consistent with the physics.

- The second option will be to consider that, if it is not possible to identify and dissociate the sources of uncertainty in the model, it is easier to deal with the consequences of those uncertainties, ie. by considering the state variables as control variables, here the reaction progress variable c. By directly correcting the state, the errors from the modeling are taken into account with also other sources of errors like numerical aspects of the resolution, e.g. discretization. Finally, this option leads to more general methods for state estimation



(a) Assimilation at $t = 78$ s

(b) Assimilation at $t = 106$ s

Figure 2.7: Comparison between simulated and measured fire front positions. EnKF applied on four control parameters: the fuel moisture content M_v, the fuel surface-to-volume ratio Σ_v and the wind magnitude and direction $\mathbf{u_w}$. Black-dotted line corresponds to the observed front, blue-squared line corresponds to the mean forecast front and the red-squared line corresponds to the mean analysis front. The ensemble-based approach allows to compare the standard deviation of each points localization: horizontal and vertical lines around the front markers.

2.3.2 State estimation applied to the 2-D temperature field

Data assimilation applied to a state estimation problem is used by J.Mandel and J.Beezley [35]. Because they aimed at building a real-time coupled atmospheric-wildland fire modeling system, the surface wildfire spread model is forced by the wind and radiated heat field. They formulate the wildfire propagating as reaction-diffusion PDE where the temperature and the amount of fuel are evolved in time.

In first works, DA consists in a standard EnKF implementation in which the control vector includes the two fields Y (fuel) and T (temperature) at each mesh nodes. Randomization of data is used to generate a wide range of prior and observation members where the state fields are not only perturbed in magnitude, but also spatially shifted, as illustrated in Fig. (2.8).

The results obtained with this approach shows good agreement on the ensemble mean with the reference solution they wish to recover even if the simulation ensemble was ignited intentionnally far away from the reference ignition region. However, sometimes the ensemble is attracted to reference solution, and sometimes not, depending on if there exists a good match to the data. So the standard EnKF remains limited for several reasons:

- This approach correct mainly the field magnitude and less the localization of the fire flame. This results, when simulated members are far away from the data, in non-physical state,

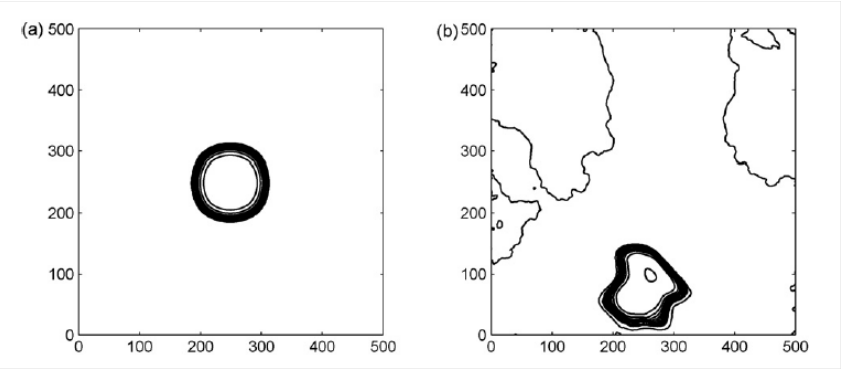


Figure 2.8: Contour plots with 100K between contour lines. (a) A reference temperature profile of a circular ignition region in the center of domain. (b) The same profile randomly perturbed in magnitude and spatially, from Mandel.J et al. work [35]

- Above all, the EnKF relies on the Gaussian assumption when the PDFs of the temperature fields are quasi-bimodal around the burning and fresh temperature.

This bimodal behavior motivates Mandel and Beezley to develop a new mathematical tool to manipulate Gaussian variables within the EnKF implementation while still undirectly distorting and deforming the state fields. The resulting algorithm is called the Morphing Ensemble Kalman filter (MEnKF, [58]). This update of the EnKF is based on image processing techniques called registration, i.e. the method of moving and streching an image to turn it into another (Fig. (2.9)). The original idea is to project all the temperature fields in a same space to be able to compare them. The Morphing Ensemble Kalman filter principle is then to apply the EnKF to the ensembles of morphing application. While the PDFs of the temperature fields are bimodal, the morphing applications were generated in a way that makes their PDF Gaussian.

In these two strategies, the main objective was to be able to correct the fire front position under the constraint of correcting a whole 2-D field. By estimating the 2-D field amplitude, we are able to correct the magnitude but not the location of the fire when the data and prior members are too far away from each other. Even with the MEnKF that allows better correction of the fire front positions, it still implies the correction of the whole 2-D field and an expensive nonlinear optimization algorithm. A new strategy based on directly correcting the fire front positions would avoid correcting the whole 2-D field magnitude. This is the subject of the internship and is the subject of the next part.

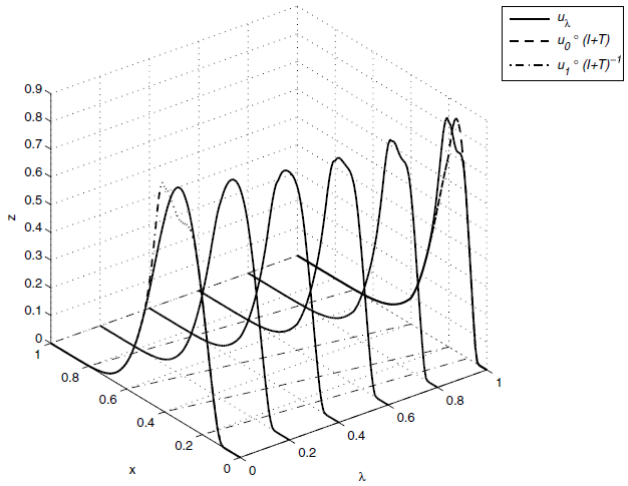


Figure 2.9: Example of morphing procedure in a one-dimensional case. The thin dash-dotted lines in the horizontal plane connect the nodes x of the morphing grid with the values of nodes after applying the registration mapping.

2.4 Conclusion

DA is applied in many fields of geosciences, from hydrology to atmospheric chemistry, and more generally in all topics related to control theory. Its extension to wildfire propagation is the long-term objective of this project.

DA allows to statistically treat a priori knowledge from a numerical model by combining it to experimental measurements when they get available. The purpose of this technique is to approach as accurately as possible the true state so as to decrease the uncertainties in the predicted states. This can be done either by controling the initial condition like in weather forecasting, or by controlling the physical properties and boundary conditions of the model. Those two approaches have both advantages and limitations and so, data assimilation controlling both state and parameters can be an interesting work as shown by Moradkhani in hydrology, see [59].

Chapter 3

Tracking the fire front location and shape using the Ensemble Kalman filter

The objective of this study is to develop a prototype for a data-driven wildfire simulation platform capable of forecasting the fire spread dynamics. The prototype features the following main components: a level-set-based fire propagation solver that adopts a regional scale approach, treats wildfires as propagating fronts, uses a description of the local rate of spread (ROS) of the fire based on Rothermel's model, described in Chapter 1; a set of measurements of the real fire front described in this current chapter 3 section 1.1 (paragraph Observations) ; and a data assimilation algorithm based on an Ensemble Kalman Filter (EnKF). This study is a continuation of our previous studies presented in Refs. [32, 34, 33] and an extension to the case of spatially-varying vegetation properties and temporally-varying wind conditions. This extension is based on a state estimation approach introduced in Chapter 2, illustrated in Figure (2.2) and inspired in part by previous studies by Mandel *et al.* [35, 58]. In the present study, the control variable is the fire front position and its related error is characterized by an approximate Gaussian PDF, which allows for a straightforward application of EnKF. The position and the shape of the flame front of a surface wildfire is directly represented by the x- and y-coordinates (for 2-D case) of its discretization markers. In this present Chapter 3, we present the mathematical formalism associated with the estimation of the fire front positions with a wildfire spread numerical model in section 1. Section 2 describes how the ensemble for the EnKF is generated. Section 3 provides some sensitivity study for the DA prototype as well as test case results showing how the ensemble generation impacts the analysis. Section 4 is dedicated to more realistic experiments that will emphasis some limitations of the EnKF algorithm correction, due to the incompleteness of observations for instance. Finally, we will present a real-case study test as the assimilation of a grassland controlled burning experiment.

3.1 State estimation problem for wildfire spread

3.1.1 Identification of the main variables and parameters

Control vector for state estimation

The fire front is numerically discretized in N_{fr} points over the two-dimensional computational domain with each markers of the fire front represented using 2-D x- and y-coordinates (plain line in Fig. (3.1)). The control vector includes the x- and y-coordinates of the N_{fr} points in a vector of size $2N_{fr}$ elements:

$$\mathbf{X} = \begin{pmatrix} x_1 & x_2 & \dots & x_{N_{fr}} & y_1 & y_2 & \dots & y_{N_{fr}} \end{pmatrix}^T. \tag{3.1}$$

In the following:

- $\mathbf{X}^f$: "f" as "forecast" to denote a background fire front (prior values),
- $\mathbf{X}^a$: "a" as "analysis" to denote a fire front obtained after an assimilation process (posterior values),
- $\mathbf{X}^t$: "t" as "true" to denote the reference front in synthetic-data experiment (where the observations are generated from a reference “true” front),

Observations

There is a growing body of literature on recent technological developments for geo-referenced wildfire front tracking [60, 61]. So far, spaceborne and airborne systems observe fires in the Middle InfraRed (MIR) region but only airborne platforms provide spatial and temporal resolution suitable for real-time geo-location of active fire contours. For instance, Paugam *et al.* [60] show that spatio-temporal variations of the rate of spread of the fire can be accurately retrieved using a Fire Radiative Power (FRP) analysis on a thousand-meter-square controlled fire experiment. As illustrated in Figure (3.1), the observed fire front is represented as a segmented line using a pre-defined number N^o_{fr} of equally-spaced markers (the observation points); the observation vector, noted $\mathbf{Y}^o$, contains the two-dimensional coordinates (x^o_i, y^o_i) of the fire front markers. The observation vector $\mathbf{Y}^o$ is a vector of size $2N^o_{fr}$ elements such that:

$$\mathbf{Y}^o = \begin{pmatrix} x^o_1 & x^o_2 & \dots & x^o_{N^o_{fr}} & y^o_1 & y^o_2 & \dots & y^o_{N^o_{fr}} \end{pmatrix}^T. \tag{3.2}$$

The FIREFLY solver uses a high-resolution computational grid that allows for a detailed representation of the local conditions while observations of the fire front position are likely to be provided with a much coarser resolution, thus we may expect N^o_{fr} to be much lower than N_{fr}. In the following, we assume for simplicity that $N^o_{fr} = (N_{fr}/r)$ where r is an integer taking values larger than 1. The observation operator $\mathcal{H}$ may be defined in several ways (for instance using a projection scheme) but preliminary tests have shown that a simple selection treatment taking 1 out of every r points provided reasonable results. The observation operator $\mathcal{H}$ is then reduced to a selection operator that matches a subset of N^o_{fr} markers among the N_{fr} of the simulated front

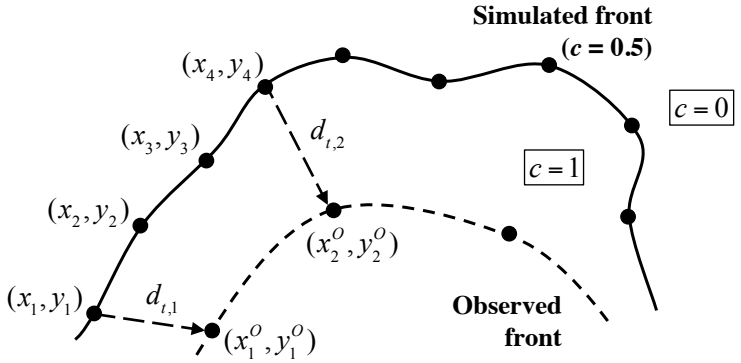


Figure 3.1: Formulation of the innovation vector as a distance between the observed and simulated fronts.

with each one of those in the observed front, as illustrated in Figure 3.1. Finally, the innovation vector is formulated as the distance between the markers of the observed front and the selected one of the simulated front:

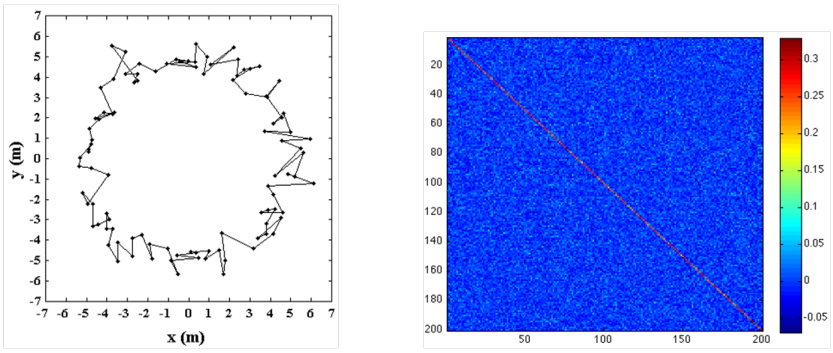
$$\mathbf{d} = \mathbf{Y}^o - \mathcal{H}(\mathbf{X}^{\mathbf{f}}). \tag{3.3}$$

Background and observation error covariance matrices

The background error covariance matrix, $\mathbf{P}_e^f$ is calculated from the ensemble of simulated fronts, see Eq. (2.35). With the current control vector definition in Eq. (3.1), the background error covariance matrix (and all covariance matrices in general) can be divided in 4 parts:

- The diagonal gathers the variances of the errors on each coordinate. The N_{fr} first elements of the diagonal are the variances of the x-coordinate and the N_{fr} last are the variances of the y-coordinate,
- the first diagonal bloc gathers the covariances of the errors on x-coordinate with the other x-coordinates,
- the second diagonal bloc gathers the covariances of the errors on y-coordinate with the other y-coordinates,
- and the extra-diagonal blocs gather the crossed-covariances between the errors on the $x-$ and $y-$coordinates.

The stochastic approach also allows to estimate $\mathbf{P}_e^f\mathbf{H}^T$ and $\mathbf{H}\mathbf{P}_e^f\mathbf{H}^T$. Because the observation operator $\mathcal{H}$ is a selection operator where points on the simulated fronts are selected every r points, $\mathbf{P}_e^f\mathbf{H}^T$ and $\mathbf{H}\mathbf{P}_e^f\mathbf{H}^T$ are respectively defined by selecting from $\mathbf{P}_e^f$ the columns and lines-columns associated with the selected markers in the simulated front. $\mathbf{P}_e^f$ is a matrix of size $2N_{fr} \times 2N_{fr}$ while $\mathbf{H}$ is of size $2N_{fr}^o \times 2N_{fr}$; meaning that $\mathbf{P}_e^f\mathbf{H}^T$ is of size $2N_{fr} \times 2N_{fr}^o$ and $\mathbf{H}\mathbf{P}_e^f\mathbf{H}^T$ of size $2N_{fr}^o \times 2N_{fr}^o$.



(a) Example of prior front (b) Covariances for 200 members

Figure 3.2: Background characteristics when each discretization marker is independently perturbed: front with 100 discretization marker (a) and $\mathbf{P}_e^f$ of size $(2 \times 100) \times (2 \times 100)$ elements determined with 200 members (b).

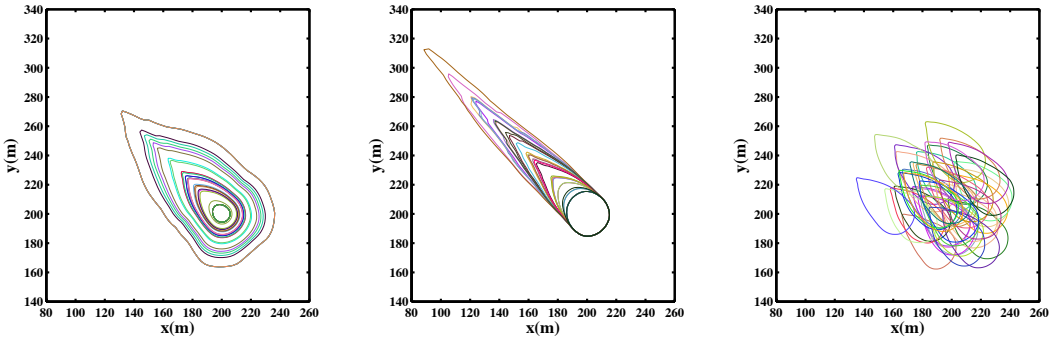
The observation error covariance matrix $\mathbf{R}$, of size $2N_{fr}^o \times 2N_{fr}^o$, is a diagonal matrix defined as in Eq. (2.36).

3.1.2 Implementation of the EnKF for the fire front correction

Generation of the ensemble

In the EnKF implementation, the first step is to generate the background ensemble for each control variable taking into account all sources of uncertainties. For front-tracking application, this would mean taking each markers coordinate and generate the ensemble for each one separately: from a reference front discretized in N_{fr} markers, the position of each marker sould be randomly perturbed adding a Gaussian noise with a zero expectation and a significant standard deviation to the x- and y- coordinates. When each points is perturbed separately, the resulting has a chaotic aspect (Fig. (3.2a)), and there is no smooth line that can link all the markers together since by construction the error on one marker is uncorrelated with the error of its neighbors, as illustrated in (3.2b). This method to generate the members is conflicting with the definition of a fire line that has to be coherent with respect to the equation described in Part 1. With this ensemble generation approach, the background members can not be considered as firelines and it will make no sense to try to recover the true front from them.

Thus, another strategy to generate the members is required. The idea is to directly perturb the model parameters as Rochoux [2], Delmotte [3] and Bart [4] did for parameter estimation: the parameters of the ROS model are perturbed with a Gaussian noise. Figure (3.3) presents the fire fronts shapes when the fire is spreading on an uniform fuel with a moderate wind blowing from the South East direction, for different conditions of fuel moisture content M_v (3.3a), wind velocity u_w (3.3b) and ignition location (x_{ign}, y_{ign}) (3.3c).



(a) Varying M_v (b) Varying $\mathbf{u_w}$ (c) Varying (x_{ign}, y_{ign})

Figure 3.3: Examples of various fronts when varying only one control parameters: (a) the fuel moisture content M_v, (b) the wind velocity $\mathbf{u_w}$, (c) the ignition location (x_{ign}, y_{ign})

The vector of all perturbed parameters will be denoted as $\mathbf{p}_m$:

$$\mathbf{p}_m = \left[x_{ign}, y_{ign}, \delta_v(x,y), M_v, M_{v,ext}, \Sigma_v, m_v^{''}, \rho_p, \Delta h_c, \mathbf{u}_w(t)\right] \tag{3.4}$$

- $\mathbf{p}_m^t$ holds for the true parameters used in synthetic-data experiments,
- $\mathbf{p}_m^f$ holds for the perturbed background parameters.

In the framework of OSSE experiments, a set of true parameters $\mathbf{p}_m^t$ is chosen and used to generate the set of background and observation data so that the uncertainties are completely controled. The implementation of the OSSE with FIREFLY is presented in Fig. (3.4). The set of true parameters is used for two purposes:

- First, it is perturbed to generate the set of background parameters $\mathbf{p}_m^f$. These parameters are then input in the model integration code $\mathcal{M}_{[t-1,t]}$ to generate the set of prior fronts $\left\{\mathbf{X}^f\right\}$,
- second, it is directly input in the model integration code to generate the true fire front $\mathbf{X}^t$. The observed fire front $\mathbf{Y}^o$ is then extrated from $\mathbf{X}^t$ by selecting N_{fr}^o points ($\mathcal{H}(\mathbf{X}^t)$) and adding noise ($\epsilon^o$).

In the framework of real-data experiment, whose general implementation is presented in Fig. (3.5):

- The set of parameters $\mathbf{p}_m$ is known from an a priori value $\overline{\mathbf{p}_m^f}$ and an estimation of the uncertainty $\sigma_{\mathbf{p}_m^f}$. The forecasted fire fronts are therefore generated from this a priori information.
- Besides, the observed fire front $\mathbf{Y}^o$ is already provided with an estimation of the uncertainty σ^o.

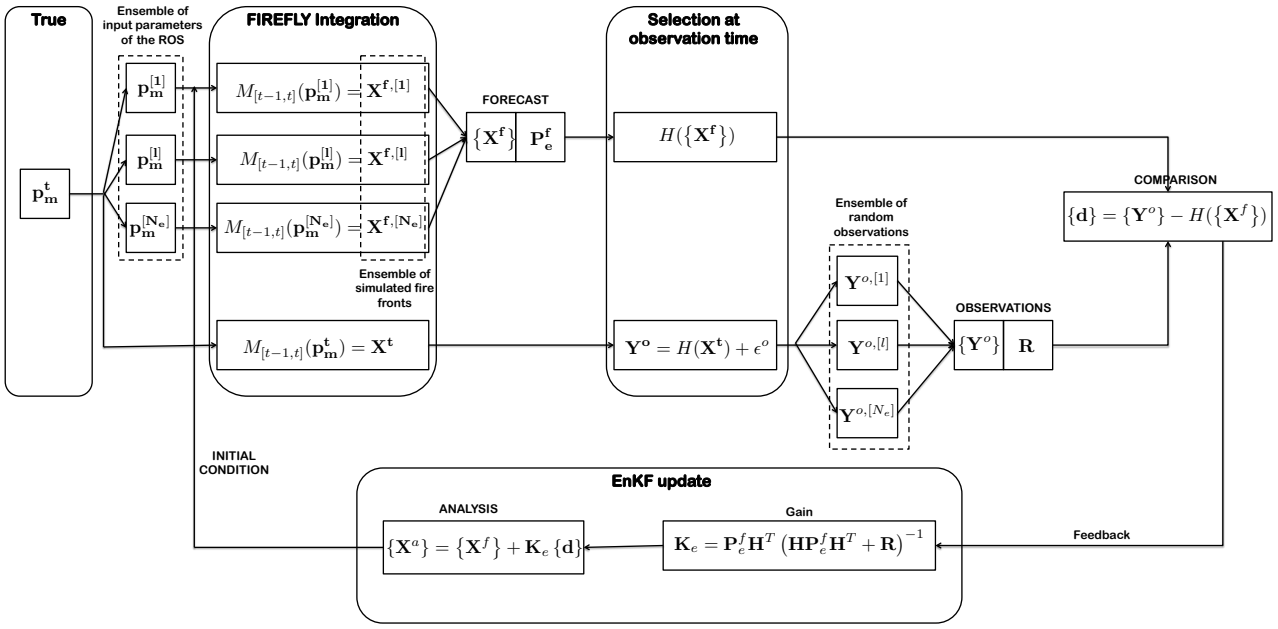


Figure 3.4: OSSE algorithm to correct the ensemble of simulated fire fronts

For the two approaches, the ensemble of background fire fronts and the observation vector are then similarly exploited with:

1. the randomization of the observed front $\{\mathbf{Y}^o\}$,
2. the determination of the innovation vectors $\{\mathbf{d}\}$,
3. and finally, the computation of the EnKF update.

Implementation of the EnKF for FIREFLY

In the framework of this intership, the coupled execution of the front-tracking solver with the different instances of the EnKF is managed by the OpenPALM coupler [62, 63, 64] developed at CERFACS and ONERA. Moreover, the OpenPALM functionality Parasol handles the paralell execution of the independant integrations of each member.

OpenPALM The software OpenPalm [62, 63, 64] is a flexible and powerful dynamic coupler developed at CERFACS since 1998. It was originally developed for the MERCATOR project in the framework of operational oceanography forecast [65]. OpenPALM can be used to couple different numerical models; e.g an ocean model with an atmospheric model for climate modeling; or to couple a physical model with a given algorithm; e.g. optimization or post-treatment algorithm. Now, it is widely used by the scientific and engineering community for data assimilation aplications such as the French project ADOMOCA (Assimilation de DOnnées pour les MOdèles de Chimie Atmosphérique) in atmospheric chemistry data assimilation [66] or DA applications

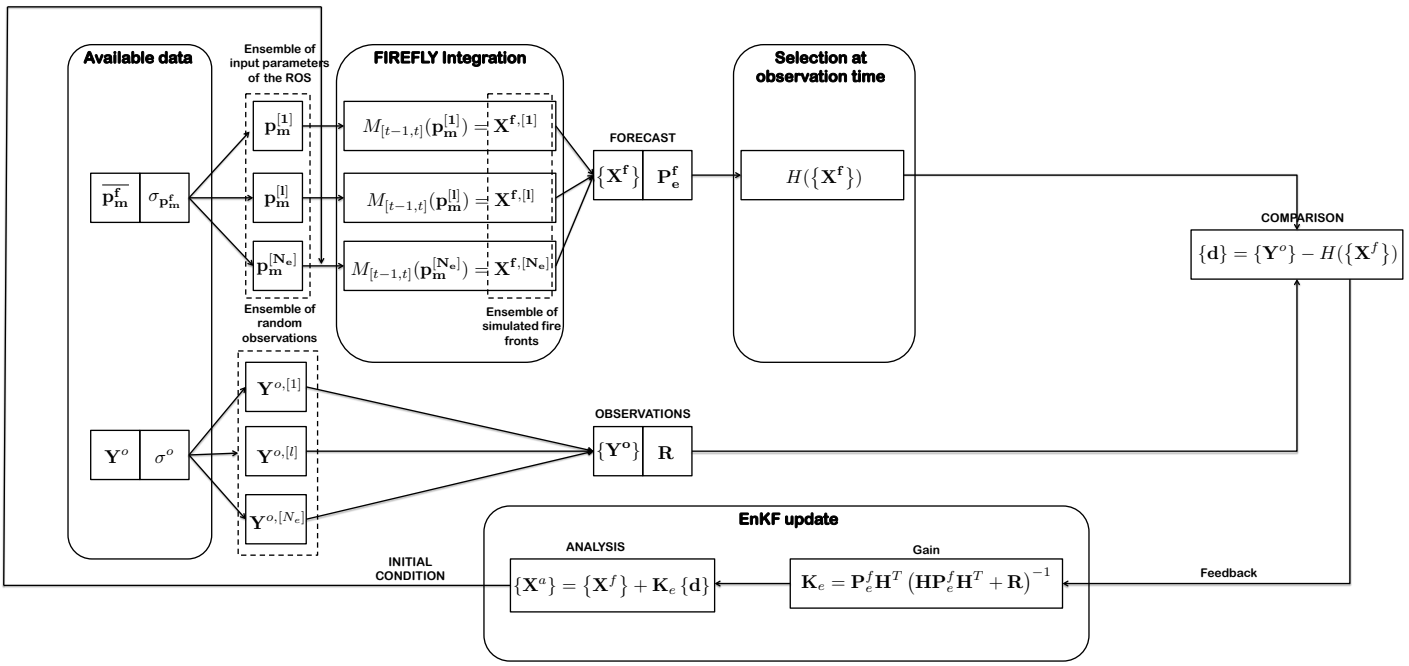


Figure 3.5: Real-data experiment algorithm to correct of the ensemble of simulated fire fronts

with the Saint-Venant code MASCARET for flood forecasting developed at SCHAPI [67]. OpenPalm is also used as a model coupler by the CFD team at CERFACS with the software ELSA in fluid structure applications [68, 69]. More recently, OpenPALM has been used in wildland fire spread modeling [33, 34].

OpenPALM is a dynamic coupler: a coupled component can be launched and can release resources upon termination at any moment during the simulation. Computing resources such as the required memory and the number of concurrent processors, are handled by the OpenPALM driver. An OpenPALM application can be described as a set of computational units arranged in a coupling algorithm. The different units are controlled by conditional and iterative constructs and belong to algorithmic sequences called computational branches. A branch is structured like a program in a high level programming language: it allows the definition of sequential algorithms. Inside a branch, the coupled independent programs, the units, are invoked as if they were subroutines of the branch program. It is then possible to define communication points between units from different branches to exchange, send and receive data. OpenPALM applications are implemented via a graphical user interface (GUI) called PrePALM. In this interface, the programmer initially defines the coupling algorithm: number of components (units), sequential (units within the same branch) and parallel (separated branches) sections, resources management. Then, the actual commnications between communications are materialized by points at the top of the unit box (meaning received data) or at the bottom (meaning dent data) linked between units by dashed-line, see Fig. (3.6). For further information, the reader should refer to [70].

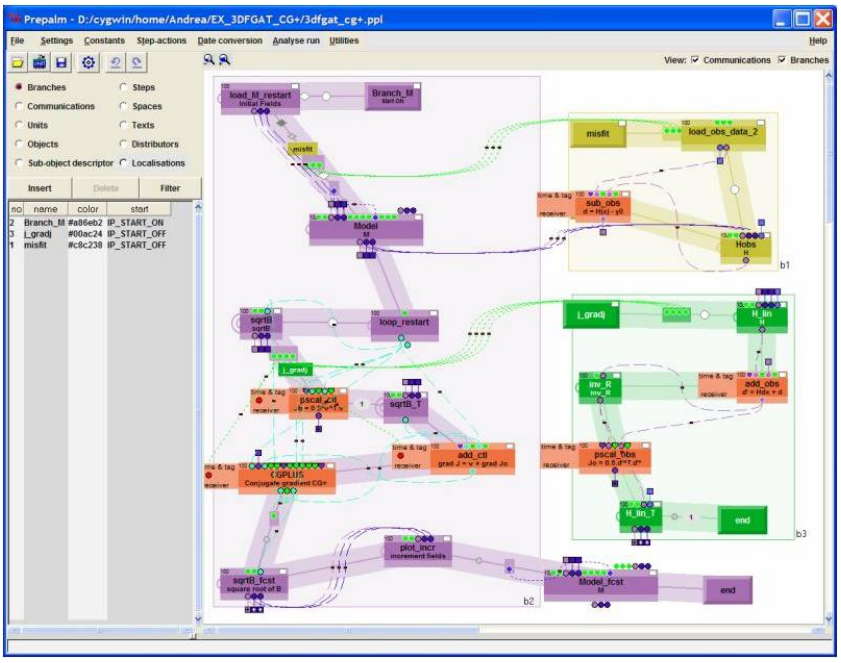


Figure 3.6: Snapshot of the G.U.I Prepalm

Parasol Parasol is an functionality of the OpenPALM software allowing to automatically launch in parallel a certain number of instances of the same program. Its spawns n instances of the same source code with different inputs (and consequently outputs). The user sets the number of allocated processors and then Parasol, executes in parallel the different instances. During the execution, it manages the gathering and the distribution of the output data in the correct order. In the context of this internship, the Parasol functionality is applied to the function, named Firefly_forward, that integrates the fire spread model in order to generate the ensemble of background fire fronts.

Parasol creates two types of subroutines:

- A *slave* subroutine in charge of executing the function to "Parasol-ize",
- A *master* subroutine which calls n instances of the slave subroutines and manage the input and output parameters.

Figure (3.7) presents the general execution of the Parasolized function Firefly_forward:

1. First, the *master* gathers the set of all input parameters,
2. the *master* calls the *slave* subroutine using the *slave* processors that have been allocated to it. For each call, it supplies a different set of input parameters,
3. each *slave* instance calls and executes the function to Parasol-ize with their own input parameters and returns the output parameters to the *master*,

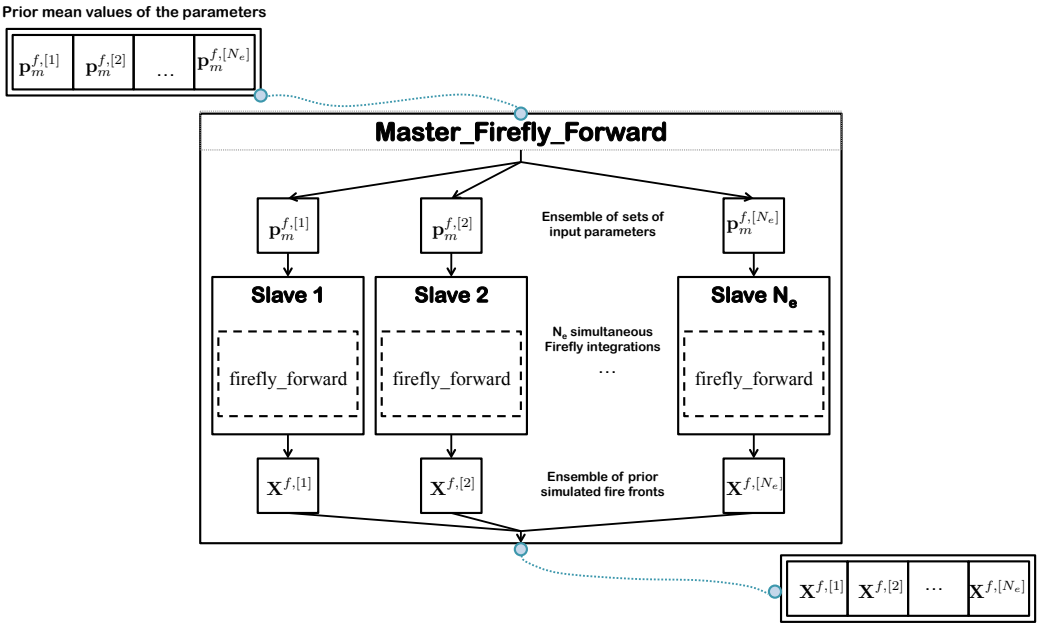


Figure 3.7: Palm unit *master* applied to the Firefly_forward function in the FIREFLY algorithm with Parasol

4. the *master* sends the ensemble of outputs from all *slave* instances for the next step of the algorithm.

3.1.3 Cycling of the analysis

After one assimilation cycle, analysis fire fronts described with N_{fr} pairs of coordinates are available. To continue with DA at the next time window, the analysis fronts have to be propagated until the next observation time. From Chapter 1, we recall that the level set solver produces and takes as initial condition a two-dimensional full field of the progress variable c while DA manipulates set of coordinates that represent the front. Fig. (3.8) presents the sequence of the different structures of the state variables during the assimilation. Then we need pretreatments to map from one structure to another and conversely. In the first sense, an extraction function build the fire front from the progress variable field. Therefore, an algorithm is required to construct the progress variable field from the analysis fire front $\mathbf{X}^a$. The reconstruction algorithm to apply to each mesh nodes leans on vectorial analysis and is is detailled in Annexe A.

3.1.4 A posteriori diagnostics

To quantify the quality of the correction, we will use the Root Mean Square (RMS) distance between the background/analysis fronts and a reference front:

- When the true front is taken as reference, the terms BMT - Background Minus True (resp. AMT - Analysis Minus True) are employed for the comparison between the true front and

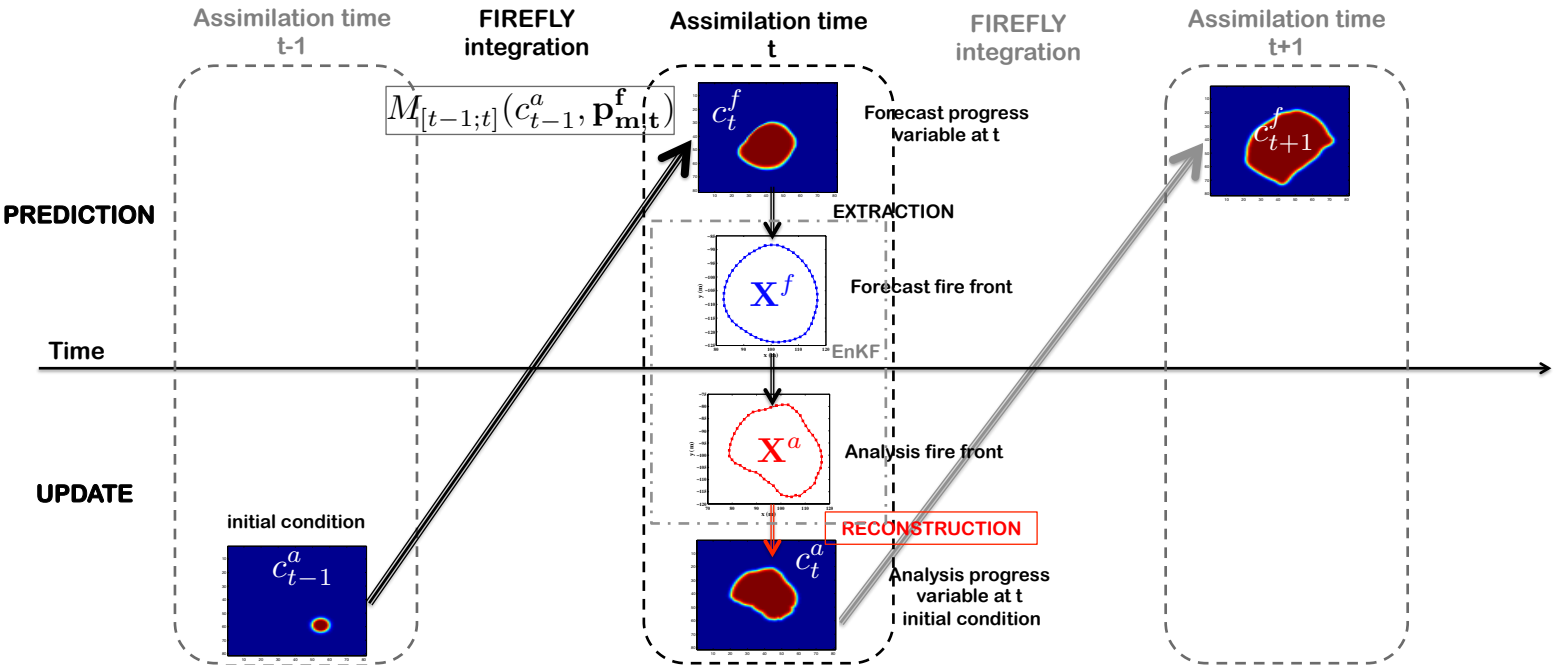


Figure 3.8: Sequence of the structures of the state variables: two-dimensional progress variable c to integrate the model and discrete fire fronts for the EnKF. Special functions are required to pass from one nature to another: the extraction function is for DA update and the reconstruction function is for the cycling to the next assimilation time.

the background (resp. analysis) fronts. This can only be applied for twin experiments and is favored for this king of experiment,

- and when the observed front is taken, the terminology becomes BMO - Background Minus Observation/AMO - Analysis Minus Observation. This one is used for real-data experiment.

Statistics are defined as the pair $(\overline{d}, \overline{\sigma})$ such that:

- $\overline{d}$ (m) is the mean over all the points along the front of the mean distance over the members between points on the simulated fronts and corresponding points along the reference front:

$$\overline{d_{BMT}} = \frac{1}{N_{fr}} \sum_{j=1}^{N_{fr}} \overline{d_{BMT,j}} \tag{3.5}$$

with

$$\forall j = 1 \dots N_{fr} \quad \overline{d_{BMT,j}} = \frac{1}{N_e} \sum_{l=1}^{N_e} \sqrt{(x_j^{f,[l]} - x_j^t)^2 + (y_j^{f,[l]} - y_j^t)^2}$$

It is similar for $\overline{d_{AMT}}$ by taking $(x_\bullet^{a,[\bullet]}, y_\bullet^{a,[\bullet]})$ instead of $(x_\bullet^{f,[\bullet]}, y_\bullet^{f,[\bullet]})$

- $\overline{\sigma}$ (m) is the mean over all the points along the front of the standard deviation of the distance between points on the simulated fronts and corresponding points along the reference front:

$$\overline{\sigma_{BMT}} = \frac{1}{N_{fr}} \sum_{j=1}^{N_{fr}} \sigma_{BMT,j} \tag{3.6}$$

with

$$\forall j = 1 \dots N_{fr} \quad \sigma_{BMT,j} = \sqrt{\frac{1}{N_e - 1} \sum_{l=1}^{N_e} \left(\sqrt{(x_j^{f,[l]} - x_j^t)^2 + (y_j^{f,[l]} - y_j^t)^2} - \overline{d_{BMT,j}} \right)^2}$$

The assimilation produces a good-quality analysis when:

$$\overline{d_{AMT}} < \overline{d_{BMT}} \tag{3.7}$$

meaning that, on average, the points from the analysis fronts are closer to the true front than the points along the background fronts.

Besides, from an ensemble point of view, the EnKF is considered satisfactory when the ensemble standard deviation is reduced:

$$\overline{\sigma_{AMT}} < \overline{\sigma_{BMT}} \tag{3.8}$$

meaning that the analysis fronts are close to each others and have similar curves in opposition with the background ensemble where the background fronts can be far away from each other and have a wide range of different curves.

Section	Name	Card($\mathbf{p_m}$)	Fuel type	Nb cycle (AT/FT)[1]	$N_{fr}/N^o_{fr}/N_e$	σ^o (m)
3.2.1	UNIF2_Pe_exp1	2	Uniform	1 (100 s/200 s)	100/1/25	1
3.2.1	UNIF2_Pe_exp2	2	Uniform	1 (100 s/200 s)	100/1/25	1
3.2.2	ZONE10_Deform	10	Zones	1 (150 s/300 s)	100/20/50	1
3.2.2	ZONE10_Ne	10	Zones	1 (150 s/300 s)	100/20/-	1
3.3.1	UNIF2_σ_o	2	Uniform	1 (100 s/200 s)	100/1/25	-
3.3.2	RAND2_Nfr_o	2	Random	1 (200 s/400 s)	100/-/25	1
3.3.2	ZONE10_Lim_MissingData	10	Zones	1 (150 s/300 s)	100/12/20	1
3.3.3	ZONE10_Lim_ForecastPerf	10	Zones	5 (150 s/600 s[2])	100/20/20	1

Table 3.1: Description of the simulations. The names gather important information about the simulations: the type of fule bed, the number of parameters used to generate the fire front and the variables of interest for the particular simulation

3.1.5 Experiment settings

Now that the DA framework is settled, the following sections 2 to 4 of this part present the main results and interpretation of the study. A series of synthetical-data experiments are studied to emphasis the contribution of each features of the DA framework and to evaluate the performances of the EnKF. For the following simulations, the fire is initialized as a circular front of radius 5m and with a given ignition position (x_{ign}, y_{ign}) which corresponds to the coordinates of the center of the initial circle. The details of all the run simulations are presented in Table (3.1).

3.2 Description of the background information

The objectives of these first experiments is to show the different contributions and effects of the background error covariance matrix on the DA analysis. In order to do so, we will carry out OSSE experiments before moving on to real experiments.

3.2.1 Source of uncertainties in the stochastic estimate of the background error covariance matrix

To begin with, we assume that the only source of uncertainty between the true front and the simulated fronts is the localization of the ignition position, i.e.:

$$\mathbf{p}_m = \{x_{ign}, y_{ign}\} \tag{3.9}$$

- The true front is then initialized with $\mathbf{p}^t_m = \{100\text{m},\ 100\text{m}\}$ and has a circular shape,
- while the mean background values are chosen as $\mathbf{p}^f_m = \{97\text{m},\ 103\text{m}\}$ with a standard deviation of 10 m on each coordinate.

[1] AT for "Analysis time", i.e the time at which the analysis is performed. FT for "Free-run time", i.e. the time until which the model is integrated without any assimilation. The free-run fire fronts can be used to compare the model performances with and without DA.

[2] The use of the free-run times will be detailled in section 3.3.3.

Theoretical and stochastic estimate properties of the background error covariance matrix

The contribution of the background error covariance matrix is studied with a simple simulation based on a uniform fuel distribution and with a simplified rate of spread as a function of the fuel depth only. Because of the uniformly-distributed fuel, the rate of spread is therefore isotropic and the simulated fire front keeps its initial circular shape. Therefore, all the forecast fronts can be written as a function of the ignition location (x_{ign}, y_{ign}) such that

$$\forall l = 1 \dots N_e \quad \mathbf{X}^{f,[l]} = \begin{pmatrix} x_{ign}^{[l]} + r_t \cos(\theta_1) \\ x_{ign}^{[l]} + r_t \cos(\theta_2) \\ \vdots \\ x_{ign}^{[l]} + r_t \cos(\theta_{N_{fr}}) \\ y_{ign}^{[l]} + r_t \sin(\theta_1) \\ y_{ign}^{[l]} + r_t \sin(\theta_2) \\ \vdots \\ y_{ign}^{[l]} + r_t \sin(\theta_{N_{fr}}) \end{pmatrix} \tag{3.10}$$

with r_t the radius of the circular fire front at the assimilation time t, identical for all the members. Also, note that we assume here that markers along the fire line with the same index on the simulated fire fronts are taken at the same location on the circle (meaning the same angle $\theta_i,\ i = 1 \dots N_{fr}$) for each member.

Then, the difference between each simulated fire fronts and the mean front sums up to a difference between the perturbated ignition locations $(x_{ign}^{[l]}, y_{ign}^{[l]}),\ l = 1 \dots N_e$ and the mean location one over the members $(\overline{x_{ign}}, \overline{y_{ign}}) = (\frac{1}{N_e}\sum_{l=1}^{N_e} x_{ign}^{[l]}, \frac{1}{N_e}\sum_{l=1}^{N_e} y_{ign}^{[l]})$:

$$\begin{array}{l} \forall l = 1 \dots N_e \\ \qquad \begin{array}{lrcl} \forall i = 1 \dots N_{fr} & \mathbf{X}^{f,[l]} - \overline{\mathbf{X}^f} & = & x_{ign}^{[l]} - \overline{x_{ign}} \\ \forall i = (N_{fr}) + 1 \dots 2N_{fr} & \mathbf{X}^{f,[l]} - \overline{\mathbf{X}^f} & = & y_{ign}^{[l]} - \overline{y_{ign}} \end{array} \end{array} \tag{3.11}$$

Theoretically, the stochastically estimated background error covariance matrix is constant in each quarter such that:

$$\mathbf{P}_{e,th}^{f} = \frac{1}{N_e - 1} \begin{pmatrix} \left[\sum_{l=1}^{N_e} \left(x_{ign}^{[l]} - \overline{x_{ign}}\right)^2\right].\mathbf{1}_{N_{fr}\times N_{fr}} & \left[\sum_{l=1}^{N_e} \left(x_{ign}^{[l]} - \overline{x_{ign}}\right)\left(y_{ign}^{[l]} - \overline{y_{ign}}\right)\right].\mathbf{1}_{N_{fr}\times N_{fr}} \\ \left[\sum_{l=1}^{N_e} \left(x_{ign}^{[l]} - \overline{x_{ign}}\right)\left(y_{ign}^{[l]} - \overline{y_{ign}}\right)\right].\mathbf{1}_{N_{fr}\times N_{fr}} & \left[\sum_{l=1}^{N_e} \left(y_{ign}^{[l]} - \overline{y_{ign}}\right)^2\right].\mathbf{1}_{N_{fr}\times N_{fr}} \end{pmatrix} \tag{3.12}$$

Finally, by applying the definition of correlation, Eq. (2.10), to the error covariance matrix in Eq. (3.12), we can define the associated correlation matrix $\mathbf{C}_{e,th}^{f}$. The matrix is still composed

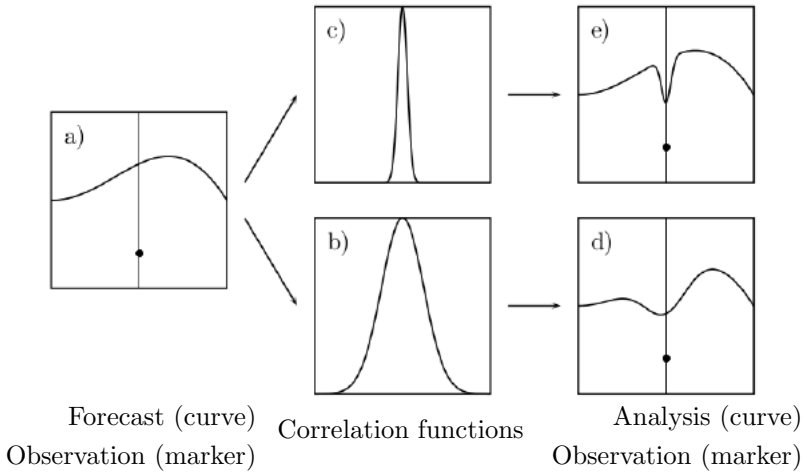


Figure 3.9: Behavior of the analysis with respect to the forecast correlation functions.

by four constant quadrants such that

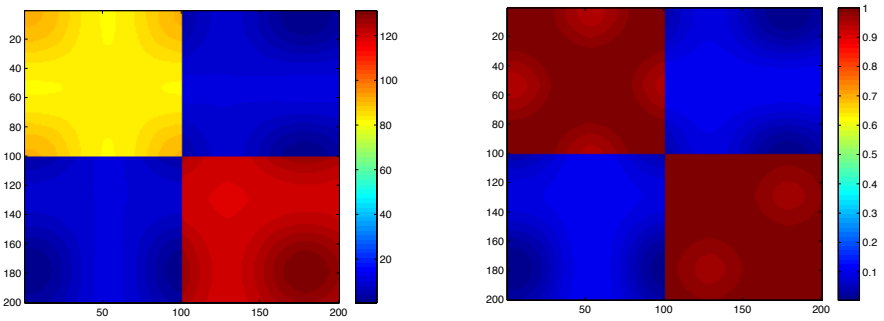
$$\mathbf{C}^f_{e,th} = \begin{pmatrix} \mathbf{1}_{N_{fr}\times N_{fr}} & \mathbf{0}_{N_{fr}\times N_{fr}} \\ \mathbf{0}_{N_{fr}\times N_{fr}} & \mathbf{1}_{N_{fr}\times N_{fr}} \end{pmatrix} \tag{3.13}$$

with the interpretation of each bloc introduced in 3.1.1.

From the correlation matrix, one-dimensional correlation functions can be extracted (a column of the matrix) representing the correlation of one coordinate of a particular marker with the x- or y-coordinate of any marker. In the present DA, the correlation functions indicate how the error on one coordinate for one marker is correlated with the error on one coordinate for the neighboring markers. This spatial impact is illustrated in Fig. (3.9):

- For the case (a-c-e): the correlation function point is narrow (c), meaning that the correlation of neighboring points quickly decreases to 0. Then, a quite reduced vicinity of the central point will be affected by the correction (e) and be moved close to the observation (large black dot in (a) and (e)).
- For the case (a-b-d): the correlation function is wider (b). Then, the zone of influence of the correction is also wider (d).

However, the matrices calculated by the Firefly-EnKF code, illustrated in Figure (3.10), are sligthly different from the theoretical ones in Eq. (3.12, 3.13) because they are not constant per quadrants. Indeed, due to inaccuracies in the isocontour algorithm (introduced in Chapter 1) that allows to establish a correspondance between the points along the fire line within the ensemble; it has been shown that there can be at the most an angular difference of 6 degrees between markers from different fronts with the same index i. The angles $\theta_i,\ i = 1 \dots N_{fr}$ are slightly different from one member to another, resulting in the oscillating variations on each quadrant of $\mathbf{P}^f_{e,\text{UNIF2\_Pe\_exp1}}$ and so of $\mathbf{C}^f_{e,\text{UNIF2\_Pe\_exp1}}$ as well.



(a) Background error covariance matrix $\mathbf{P}^f_{e,\text{UNIF2\_Pe\_exp1}}$

(b) Background error correlation matrix $\mathbf{C}^f_{e,\text{UNIF2\_Pe\_exp1}}$

Figure 3.10: Experimental background covariance and associated correlation matrices. The difference of variance between the x-variance (yellow, ~ 80) and the y-variance (red, ~ 120) is due to sampling

Behavior of the analysis with respect to the background error covariance matrix

For this OSSE, uniform and circular test case, the analysis fire fronts are determined using the theoretical background error covariance matrix at first and then the experimental matrix $\mathbf{P}^f_{e,\text{UNIF2\_Pe\_exp1}}$. The ensemble is composed of $N_e = 25$ members and the observed fire front is described by $N^o_{fr} = 1$ marker with an uncertainty defined by a standard deviation $\sigma^o = 1$ m. Indeed, with the above description of the ensemble and the uniform settings, the EnKF provides an isotropic correction over the circles, by shifting all markers together in a way that fits best the observed front. Thus, it can be assumed that only one observation marker is enough to correct the position of the fire front.

As illustrated in Fig. (3.11), the analysis fire fronts keep the circular shape of the background fire fronts. In this particular case, tracking the true front is equivalent to tracking the center of the initial location of the fire. Also, it is important to note that with this uniform definition of the background ensemble, it is impossible to deform the fire front.

The results are similar with either the theoretical matrix or $\mathbf{P}^f_{e,\text{UNIF2\_Pe\_exp1}}$. With $\mathbf{P}^f_{e,th}$, the information on the isotropy of the spread is completely contained in the diagonal blocks and the correlation functions are identically equal to 1 for correlations between the same type of coordinate and identically equal to 0 for correlations between different type of coordinate. For the simulation run with $\mathbf{P}^f_{e,\text{UNIF2\_Pe\_exp1}}$, the non-zero correlations in the extra-diagonal quadrants compensate the correlations lower than 1 in the diagonal quadrants. The Figure (3.12) presents the correlation function of the first marker with respect to the distance to the other markers (note that because the fire front is circular, the markers are equally displayed at each sides of the studied one; the positive distance is for the increasing-indexed markers and the negative distance is for the decreasing-indexed ones):

- When the dashed-red line is lower than 1 (around 0.95), the x-coordinates of markers

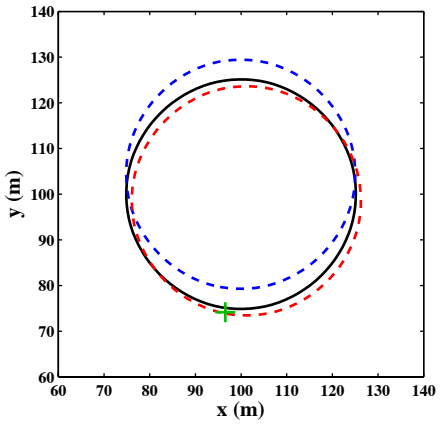


Figure 3.11: Results of an assimilation over 1 cycle with the true front (black line), the mean background/analysis fire front (blue/red-dashed line) over the members and the observation (green cross symbol).

far away are not changed with the same amplitude as the x-coordinate of the considered marker, even though there are drawn in the same direction. But simultaneously, the dotted-light blue line is at its highest (around 0.1). Meaning that when the y-coordinate of the considered marker is changed, the x-coordinates of markers far away are sligthly changed the same way.

- Similar comments can be done by comparing the orange line with the dashed-dotted blue line.

To illustrate these compensation effects, another simulation is run with a particular background error covariance matrix: the diagonal blocks are normally generated while the extra-diagonal blocks are forced to 0. The corresponding matrices are illustrated in Figure (3.13). Figure (3.14) compares the shape of an analysis fire front against a perfect circle when it is obtained with $\mathbf{P}^f_{e,\text{UNIF2\_Pe\_exp1}}$ (see Fig. (3.14a)) and with $\mathbf{P}^f_{e,\text{UNIF2\_Pe\_exp2}}$ (see Fig. (3.14b)). We see then that the analysis fire front keeps the circular shape when calculated using $\mathbf{P}^f_{e,\text{UNIF2\_Pe\_exp1}}$ but does not keep it when calculated using $\mathbf{P}^f_{e,\text{UNIF2\_Pe\_exp2}}$. In the simulation UNIF2_Pe_exp2, the compensation effect of the extra-diagonal block has been removed and then the analysis fire fronts lose the circular shape.

We will now only considere the full experimental background error covariance matrix $\mathbf{P}^f_{e,\text{UNIF2\_Pe\_exp1}}$ that we will simply denote as $\mathbf{P}^f_e$.

With these first basic simulations, we were able to emphasis the role of the error covariance matrix and its associated correlation matrix in the correction process. The correlations translate here the area of influence of each marker on the other markers. The isotropy (or anisotropy)

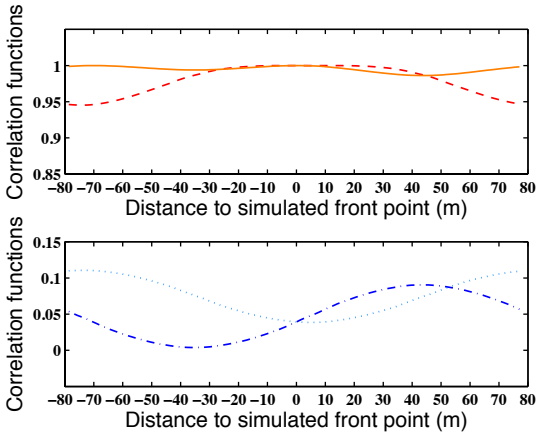
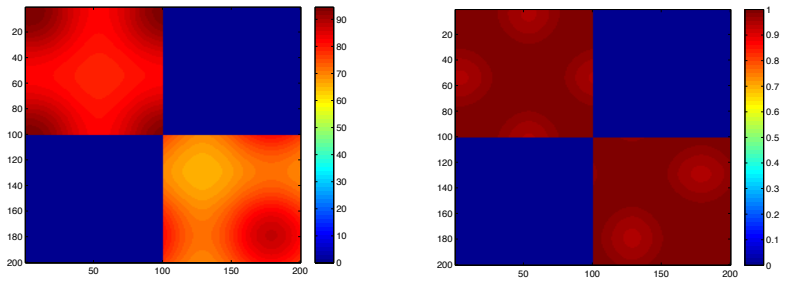


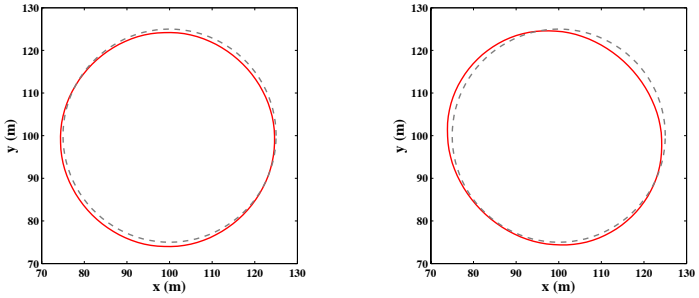
Figure 3.12: Correlation functions associated to the first marker: x-coordinate w.r.t the x-coordinates (dashed-red line), y-coordinate w.r.t the y-coordinates (plain-orange line), x-coordinate w.r.t the y-coordinates (dashed-dotted-blue line), y-coordinate w.r.t the x-coordinates (dotted-light blue line).



(a) Background error covariance matrix $\mathbf{P}^f_{e,exp2}$

(b) Background error correlation matrix $\mathbf{C}^f_{e,exp2}$

Figure 3.13: Forced experimental background covariance and associated correlation matrices.



(a) $\mathbf{P}^f_{e,exp1}$ (b) $\mathbf{P}^f_{e,exp2}$

Figure 3.14: Comparison of analysis member shape (plain-red line) with a perfect circle (dashed-grey line): (a) analysis obtained using the full experimental matrix $\mathbf{P}^f_{e,exp1}$, (b) analysis obtained using the forced experimental matrix $\mathbf{P}^f_{e,exp2}$

Input parameter	True value	Mean ensemble value	Standard deviation
$\delta_{v,1}$ (m)	0.25	0.25	0.10
$\delta_{v,2}$ (m)	1.25	1.25	0.10
$\delta_{v,3}$ (m)	0.75	0.75	0.10
$\delta_{v,4}$ (m)	1.75	1.75	0.10
M_v (%)	20	20	10
Σ_v (1/m)	11500	11500	4000
$\mathbf{u}_w$ (m/s, deg)	(1.0, 315)	(0.75, 315)	(0.15, 45)
x_{ign} (m)	350	350	20
y_{ign} (m)	350	350	20

Table 3.2: Properties of the background parameters ensemble in the spatially-varying OSSE test.

information in the spreading is included in the statistics of the matrices. To be able to stochastically represent more complex fire fronts, it is important to consider non-uniform conditions when generating the fire fronts.

3.2.2 Non uniform propagation test case: towards a full representation of the uncertainties in the ROS model

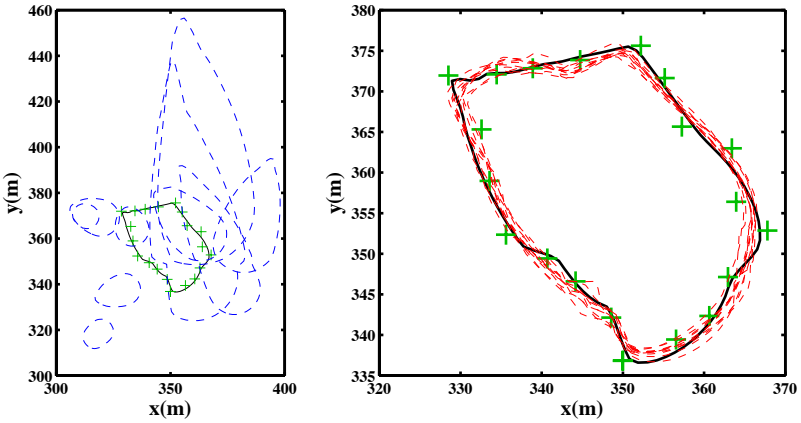
Deformation of the fire fronts using spatially-distributed parameters

In the following simulation, we want to be able to deform the simulated fire fronts to obtain more realistic analysis fire fronts. To do so, the idea is to consider all kind of uncertainties that can affect the fire spread so that the propagation is anisotropic. The strategy adopted is to consider spatially-distributed parameters. For example, the fuel is described as the 4 quadrants of the square-shaped computational domain with different value of depth. For the following series of experiments, the previous set of parameters is extended to 10 parameters so that the sources of perturbations during the propagation of the fire are extended as well:

$$\mathbf{p_m} = \{x_{ign},\ y_{ign},\ \delta_{v,1},\ \delta_{v,2},\ \delta_{v,3},\ \delta_{v,4},\ M_v,\ \Sigma_v,\ \mathbf{u_w}\} \tag{3.14}$$

with their respective true and mean/standard deviation forecast values detailed in Table (3.2).

The observed fire front is described by $N_{fr}^o = 20$ markers uniformly-distributed along the true front and with an error standard deviation $\sigma^o = 1$ m. The ensemble is composed of $N_e = 50$ members and the resulting simulated fire fronts (background/analysis) are presented in Fig. (3.15). We see that thanks to the spatially-varying parameters such as the fuel distribution, we are able to deform the simulated fire fronts and generate realistic analysis fronts close to the observed front.



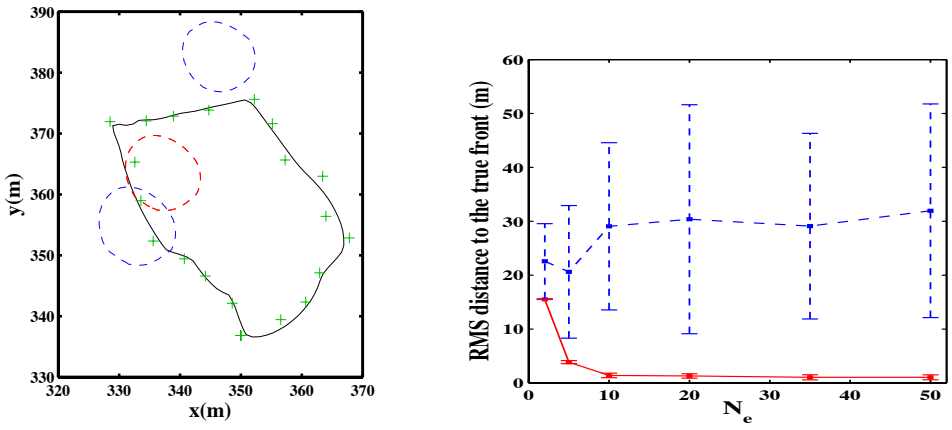
(a) Background (b) Analysis

Figure 3.15: OSSE test with anisotropic ROS. (a) Comparison between true (solid black line) and forecasted (dashed blue lines) fire front positions for $N_e = 50$. (b) Comparison between true (solid black line) and analyzed (dashed red lines) fire front positions for $N_e = 50$.

Statistics of the background ensemble

However, the spatially-distributed parameters only allow to deform the simulated fire fronts but do not guarantee correct analysis fire fronts. Indeed, Figure (3.16a) shows the forecasted and analysed fire fronts compared to the true and observed front for a low number of members ($N_e = 2$). We can see that the EnKF can barely find the shape of the fronts. This is due to the poor representation of the statistics of the background ensemble. Then, while N_e increases, the EnKF is able to catch the shape and localization of the true fire front even if the forecasted fire fronts are really different and far from it (see Fig. (3.15) with $N_e = 50$). This is also demonstrated in Fig. (3.16b) that plots the RMS distance to the true front for the background and analysis front with respect to the N_e. We can see that the analysis gets closer to the true front when N_e increases.

From the entire section 3.2, we saw that to obtain good analysis fire fronts, we need a complete background ensemble that reunites all kinds of uncertainties from the model and the numerical settings to be able to generate deformed simulated fire fronts with enough spatial variability. We also need an ensemble with enough members to calculate reasonable statistics. In the next section, we focus on the result of the assimilation and its sensibility to the observed data. We also care about the forecasting performances of the state estimation approach. The following series of simulations are still OSSE experiments to be able to validate the algorithm and test its limits.



(a) $N_e = 2$ (b) RMS

Figure 3.16: OSSE test with anisotropic ROS and varying N_e. (a) Comparison between true (black solid line), forecasted (dashed blue lines) and analysed (dashed red lines) fire front positions for $N_e = 2$ (b) RMS distance between the true and forecasted fire front positions (blue squares) and between the true and analyzed fire front positions (red squares) as a function of the number of members N_e.

3.3 Results of the assimilation

3.3.1 Sensitivity to the observation error standard deviation σ^o

As shown in section 3.2.1, for the uniform experiments UNIF2_Pe_exp1 and UNIF2_Pe_exp2 in uniform conditions, only one observation point is enough to correct the fire front given that the ensemble is rich enough in term of members and uncertainties. Therefore, the quality of the correction is only affected by the value of σ^o, the standard deviation of the error on the observations. Fig. (3.17a-c) presents the comparison between the true and the forecasted/analyzed fire front positions. It can be noted that when several observations are used, the analysis still features a circular shape but located at an optimal distance of the observations with respect to the observations errors and the model errors (not shown here).

The results in Fig. (3.17b) were produced with a low value of the observation error standard deviation, $\sigma^o = 1$ m compared to the circumference of the fronts while those in Fig. (3.17c) are produced with a higher value, $\sigma^o = 10$ m. Fig. (3.17d) shows then the influence of this error on the EnKF performance using the RMS formulation presented in section 1.4 as a function ofσ^o. The vertical bars in Fig. (3.17d) give a graphical representation of the magnitude of the standard deviations in the forecast and analysis ensembles. The figure shows that when the observation error is small, the EnKF algorithm successfully drives the analysis ensemble towards the true state; in contrast, when the observation error is large, the EnKF algorithm has reduced effects and the analysis ensemble remains close to the forecast ensemble; for intermediate values of the observation error, the EnKF algorithm produces optimized predictions lying between forecast and observation. These different regimes illustrate how DA combines information from

both models and observations, and produces better results that those that would be obtained if models or observations were considered separately.

In the following tests, the observation error is assumed to be small and the performance of EnKF will be evaluated by its ability to track the observed fire front location.

3.3.2 Sensitivity to the resolution of the observed fire fronts

Sensitivity to the number of observations N_{fr^o}

The purpose of this simulation is to study the contribution of the number of markers N_{fr}^o uniformly-distibuted along the observed front by taking N_{fr}^o from 1 to 100 while N_{fr} remains at 100. Generally, the parameters properties are not uniformly-distributed leading to anisotropic rate of spread and thus anisotropic propagation. In the following experiment, named RAND2_Nfr_o, the fuel distribution is anisotropic. It is randomly distributed, meaning that the fuel layer is composed of random pockets where the fuel thickness varies from 0 to 1 m, see Fig. (3.18a). This implies that the true front and so the simulated/observed front are not circular anymore (see Fig. (3.18b)).

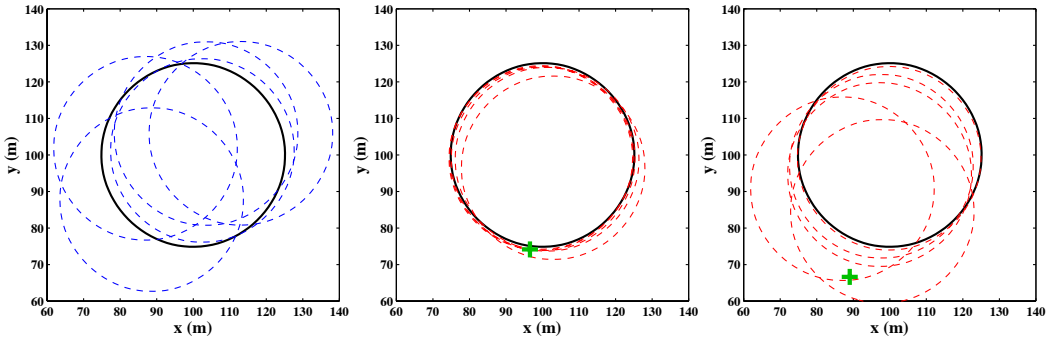
When there is only one observation with a low uncertaincy, the simulated are corrected only in the vicinity of the marker. As illustrated in Fig. (3.18c), the analysis fire fronts all gather close to the observation point and remain separated away from it. Then in Fig. (3.18d), the analysis fire fronts are more accurate and almost merged with the true front when the observed front is described with more markers. This is shown in Figure (3.18e) with the RMS distance as a function of N_{fr}^o. The figure shows similar results as in Figure (3.17d). Moreover, when the number of observations keeps increasing, the added markers are really close to each other and some information gets repetitive. We can see in Figure (3.18e) that the analysis RMS is stable from $N_{fr}^o = 20$ to $N_{fr}^o = 100$, implying that a too large number of observations compared to the resolution of the simulated fronts is unnecessary.

In other words, the performance of the DA algorithm and its ability to capture the high-resolution features of the fire front depend strongly on the density of the observation network; where this density depends itself on the spatial variability of the fire front. Smooth observed fronts does not need as many markers as chaotic and irregular ones.

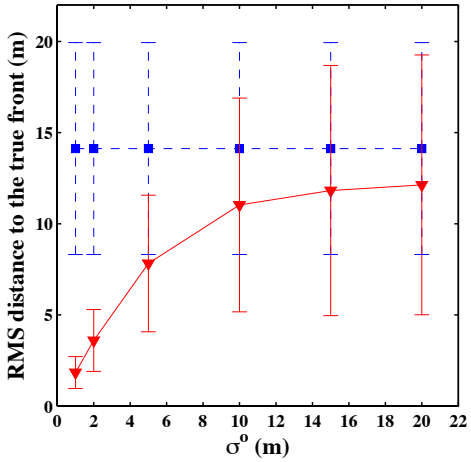
Performances according to the repartition of the observed marker along the front

Directly following the results in previous section, we compare the performances of the EnKF when first, the observation markers are uniformly-distributed, and second when the observed front is incomplete.

The anisotropy of the propagation is now represented by a wide range of uncertainties in the ensemble (see the forecast members in Fig. (3.19a)). Figures (3.19b-c) illustrates the forecast correlation functions for the 70th marker in the fire fronts (localized as a pink squared marker

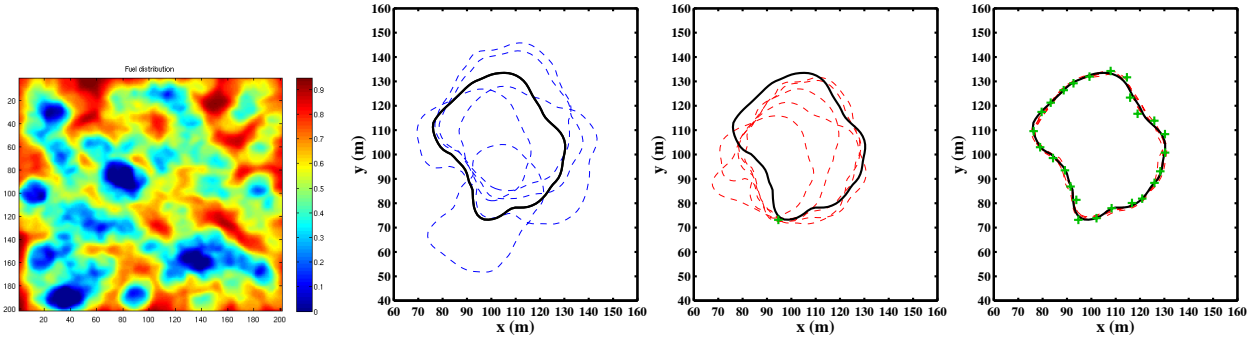


(a) Forecast (b) Analysis with $\sigma^o = 1$ m (c) Analysis with $\sigma^o = 10$ m

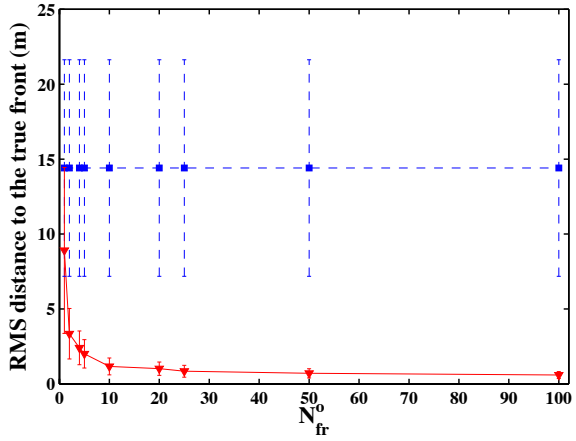


(d) EnKF performance versus σ^o

Figure 3.17: OSSE test with constant ROS and one observation. (a) Comparison between true (black solid line) and forecasted (dashed blue lines) fire front positions; the green cross symbol is the observation. (b) Similar comparison between true (solid black line) and analyzed (dashed red lines) fire front positions. (c) Averaged distance between the true and forecasted fire front positions (blue squares) and between the true and analyzed fire front positions (red squares) as a function of the observation error standard deviation σ^o.



(a) Random fuel distribution (b) Forecast (c) Analysis with $N^o_{fr} = 1$ (d) Analysis with $N^o_{fr} = 25$



(e) EnKF performance versus N^o_{fr}

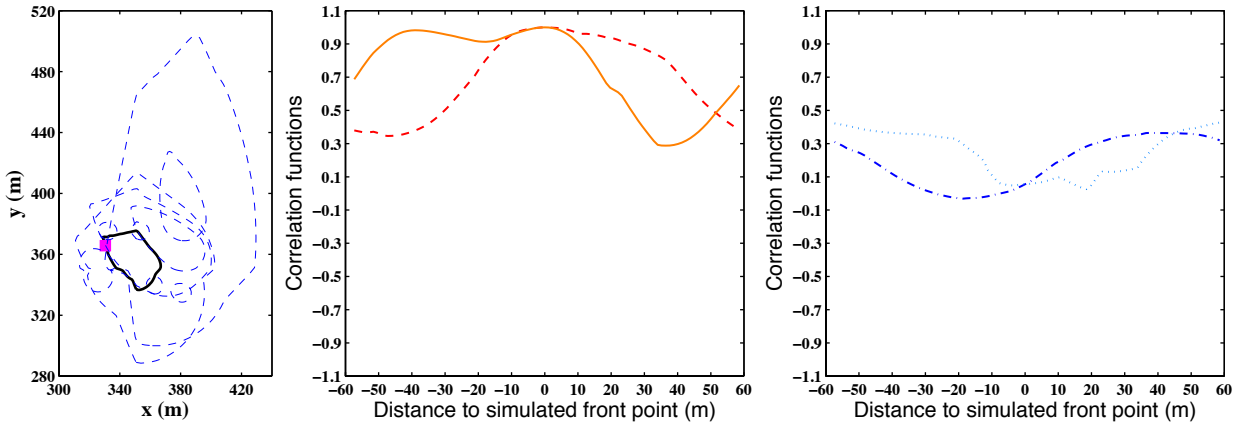
Figure 3.18: OSSE test with anisotropic ROS. (a) An example of random fuel disribution. (b) Comparison between true (black solid line) and forecasted (dashed blue lines) fire front positions. (c) Comparison between true (solid black line) and analyzed (dashed red lines) fire front positions obtained with 1 observation (green cross symbol). (d) Similar comparison between true (solid black line) and analyzed (dashed red lines) fire front positions obtained with 25 observation (green cross symbols).(e) Averaged distance between the true and forecasted fire front positions (blue squares) and between the true and analyzed fire front positions (red triangles) as a function of the number of observation markers N^o_{fr}.

in the true front in Fig. (3.19a)). Those correlation functions presents a curve shape similar with the forecast correlation functions obtained in Fig. (3.12) but their variations are more pronounced. While studying correlations, we can measure the area of influence of a particular marker by determining its correlation length scale at each of its side. In Fig. (3.19b), the correlation length-scale correspond to the distance to the studied point (abscissa equals to 0) where the correlation function is equal and higher than 0.5. In Fig. (3.12), this characteristic length was "infinite" while here it is finite; e.g in Fig. (3.19b), the left correlation length scale for the red curve is approximately equal to 20 m and the right one to 45 m. Therefore, the error in the location of one observed point is only correlated with the error in the location of the other adjacent points along the fire front.

As expected when the observed front is fully described, see Figure (3.20a), the analyzed front positions feature a much reduced scatter in terms of fire front shapes and are located close to the true front position. The DA algorithm translates the information coming from one observation point into a correction restricted to the closets neighbors only. As a consequence, when several observations are available, a non-uniform correction is obtained and the DA algorithm is then able to change the shape of the fire front and more easily match the observations.

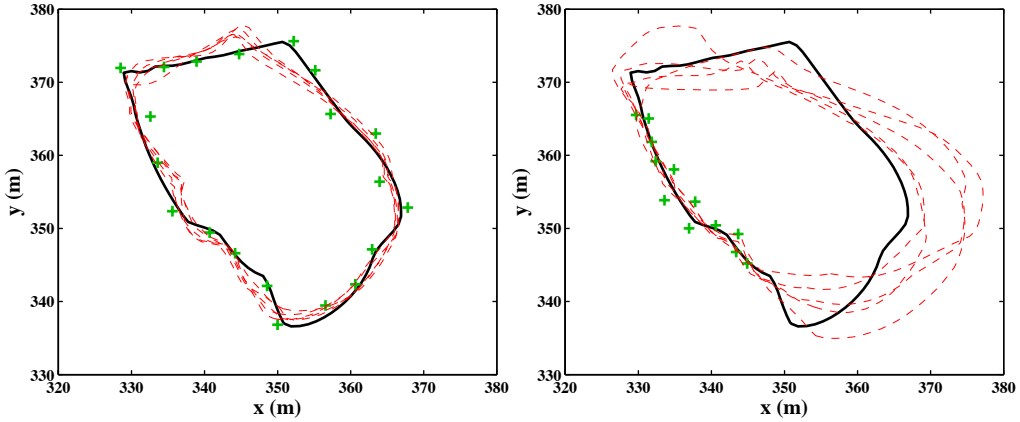
While Figure (3.20a) shows that the direct observation of the fire front position can overcome various uncertainties in the wildfire spread model parameters, Figure (3.20b) illustrates that the spatial distribution of the observations along the fire contour has a significant impact on the analysis. This figure considers a practically relevant situation in which the observations are limited to a certain section of the fire front (the informed section) and therefore provide an incomplete picture. In this situation, while the EnKF algorithm produces an analysis that is close to the true state in the informed section, the benefits of data assimilation are much reduced in the non-informed sections.

It is worth pointing out, however, that despite a reduced level of performance, EnKF informed by incomplete observations remains capable of improved performance compared to a stand-alone forecast. It is here interesting to observe the forecast correlation functions of markers which are far enough from the observations to not be affected by them; such correlations are presented in Fig. (3.21). Indeed, for those markers, the only information used to correct their position is the information contained in the correlation functions. The markers with observations facing them are those at the largest distance from the studied marker (distance greater than 55 meters from each sides). The correlations with those particular markers are all very low (around 0.3) excepted for the correlation of the y-coordinate with respect to the other y-coordinates (orange line in Fig. (3.21a)). Therefore, only the y-coordinate of the distant markers can be corrected. This is actually verified in Fig. (3.20b): the analysed fire fronts are correctly positionned according to the vertical coordinate and the horizontal dispersion of the markers opposed to the observations is way more important.



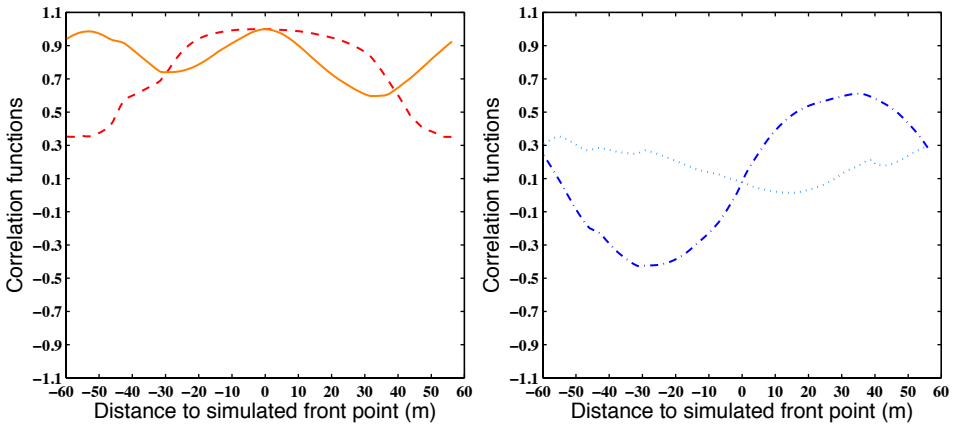
(a) Forecast fire fronts (b) Forecast correlations functions between same type of coordinates (c) Forecast crossed-correlations functions between different type of coordinates

Figure 3.19: Spatially-varying OSSE test with uncertain ROS model parameters and uncertain ignition location. (a) Comparison between true (black solid line) and forecasted (blue dashed lines) fire front positions. (b-c) Correlation functions associated with the 70th marker (pink squared-symbol in a): (b) x-coordinate w.r.t the x-coordinates (dashed-red line) and y-coordinate w.r.t the y-coordinates (plain-orange line). (c) x-coordinate w.r.t the y-coordinates (dashed-dottes-blue line) and y-coordinate w.r.t the x-coordinates (dotted-light blue line).



(a) Analysis with $N_{fr}^{o} = 20$ (b) Analysis with $N_{fr}^{o} = 12$ for an incomplete set of observations

Figure 3.20: Spatially-varying OSSE test with uncertain ROS model parameters and uncertain ignition location. (a) Comparison between true (black solid line) and analyzed (red dashed lines) fire front positions; the green cross symbols are the observations. (b) See caption of Fig.(a); case of an incomplete set of observations.



(a) Forecast correlations functions between same type of coordinates

(b) Forecast crossed-correlations functions between different type of coordinates

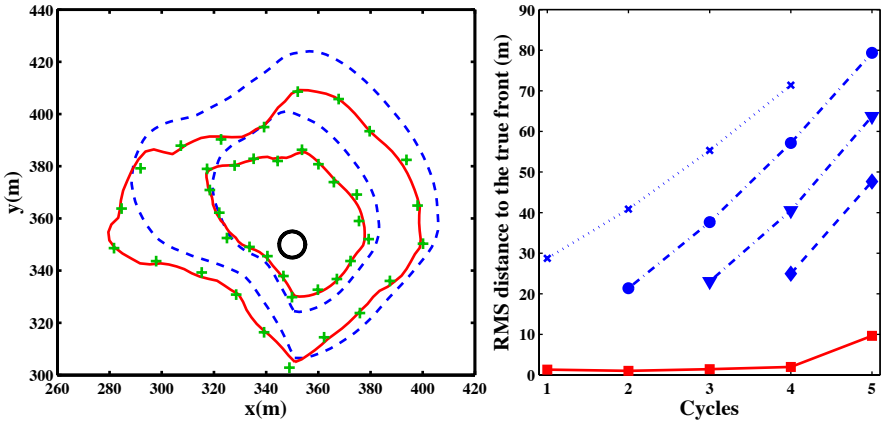
Figure 3.21: Correlation functions associated with the 25th marker whose far awy from all observation marker in observed front in Fig. (3.20b): (a) x-coordinate w.r.t the x-coordinates (dashed-red line) and y-coordinate w.r.t the y-coordinates (plain-orange line). (b) x-coordinate w.r.t the y-coordinates (dashed-dottes-blue line) and y-coordinate w.r.t the x-coordinates (dotted-light blue line).

3.3.3 Forecasting performances

We now consider multiple analysis cycles for the spatially-varying OSSE test and examine the behavior of the forecast between successive observations or after the last observation was made. The true fire front spread is simulated for time-varying wind conditions, whereas the forecast ensemble is simulated for constant wind conditions, using the parameters presented in Table (3.2).

The performance of the forecast is expected to deteriorate over time for two reasons. First, because the impact of the fire front correction at a previous time decreases as the forecast lead time increases. Second, because the present implementation of the EnKF does not provide any correction for ROS modeling errors or any correction for uncertainties in the ROS model input parameters (including the incorrect assumption of a constant wind). In this test, we apply the EnKF algorithm over five successive analysis cycles. The complete assimilation window is of 600 s but each cycle is divided into four observation times at $t_1 = 150$ s, $t_2 = 300$ s, $t_3 = 450$ s and $t_4 = 600$ s with the analysis update performed at $t_1 = 150$ s. To compare the forecast performances, the analysis produced as t_1 of the current cycle is used as the initial condition for the next assimilation cycle while the fire spread model integration is continued from the current forecast at t_1 to t_4.

Figure (3.22a) presents a comparison between the averaged forecast fire front positions, the observations (considered to be close to the true state) and the averaged analysis fire front



(a) Forecast and analysis (b) EnKF performance versus time

Figure 3.22: Spatially-varying OSSE test with uncertain ROS model parameters and uncertain ignition location; multiple analysis cycles. (a) Comparison between mean forecast (blue dashed lines), observations (green crosses) and mean analysis (red solid lines) at times $t_2 = 300$ s and $t_4 = 600$ s; the small circle in the center of the figure corresponds to the initial conditions. (b) Averaged distance between the true and forecasted/analyzed fire front positions as a function of the analysis cycle number; crosses correspond to a forecast with no update; circles, triangles, diamonds correspond to a forecast with an update at $t_1 = 150$ s, $t_2 = 300$ s, $t_3 = 450$ s, respectively; square symbols correspond to an analysis performed at times t_1, t_2, t_3 and t_4.

positions at time $t_2 = 300$ s and $t_4 = 600$ s. Similar to the results obtained in previous OSSE tests, the RMS distance to the true front is significantly reduced by EnKF, from 20 m for the forecast to less than 1 m for the analysis (see also the curve with square symbols in Fig. (3.22b)). Figure (3.22b) shows the typical cyclic evolution of the deviations of model predictions from the true: during the update step of the analysis cycle n, the analysis provides a correction to the front position and the distance between the true state and the forecast is significantly reduced; the analysis ensemble at the end of cycle n provides the initial conditions for the next cycle $(n + 1)$; during the prediction step of cycle $(n + 1)$, the wildfire spread model simulates the fire evolution and the distance between the true state and the forecast increases significantly; during the update step of the analysis cycle $(n+1)$, the distance between the true state and the forecast is again reduced and the cycle is repeated. For instance, in Fig. (3.22b), the curve with circles correspond to an averaged forecast that has been updated at time $t_1 = 150$ s, thereby leading to a close approximation of the true state; without additional observations, this forecast is seen to deviate from the true state (the RMS distance between the true state and the forecast is approximately 80 m at time $t_5 = 750$ s).

In summary, these results show that in a state estimation approach, the EnKF updates have to be performed at regular time intervals in order to allow for an accurate tracking of the true fire front position.

The ensemble of all OSSE simulations has shown that this data-driven prototype is capable of efficiently tracking the fire front location and local deformations. The state estimation approach is the most efficient for short-term predictions with a complete set of observations, but remain exploitable in cases with partially-observed fire front. Finally, to validate this state estimation approach for more realistic cases, the prototype will be evaluated on a set of data from a controlled grassland fire experiment verifying the assumption that the observed data are frequent enough and uniformly-distributed along the fire front.

3.4 Validation study: Application to a controlled grassland fire experiment

We now evaluate the EnKF-FIREFLY prototype in a validation study corresponding to a controlled grassland fire experiment. The experimental configuration corresponds to a small (4 m x 4 m), flat and and horizontal, open-field grassland lot burning under moderate wind conditions. The properties of the grass are (approximately) known and so are the wind conditions. The fire spread is recorded during 350 s using a thermal-infrared camera; the thermal maps are post-processed (the fire front is defined at the 600 K iso-temperature contour) and thereby provide full fire contours at 14 s intervals; based on the spatial resolution of the camera, the estimated standard deviation of the measurement error is $\sigma^o = 0.05$ m. An ensemble of $N_e = 50$ forecasts is produced based on assumed uncertainties in the ROS input model parameters as well as uncertainties in the initial conditions taken at time $t_0 = 50$ s. Moreover, the fire is ignited at the bottom boundary of the domain. In terms of input parameters for the Rothermel's model, it means that the y-coordinate of the initial condition is known.

In this configuration, the vegetation and wind properties are assumed to be uniformly distributes, eventhough that is not perfectly true. Plus, from OSSE experiments, we know that the simulated fire fronts can be properly deformed only if we settle spatial-distributed parameters. Besides, the fire starts at the bottom of the domain; so the spread will be contained in only two zones of the previous four-zones fuel distribution. To overcome this, a new version of the four-zones fuel is introduced where the domain's width is divided in four. This fuel distribution should be able to deform more the simulated fire front. The input parameter y_{ign} is not needed anymore so $\mathbf{p_m}$ reduces to a vector of 9 parameters as follows:

$$\mathbf{p_m} = \{x_{ign},\ \delta_{v,1},\ \delta_{v,2},\ \delta_{v,3},\ \delta_{v,4},\ M_v,\ \Sigma_v,\ \mathbf{u_w}\} \tag{3.15}$$

with their respective true and mean/standard deviation forecast values detailed in Table (3.3). Note that the variable x_{ign} represents here a general shift of the initial condition along the x-axis. Following the conclusions from the OSSE experiements this treatment is adopted in order to generate a rich forecast ensemble with a wide range of simulated fire front shapes and locations. Finally, in these simulations, $N_{fr} = 100$ and $N^o_{fr} = 50$.

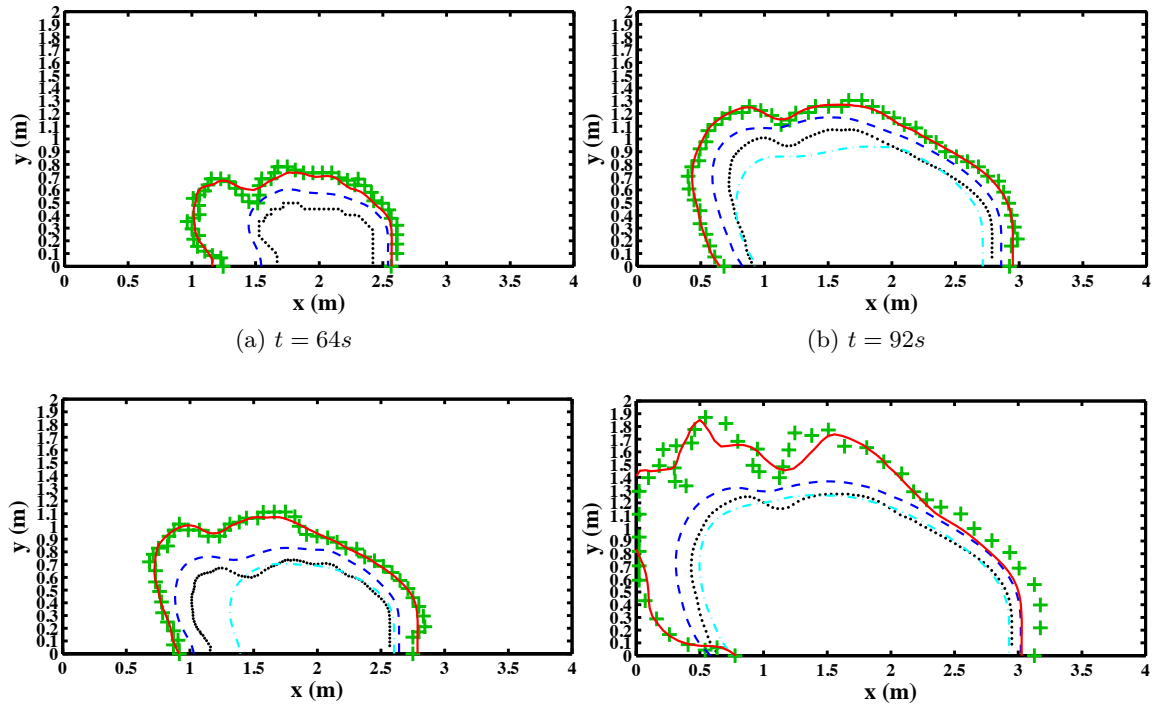
Figure (3.23) presents a comparison between the averaged forecast fire front positions, the observations and the averaged analyzed fire front positions, at time $t_1 = 64$ s, $t_2 = 78$ s, $t_3 = 92$ s

Input parameter	Mean ensemble value	Standard deviation
$\delta_{v,1}$ (m)	0.06	0.04
$\delta_{v,2}$ (m)	0.08	0.04
$\delta_{v,3}$ (m)	0.1	0.04
$\delta_{v,4}$ (m)	0.12	0.04
M_v (%)	22	6
Σ_v (1/m)	11500	4000
$\mathbf{u}_w$ (m/s, deg)	(1, 307)	(0.4, 45)
x_{ign} (m)	2	0.65

Table 3.3: Properties of the ensemble forecast in the real-data test.

and $t_4 = 106$ s. Note that in Figure (3.23b-d), the mean forecast (blue dashed line) is initialized at time t_1, t_2, t_3 by the analysis produced by the previous DA cycle (red solid line in Figure (3.23)a-c) while the free forecast (cyan dashed-dotted line) is initialized at time t_0, t_1, t_2 and does not use any analysis. In Figure (3.23), it is seen that the mean forecast significantly underestimates the rate of spread of the fire; in contrast, the mean analysis provides accurate estimates of the fire front position. The agreement between the analyzed and observed fire front positions is improved compared to what was previously obtained in refs [32, 34, 33] using a spatially-uniform parameter estimation approach for the same test case. In particular, the analyzed fire fronts feature a topology that is very close to that of the observed front, a result that requires an accurate and non-uniform correction of the locations of the fire front markers.

These results are similar to those obtained in Figure (3.22b) and suggest that while a state estimation approach provides excellent forecasting performance at short lead times, this level of performance is not persistent and needs to be renewed by frequent observations or parameter correction.



(a) $t = 64s$ (b) $t = 92s$

(c) $t = 78s$ (d) $t = 106s$

Figure 3.23: Validation test performed using data from a controlled grassland fire experiment: comparison between initial conditions at previous time ($t_0 = 50$ s for cycle 1) (black open circles), mean forecast (blue dashed line), observations (green crosses) and mean analysis (red solid line). (a) time $t_1 = 64$ s; (c) time $t_2 = 78$ s; (b) time $t_3 = 92$ s; (d) time $t_4 = 106$ s.

3.5 Conclusion

Assuming that observations of the fire front position are available at frequent times but can possibly provide an inaccurate and incomplete description of the fire front, the results indicate that data-driven simulations are capable of correcting inaccurate predictions of the fire front position and of subsequently providing an optimized forecast of the wildfire behavior. It was demonstrated that in order to allow for a spatially-varying correction of the front position, the generation of the EnKF ensemble should represent the anisotropy in fire propagation that results from spatial variations in vegetation properties and from the presence of wind. This anisotropy was implicitly introduced in EnKF by selecting spatially-dependent vegetation properties and different wind conditions between the members. Finally, because this DA implementation does not act on the modeling errors, the results also indicate that the forecasting performance of a state estimation approach is limited to near-term predictions (i.e., short lead times).

Chapter 4

Insertion of the slope contribution in the wildfire spread model

The most important current limitation of the wildfire spread model is its limitation to flat terrain topography. To be applicable in real-life wildfire case, the fire spread model has to be able to take into account any topographical description. The objective of the following part is to extend the flat-terrain FIREFLY solver to 3D-topography FIREFLY solver. First, we describe the 3D extrapolation strategy to incorporate the slope contribution in FIREFLY and validate the model on a simple slope plane. Then, the 3D-FIREFLY solver is applied to more realistic topography. Finally, a last simulation is performed to combine the 3D-FIREFLY solver and the EnKF algorithm.

4.1 3-D extrapolation of the Rothermel's model on a simple slope plane

4.1.1 The slope variables

In the Rothermel's model intial 1D formulation with wind and slope correction, the expression of the ROS is given in uphill heading direction. This particular direction lies into a 3-D plane defining the complete slope. This slope plane is characterized by two angles.

First, we define the orientation of the slope in the 3D space, currently called the aspect. In rainfall studies [71, 72] and on numerous works focused on slope and wind contribution on the ROS [73, 74], the aspect is set as the direction that the slope faces or also the downhill direction. The aspect is therefore represented by an angle, named the aspect angle and illustrated in Fig. (4.1), γ_a defined in the horizontal plane from the North direction in the clockwise sense, and takes values between 0° (slope facing the North direction) and 360° excluded.

Then, we define the stepness of the slope plane; represented by the slope angle γ_s in Fig. (4.1): from the $\gamma_a + \pi$-direction in the increasing z-coordinate, γ_s takes values between 0° (flat terrain) and 90° (vertical wall).

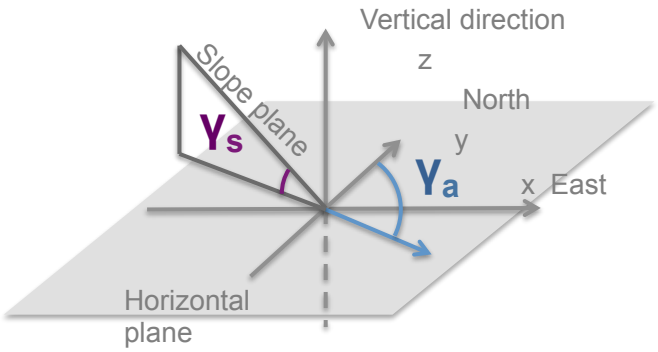


Figure 4.1: Angles defining a slope plane.

Any slope plane can be characterized by a pair of aspect and slope angles (γ_a, γ_s). A more detailed geometric description is presented in Appendix B. In a more general way, any topography can be characterized by an ensemble of such pairs defining the slope characteristics at some remarquable points. Before presenting some derived topography descriptions, we need first to extend the 1D Rothermel's formulation for a uphill-only wildfire spread, in any 2D direction in the slope plane.

4.1.2 2-D extrapolation of the Rothermel's model in the slope plane

A 2D extrapolation of the slope contribution in the Rothermel's model when the wildfire does not spread in the uphill direction (represented by the angle $\gamma_a + 180°$) was proposed by C. Lautenberger [75]. Reconsidering the flame angle θ from Chapter 1, the slope correction is based on the following assumptions:

- When the wildfire spreads in the uphill direction, i.e. $\theta = \gamma_a + 180$; the slope correction is maximum giving: $\phi_s = \phi_s^*$ [1],
- When the wildfire spreads in the normal direction of the uphill/downhill one, i.e. $\theta = \gamma_a \pm 90$; the wildfire remains at the same height while spreading and the slope contribution is equal to 0, giving $\phi_s = 0$,
- finally, for wildfire spread in downslope directions, $\theta \in [\gamma_a - 90; \gamma_a + 90]$, the slope contribution is negative in the ROS formulation. The Rothermel's model was not established for such spreading configurations. It is then assumed that the wildfire cannot spread at a ROS lower than the ROS value without any wind and slope, namely Γ_0. So, the ROS is forced at Γ_0.

Therefore, the 2D extrapolation of the Rothermel's model with slope correction only is given by

$$\Gamma = \Gamma_0 \max\left(1,\ 1 + \cos(\theta - \gamma_a - \pi)\phi_s^*\right). \tag{4.1}$$

[1] ϕ_w^* as described in Chapter 1.

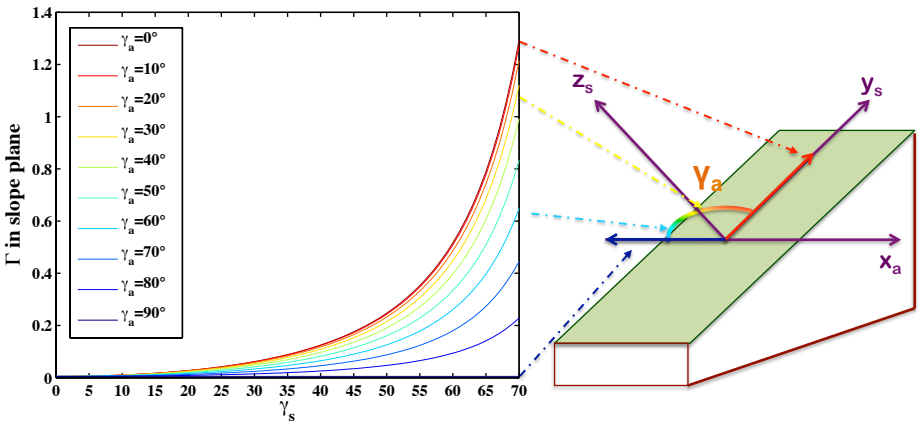


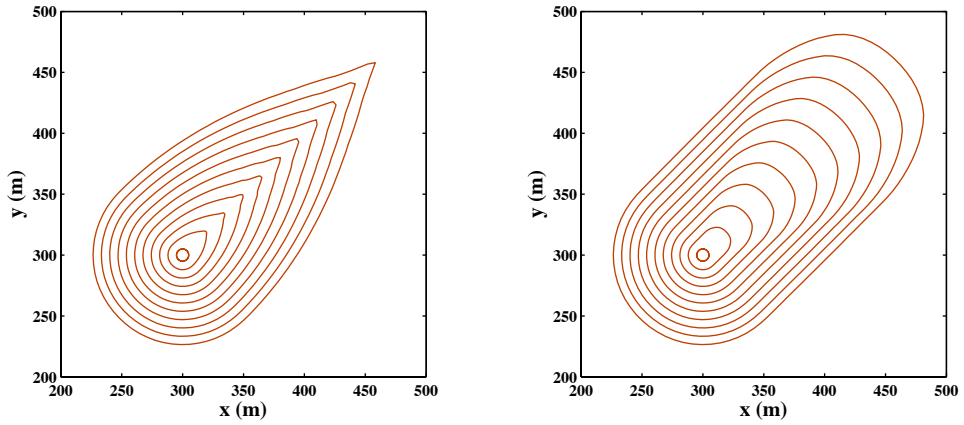
Figure 4.2: Rate of spread Γ with slope correction with respect to the slope angle (abscissa axis on the plot) and the aspect angle (colored curves) for a spread direction $\theta = 180°$. In the configuration of the experiment, $\Gamma_0 = 0.048\text{m.s}^{-1}$.

Figure (4.2) presents the value of the ROS with respect to the slope angle γ_s and for different aspects γ_a with $\theta = 180°$. The tested values of γ_a are such that the fire goes up, so the effects of the slope are positive. For $\gamma_a = 90°$ (dark blue line), the fire is propagating across the slope, so there is no change in the altitude and there is no contribution of the slope on the ROS. In the opposite, when the fire propagates uphill ($\gamma_a = 0°$, red line), the contribution of the slope is maximal.

It is interesting to note that the wind correction proposed by C.Lautenberger was similar to its slope correction, giving the flame angle θ and the wind angle θ_w. The ROS expression with wind and slope correction was therefore

$$\Gamma = \Gamma_0 \max\left(1,\ 1 + \cos(\theta - \theta_w - \pi)\phi_w^* + \cos(\theta - \gamma_a - \pi)\phi_s^*\right). \tag{4.2}$$

This wind correction differs from the current wind correction implemented in FIREFLY. First of all, the definition of the wind angle is different: for FIREFLY, θ_w represents the direction in which the wind blows while for C. Lautenberger, θ_w is the direction from which the wind blows. In the literature, this last definition of the wind angle is the most widely used. Also, in the current FIREFLY version, the ROS is projected on the wind vector and the resulting wind velocity is used to determine the value of the wind factor ϕ_w. In C. Lautenberger's formulation, it is the other way around: the wind factor is calculated with the total wind velocity and the wind contribution is projected on the flame vector. Figure (4.3) compares the two wind formulations for a wind blowing from the South-West direction at a velocity 0.75 m.s^{-1}. For now on, we will use Lautenberger's definition of the wind angle and its wind correction in FIREFLY in order to associate it with the slope correction presented in Eq. (4.1).



(a) FIREFLY (b) CL

Figure 4.3: Fire fronts profile every 100 s on a flat terrain with a wind blowing from the South-West ($\theta_w = 225°$ with the C.L definition): (a) FIREFLY wind correction, (b) CL wind correction.

The slope correction in Eq. (4.1) gives the expression of the ROS in the slope plane. The consideration of the third coordinate within the FIREFLY implementation can turn out to be quite complicated and expensive. For the EnKF implementation, this would mean increasing the size of all the variables (vectors and matrices) to take into account the z-coordinate which besides, cannot be considered the same way as the x- and y- coordinates. Indeed, for a given pair (x, y), the associated z-coordinate has to be equal to the corresponding height in the topographical distribution. Therefore, we adopt here the same strategy as C. Lautenberger where the 3D surface and so the flame vector are projected onto the horizontal plane. Thus, the definition of the DA variables remains limited to the 2D x- and y-coordinates as they were defined when the slope was not taken into account. Using space geometry properties, we define the projected ROS at a given mesh node with aspect γ_a and slope γ_s by

$$\tilde{\Gamma} = \frac{\Gamma}{\sqrt{1 + \tan^2(\gamma_s)\cos^2(\gamma_a - \theta)}}. \tag{4.3}$$

The details of the determination of the projection coefficient is presented in Appendix C.

This strategy is interesting because it does not imply important changes in the current implementation of the FIREFLY solver. Moreover, it can be adapted for all types of topographical configuration as long as we know the value of the angles γ_a and γ_s at each mesh node. Before applying the algorithm for complex topography, we have to validate it on the simple case of a slope plane.

Fuel depth	δ_v	1 (m)
Fuel moisture content	M_v	15%
Surface-to-volume ratio	Σ_v	11500 (1/m)

Table 4.1: Fuel characteristics

4.1.3 Validation of the slope correction in a simple slope case

Validation of the headfire ROS

The simulations are run on a fuel with uniform characteristics and no wind, see Table (4.1). The terrain is a simple slope with slope angle $\gamma_s = 15°$ and different aspects γ_a: respectively $\gamma_{a,1} = 180°$, $\gamma_{a,2} = 270°$ and $\gamma_{a,3} = 225°$. The initial fire is a circle of radius 5 m in the center of the square computational domain of size 600 m $\times$ 600 m. The fire fronts are saved every 100 seconds over a spread of 1000 seconds. The results are presented in Fig. (4.4). The results show that the fire front is faster in the uphill direction that in the downhill direction, for any aspect. The topography implementation is then validated by comparing the ROS of the simulated head of the fire and the theoretical one.

In the conditions of the simulation, we have the following constants:

$$\Gamma_0 = 6.82 \times 10^{-2} \text{m.s}^{-1}, \quad \phi_s^* = 2.9537. \tag{4.4}$$

Given Eq. (4.1) and (4.3), the projected ROS at the head of the fire (i.e. $\theta = \gamma_a + \pi$) is equal to:

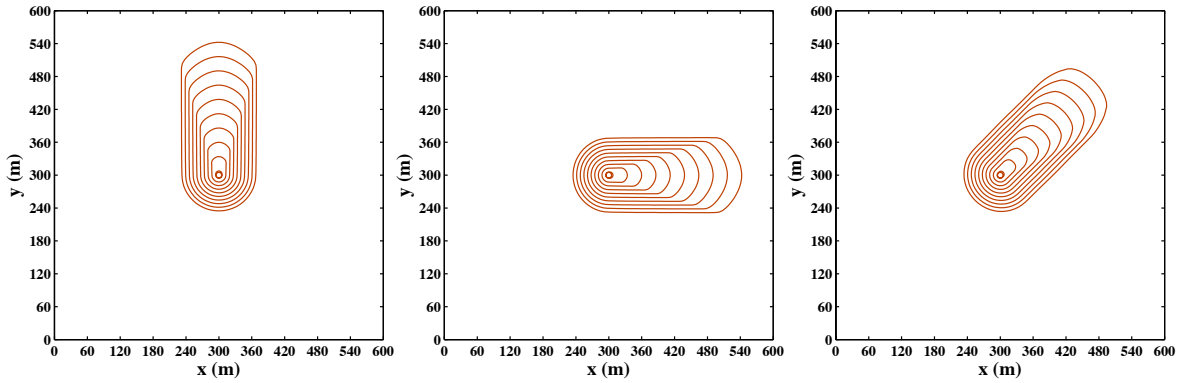
$$\tilde{\Gamma}(\gamma_a + \pi) = \tilde{\Gamma}_{\text{th}} = \frac{\Gamma_0(1 + \phi_s^*)}{\sqrt{1 + \tan^2(\gamma_s)}} = 0.2608 \text{ m.s}^{-1}. \tag{4.5}$$

It is important to recall that, because all the fuel characteristics are uniformly-distributed, Γ_0 is constant in time and in any direction θ. The value of Γ depends only on the value of the aspect angle γ_a.

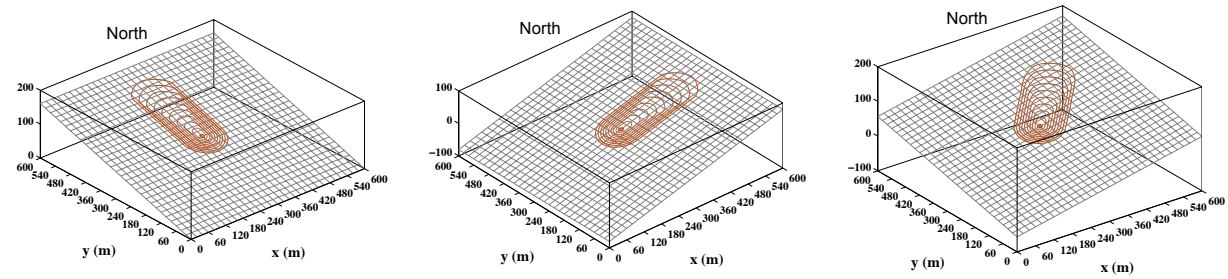
To determine the rate of spread in a direction θ, we have to determine the ratio of the covered distance with the time during which the distance was actually covered. From the simulation, the time between two fire fronts observations is of 100 seconds. The covered distance is determined by taking in each front the marker with the same index, see Fig. (4.5).

Figure (4.6) compares the covered distance determined with the theoretical rate of spread by plotting $d_{\text{th}} = \tilde{\Gamma}(\gamma_a + \pi)t$ (square-symboled red line) and the plot of the simulated covered distance against observation times (dotted blue line) when $\gamma_a = 180°$ and $\gamma_a = 225°$. The simulated rate of spread at the head of the fire is determined here by taking the mean over the observation times of the ratios of the covered distance over the corresponding observation time:

$$\tilde{\Gamma}_{\text{exp},\gamma_{a,1}} = 0.2622 \text{ m.s}^{-1} \quad \text{and} \quad \tilde{\Gamma}_{\text{exp},\gamma_{a,3}} = 0.2637 \text{ m.s}^{-1} \tag{4.6}$$

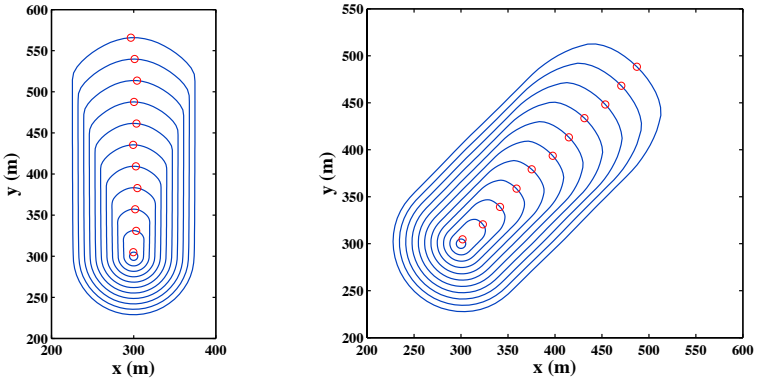


(a) $\gamma_a = 180°$ (b) $\gamma_a = 270°$ (c) $\gamma_a = 225°$



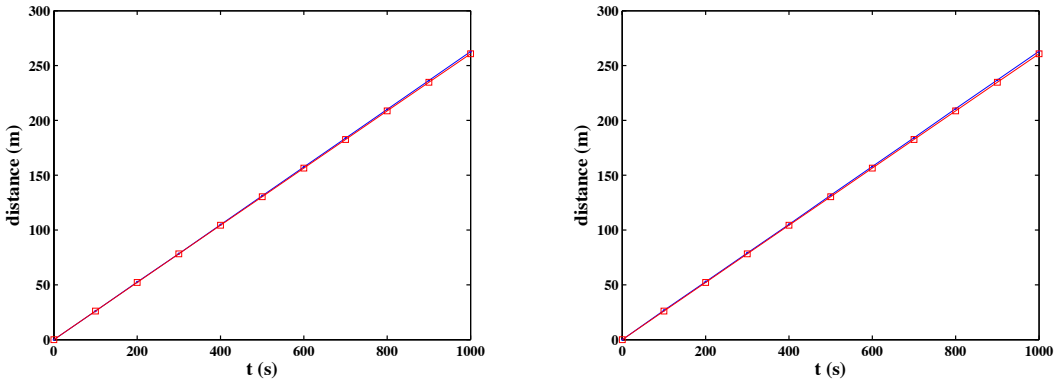
(d) $\gamma_a = 180°$ (e) $\gamma_a = 270°$ (f) $\gamma_a = 225°$

Figure 4.4: Results for wildfire spread on a plane slope with slope angle $\gamma_s = 15°$ and aspect angle $\gamma_{a,1} = 180°$ (a,d); $\gamma_{a,2} = 270°$ (b,e); $\gamma_{a,3} = 225°$ (c,f)



(a) $\gamma_{a,1} = 180°$ (b) $\gamma_{a,3} = 225°$

Figure 4.5: Selected markers over the simulated fronts to determine the covered distance in the direction $\theta = \gamma_a + \pi$.



(a) $\gamma_{a,1} = 180°$ (b) $\gamma_{a,3} = 225°$

Figure 4.6: Theoretical covered distance (m) (square-symboled red line) compared to the simulated covered distance (m) (dotted blue line) with respect to the time in seconds.

Domain		600 m×600 m, $dx = dy = 1$ m
Time		1000 s, $dt = 0.5$ s
Ignition config		circular
Fuel depth	δ_v	1 (m)
Fuel moisture content	M_v	15 s%
Surface-to-volume ratio	Σ_v	11500 (m^{-1})
Wind velocity	u_w	0 ($m.s^{-1}$)
Slope	γ_s	0°

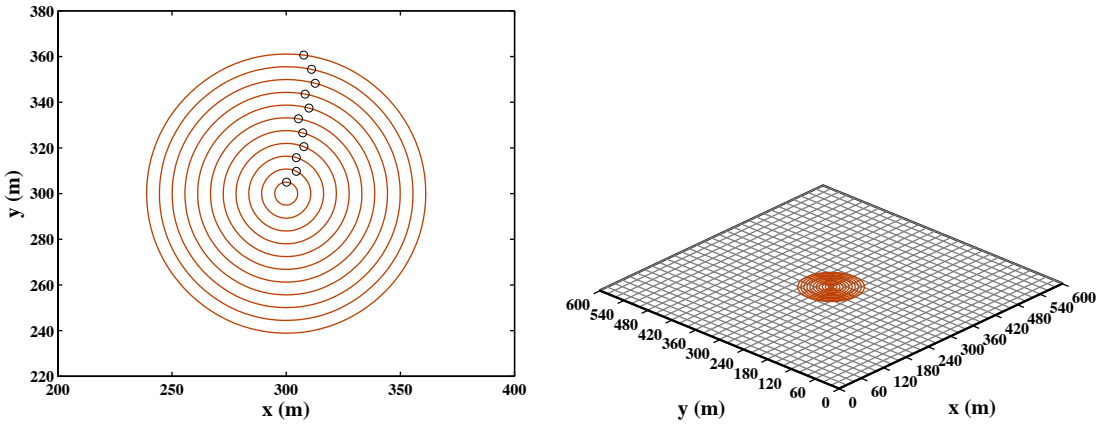
Table 4.2: Configuration for the simulation to validate an isotropic spread on a flat terrain.

The rates of spread agree with the theoretical rate of spread. However, the simulated rates of spread slightly over-estimate the theoretical one because the selected markers to determine the simulated covered distance are slightly shifted from one another and not perfectly aligned. The simulated covered distance is then slightly greater than the real one, so is the rate of spread.

Validation on a flat terrain

FIREFLY was originally designed for flat terrains. If we consider the simulation configuration presented in Table (4.2), the uniform-distribution of all the parameters should enssure an isotropic spread at the ROS $\Gamma_0 = 0.0559$ $m.s^{-1}$, corresponding to the ROS with no slope and no wind contribution given the fuel characteristics. The resulting fronts are illustrated in Figure (4.7).

Visually, we verify that the spread is isotropic given the circular shape of the fronts. The black circular markers in Fig. (4.7a) are the markers, with the same index over the fronts, used to determine the ROS. The comparison of theoretical and experimental covered distances with respect to time is presented in Figure (4.8). The mean experimental velocity is $\Gamma = 0.0623$ $m.s^{-1}$.



(a) 2D view

(b) 3D view

Figure 4.7: Resulting fronts for a propagation on a plane with a zero-slope

This velocity is slightly above the theoretical one but this can be explained by the sequence of markers used to determine the velocity. We can see in Fig. (4.7a) that the markers are clearly not aligned then the covered distance is longer, giving a higher ROS.

We can then conclude that the results with a zero-slope terrain are equivalent to original FIREFLY results for a flat terrain and identical fuel characteristics. This conclusion allows us to validate the slope-correction algorithm for a simple slope. In the next section, we apply the slope-correction FIREFLY solver on more complex topographical distributions. An intermediate topographical distribution between the simple slope plane and the random topographical description is the canyon topography. Those simulations are presented in Appendix D.

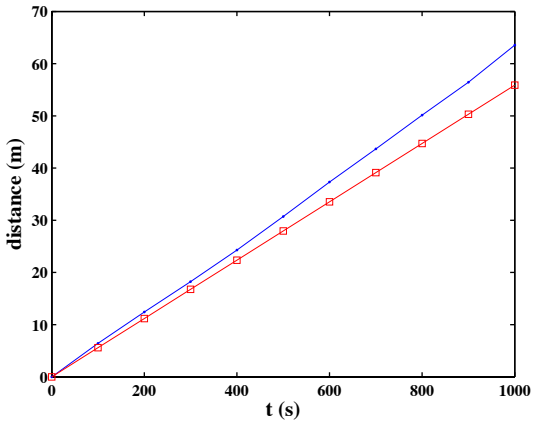


Figure 4.8: Comparison between theoretical and experimental velocities. red-squared line is for the theoretical ROS and blue-dotted line is for the experimental ROS.

4.2 Wildfire spread on a complex topography

4.2.1 Implementation strategy

The source of information to describe any topography is given as altimetric data $h(x, y)$, i.e. for each point (x, y) the corresponding topographic elevation h. As proposed in [76], it is possible to recover the slope from the field h by taking

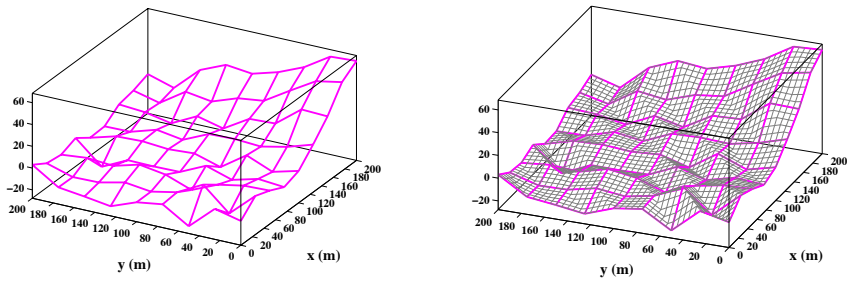
$$\tan(\gamma_s) = \sqrt{\left(\frac{\partial h(x,y)}{\partial x}\right)^2 + \left(\frac{\partial h(x,y)}{\partial y}\right)^2}. \tag{4.7}$$

Also, the aspect is defined as direction indicating the negative gradient direction such that

$$\frac{-1}{\sqrt{\left(\frac{\partial h}{\partial x}\right)^2 + \left(\frac{\partial h}{\partial y}\right)^2}} \begin{pmatrix} \frac{\partial h(x,y)}{\partial x} \\ \frac{\partial h(x,y)}{\partial y} \end{pmatrix} = \begin{pmatrix} \sin(\gamma_a) \\ \cos(\gamma_a) \end{pmatrix}. \tag{4.8}$$

To implement the gradient of h in FIREFLY, we use a basic centrale finite-difference scheme to compute slope and aspect angles at each mesh node following Eq. (4.7) and (4.8).

Moreover, altimetric data may be defined on a coarser mesh grid than the FIREFLY mesh grid. Therefore, we use a linear interpolation method with 3 points to interpolate the altimetric data mesh grid on the FIREFLY mesh grid. Figure (4.9) presents the result of the interpolation algorithm described above. The two grids are defined on a computational domain of 200m×200m. The resolution of the altimetric data grid is $dx_a = dy_a = 25$ m while the resolution of the finer grid is $dx = dy = 5$ m. The altimetric data are synthetically obtained by randomly perturbing the altimetric data of a reference canyon with the reference angles defined as $\theta_c = 30°$, $\gamma_c = 5°$, $\gamma_{s,1} = 35°$ and $\gamma_{s,2} = 5°$.



(a) Reference data (b) Interpolated data

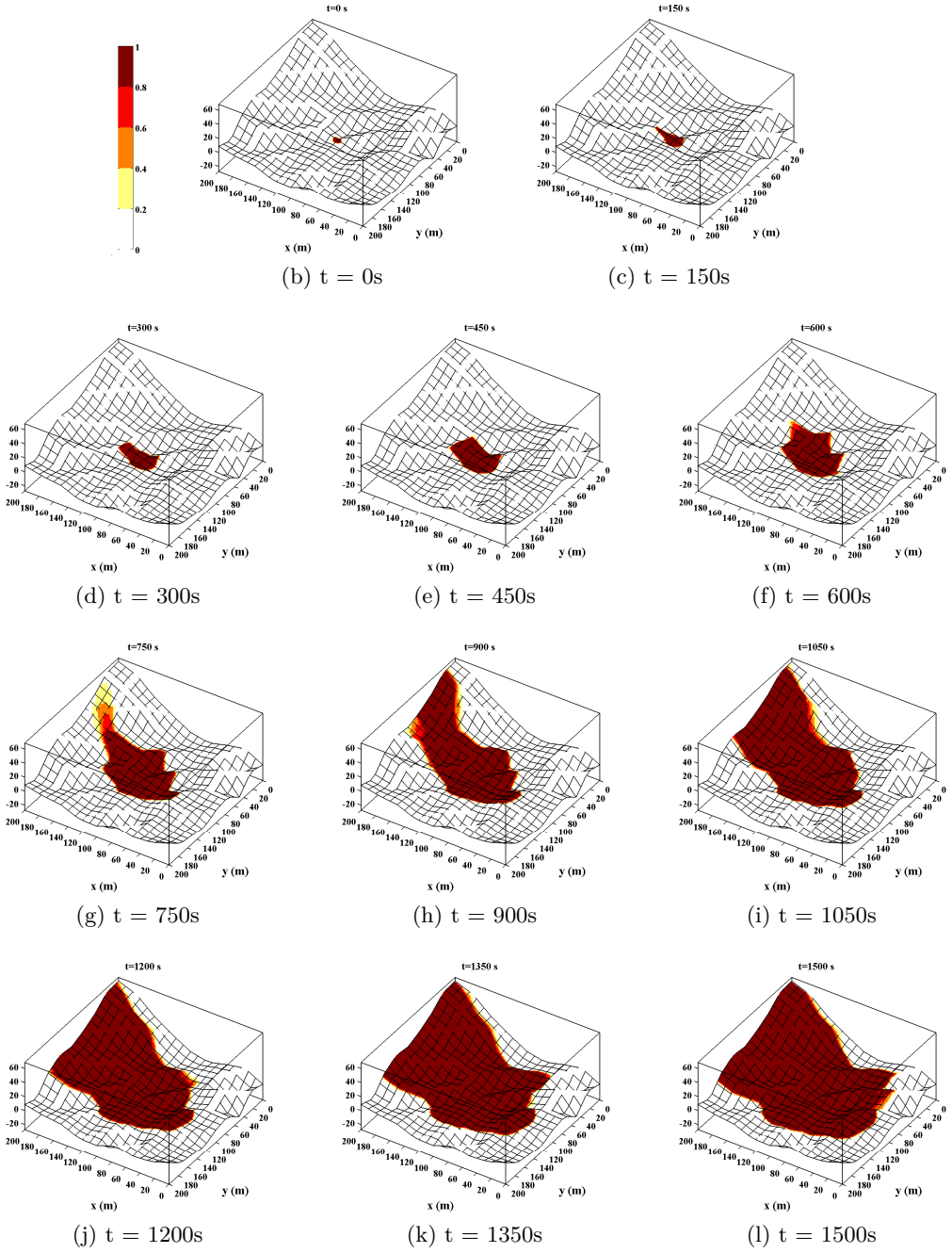
Figure 4.9: Interpolation of altimetric data (a) on a finer grid (b).

4.2.2 Numerical wildfire spread over a complex topography

The topography described in Fig. (4.9) is introduced in the wildfire spread numerical model. The fire is initialized with a circular shape of radius 5 m and positionned at the center of the domain (ie. $(x_{ign},\ y_{ign}) = (100\text{ m},\ 100\text{ m})$). The wildfire spread over an uniform fuel distribution of height $\delta_v = 1$ m, moisture content $M_v = 20$ % and surface-to-volume ratio $\Sigma_v = 10000\ \text{m}^{-1}$. A moderate wind of velocity $u_w = 0.75\ \text{m.s}^{-1}$ is blowing in the opposite direction of the important slope, ie. $\theta_w = 135°$ (recall that the wind angle indicates the direction from which the wind blows). The model is integrated over 1500 seconds with a time step of 0.5 s and the fire front, described with $N_{fr} = 2000$ markers is oberved every 75 seconds. Figure (4.10) illustrates the growth of the burnt surface (dark red surface) over the computational domain along time while Figure (4.11) similarly illustrates the spread of the corresponding fire front along time from anther viewpoint.

Even if we do not have experimental data to enssure the validity of this numerical wildfire spread, the observed results are coherent with the physics of wildfire spread. The wildfire spread is the fastest when it reaches the steepest slope and spreads in the direct upslope direction. In the other direction, the topography is almost plane so the spread is more moderate. In this simulation, the wind is to weak to clearly affect the fire propagation. In a similar simulation when a wind velocity is higher, i.e. $u_w = 1.5\ \text{m.s}^{-1}$, we can observe a fast spread in the opposite direction to the slope at the beginning of the simulation and later, when the fire front reaches the steepest slope, the spread accelerates in the slope direction as well.

Those results validate this prototype of wildfire spread model taking into account any topography. The final step of the internship is to combine the 3D spread with the correction of the EnKF.



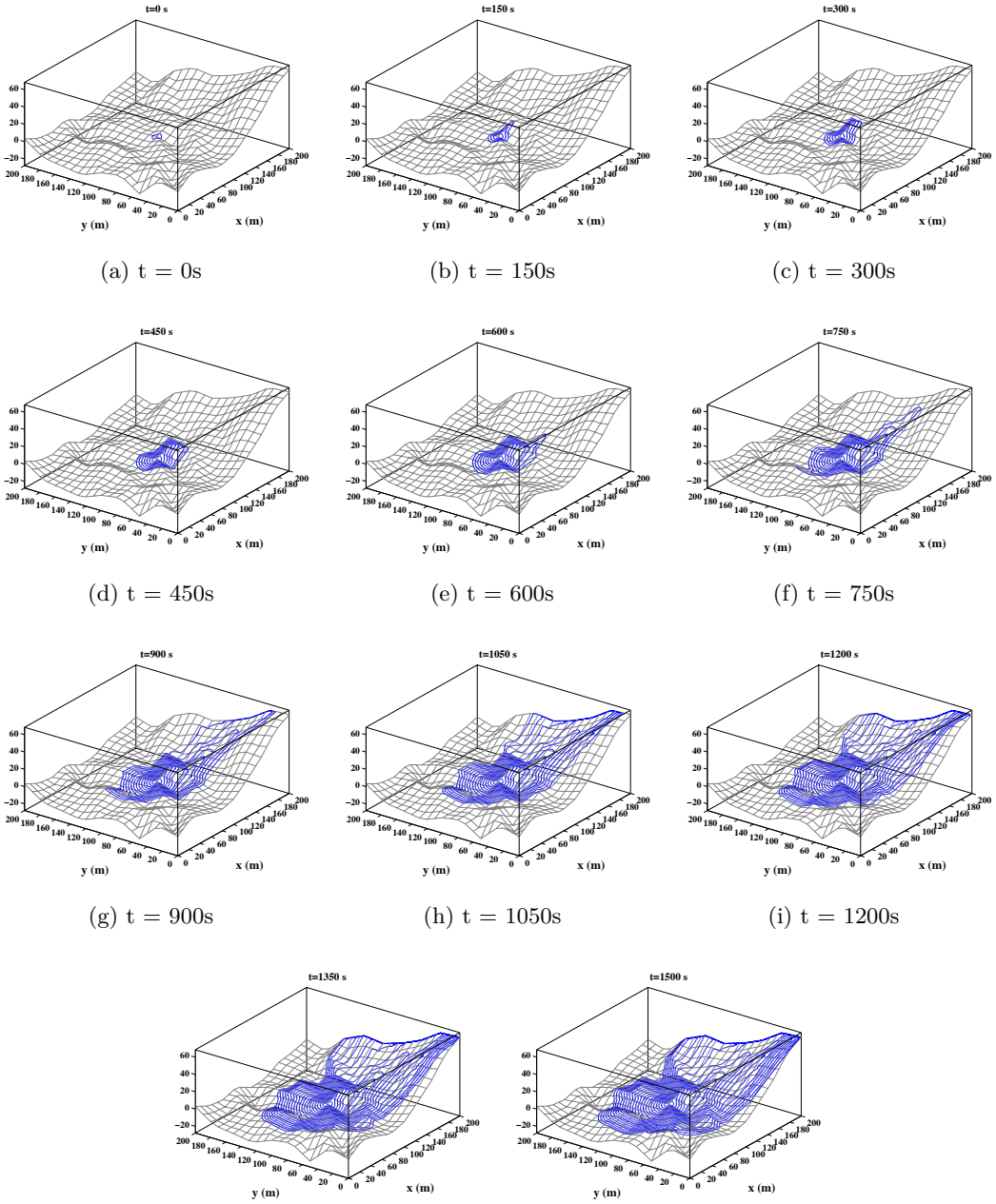
(b) t = 0s (c) t = 150s

(d) t = 300s (e) t = 450s (f) t = 600s

(g) t = 750s (h) t = 900s (i) t = 1050s

(j) t = 1200s (k) t = 1350s (l) t = 1500s

Figure 4.10: Numerical wildfire spread over a complex topography. Growth of the burnt area over time.



(a) t = 0s (b) t = 150s (c) t = 300s

(d) t = 450s (e) t = 600s (f) t = 750s

(g) t = 900s (h) t = 1050s (i) t = 1200s

(j) t = 1350s (k) t = 1500s

Figure 4.11: Numerical wildfire spread over a complex topography. Spread of the fire front over time.

4.3 Final simulation combining wildfire spread over a complex topography and correction using the EnKF

This final simulation aims at combining the 3D Firefly simulator developed within the framework of this current chapter and the EnKF implementation. As a first simulation with synthetically-generated observations combining the two algorithms, the topography description is the same for the generation of the true and observed fire fronts as for the simulated (forecast/analysis) fire fronts, namely a simple canyon whose angles are given in Table (4.3). The DA is performed over one cycle of 2000 s: this final time $t = 2000$ s corresponds to the free-run time and the mid-time $t = 1000$ s is the assimilation time when the EnKF is applied. The general characteristics of the fuel distribution and wind conditions are given in Table (4.4) and the values of the variables associated to the EnKF are given in Table (4.5). The resulting fire fronts are presented in Fig. (4.12).

Angles name	Value (deg)
θ_c	30
γ_c	5
$\gamma_{s,1}$	22
$\gamma_{s,2}$	5

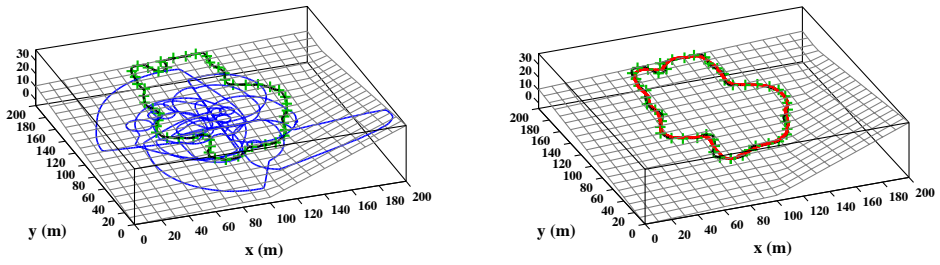
Table 4.3: Simulation with combined topography and EnKF: Angles defining the canyon topography.

Input parameter	Mean ensemble value	Standard deviation	True value
$\delta_{v,1}$ (m)	0.2	0.25	Random Fuel
$\delta_{v,2}$ (m)	0.25	0.25	Random Fuel
$\delta_{v,3}$ (m)	0.15	0.25	Random Fuel
$\delta_{v,4}$ (m)	0.1	0.25	Random Fuel
M_v (%)	15	5	10
Σ_v (1/m)	10000	4000	11500
$\mathbf{u}_w$ (m/s, deg)	(0.5, 135)	(0.2, 90)	(0.6,180)
x_{ign} (m)	100	20	100
y_{ign} (m)	100	20	100

Table 4.4: Simulation with combined topography and EnKF: Properties of the true and ensemble forecast local conditions.

EnKF variables	Value
N_e	200
N_{fr}	300
N_{fr}^o	50
σ^o (m)	1

Table 4.5: Simulation with combined topography and EnKF: Values of the EnKF variables.



(a) Background (b) Background

Figure 4.12: Resulting simulated fire fronts for a simulation combining the 3D-FIREFLY solver and the EnKF DA algorithm. The plain black line is the true front, the green cross-symbols are the observed data, (a) the blue lines are for the background fire front (b) the red lines are for the analysis fire front.

4.4 Conclusion

This chapter presents a prototype data-driven wildfire simulator capable of forecasting fire spread dynamics induced by complex terrain topography for synthetic data assimilation experiments.

The strategy of projecting the 3-D terrain onto the 2-D horizontal plane is interesting here because the general integration of the model remains the same. Therefore, the execution time for the determination of the simulated fire fronts remains adapted for operational applications. The update model is also easily combinable with EnKF instances : still only the two first coordinates of the simulated fire fronts markers are corrected within the DA algorithm and the third coordinate results from the topographical distribution. Here, a combined 3D-FIREFLY and EnKF OSSE-simulation is run where there is an identical topography for the true/observed fronts and the simulated ones. However, the terrain topography is not considered as a source of uncertainties and therefore, the same topography description is used to generate the observations and the simulation ensembles. The next step will be to modify the algorithm to perturb the slope information similarly to ROS parameters contained in $\mathbf{p_m}$; to include the uncertainty on the slope information in the correction process.

Conclusion and perspectives

During this internship, we developed a prototype data-driven wildfire simulator capable of forecasting the fire spread dynamics. The prototype simulator is based on previous studies from past years internships and a PhD thesis; and is updated by presenting, in a first step, a state estimation approach based on data assimilation and, in a second step, a strategy to account for complex terrain topography in FIREFLY.

For the validation of the state estimation EnKF implementation, the studies assume that observations of the fire front position are available at frequent times but possibly provide an inaccurate and incomplete description of the fire front. The evaluation of the prototype simulator on synthetically-generated observations simulations and on real-data experiments showed that this implementation is able to recover the shape and position of the fire front. However, this approach remains limited to short-term predictions as the impact of the initial position of the fire front becomes negligeable before the properties of the vegetation and the wind.

Future works aim at developing a dual state estimation/parameter estimation approach that would overcome the limitations illustrated in the present study and past studies on parameter estimation only. The parameter estimation approach could be extended to the case of coarse-resolution spatial variations of the ROS model parameters. Assuming that the errors on the parameters vary slowly in time, the correction provided by data assimilation can reasonably be used for forecast, thus allowing for mid- to long-term forecast. In addition, the state estimation approach could be used for short-term forecast in order to locally correct the shape of the fire front.

Besides, the fire spread model was successfully extended to 3-D topographical description and appeared to be easily combined with the EnKF algorithm. Future improvements in the wildfire model include the application of the DA approach to a fire spread model coupled with an atmospheric model (as in [77] with the Meso-NH/ForeFire coupling).

The ultimate goal of this research is to provide real-time fire forecasts using thermal-infrared imaging data including a description of both wildfire dynamics and fire plume emissions.

This internship gave me the opportunity to strenghten my competences in programming, modeling and data assimilation while also offer me basic knowledge in fire modeling and geo-

sciences. Besides, I could discover and experiment different research environment by working in two very different labaratories. This was a very enriching personal and professional experience. I ensure my interest for research and also for data assimilation and geosciences; and I will pursue my journey with a PhD thesis on a French-American project using data assimilation.

Bibliography

[1] D.X. Viegas. Overview of forest fire propagation research. *Proceedings of the International Association of Fire Safety Science*, 10:95–108, 2011.

[2] Mélanie Rochoux. Forest fire propagation using data assimilation. Technical report, CERFACS, 2010.

[3] Blaise Delmotte. Parameter estimation using data assimilation for simulation of forest fire spread. Technical report, CERFACS, 2011.

[4] Jean-Michel Bart. Data assimilation for simulation of wildfire spread using an ensemble kalman filter technique. Technical report, CERFACS, 2012.

[5] F. Beal and H. Eickner. Thermal degradation of wood components: A review of literature. Research paper FPL 130, USDA Forest Service, 1970. Madison, Wisconsin.

[6] F. Williams. Urban and wildland fire phenomenology. *Progress in Energy Combustion Science*, 8:317–354, 1982.

[7] R. Weber. Analytical models for fire spread due to radiation. *Combustion and flame*, 78:398–408, 1989.

[8] A. Grishin, A. Gruzin, and E. Gruzina. Aerodynamics and heat exchange between the front of a forest fire and the surface layer of the atmosphere. *Journal of Applied Mechanics and Technical Physics*, 25(6):889–894, 1984.

[9] R. Linn, J. Reisner, J.J. Colman, and J. Winterkamp. Studying wildfire behavior using firetec. *International Journal of Wildland Fire*, 11:233–246, 2002.

[10] W. Mell, M.A. Jenkins, J. Gould, and P. Cheney. A physics-based approach to modeling grassland fires. *International Journal of Wildland Fire*, 16:1–22, 2007.

[11] R.O. Weber. Toward a comprehensive wildfire model. *International Journal of Wildland Fire*, 4:245–248, 1991.

[12] A.L. Sullivan. A review of wildland fire spread modelling, 1990-present. part i: Physical and quasi-physicale models. *International Journal of Wildland Fire*, 18(4):349–368, 2009.

[13] R.C. Rothermel. *A Mathematical Model for Predicting Fire Spread in Wildland Fuels.* Forest Service, U.S. Dept. of Agriculture, 1972.

[14] A.L. Sullivan. A review of wildland fire spread modelling, 1990-present. part i: Empirical and quasi-empirical models. *International Journal of Wildland Fire*, 18(4):369–386, 2009.

[15] P.L Andrews. Behave: Fire behavior prediction and modeling system, burn subsystem part 1. Technical report, Forest Service, U.S. Dept. of Agriculture, Technical Report INT-194, 1986.

[16] M.A. Finney. Farsite: Fire area simulator?model development and evaluation. Technical report, Forest Service, U.S. Dept. of Agriculture, Research Paper RMRS-RP-4, 1998.

[17] I.R. Noble, G.A.A. Bary, and A.M. Gill. Mcarthur's fire danger meters expressed as equations. *Australian Journal of Ecology*, 5:201–203, 1980.

[18] K.G. Hirsch. Canadian forest fire behavior prediction (fbp) system: User's guide. Technical report, Northern Forest Centre, Special Report No. 7, 1996.

[19] R.G. Rehm and R.J. McDermott. Fire front propagation using the level-set method. Technical report, National Institute of Standards and technology, Technical Note 1611, 2009.

[20] W.H. Fransden. Fire spread through porous fuels from the conservation of energy. *Combustion and Flame*, 16:9–16, 1971.

[21] S. Osher and J.A. Sethian. Fronts propagating with curvature-dependent speed: algorithms based on hamilton-jacobi formulations. *Journal of Computationnal Physics*, 79:12–49, 1988.

[22] A. Kernstein, W. Ashurst, and F. Williams. Field equation for interface propagation in an unsteady homogeneous flow field. *Physical Review*, 37:2728–2731, 1988.

[23] K.N. Chaudhury and K.R. Ramakrishnan. Stability and convergence of the level set method in computer vision. *Pattern Recognition Letters*, 28:884–893, 2007.

[24] V. Mallet, D.E. Keyes, and F.E. Fnedell. Modeling the propagation of a wildfire through a mediterranean shrub using a multiphase formulation. *Computers and Mathematics with Applications*, 57:1089–1101, 2009.

[25] G. Puigt and H. Deniau. Cfd e-learning: Euler equation and discretization of convection. Technical report, CERFACS, 2011.

[26] R.E. Kalnay. *Atmospheric Modeling, Data Assimilation and Predictability.* Cambridge University Press, 2003.

[27] D.F Parrish and J.C Derber. The national meteorological center's spectral statistical interpolation analysis system. *Monthly Weather Review*, 120:1747–1763, 1992.

[28] F. Rabier, H. Jarvinen, E. Kilnder, J.F. Mahfouf, and A. Simmons. The ecmwf operational implementation of four-dimensional variational assimilation. part i: Experimental results with simplified physics. *Quarterly Journal of the Royal Meteorological Society*, 126:1143–1170, 2000.

[29] P. Brasseur and J. Verron. The seek filter method for data assimilation in oceanography: a synthesis. *Ocean Dynamics*, 56:650–661, 2006.

[30] Oceanography Society, 22. *Special Issue on the Revolution in Global Ocean Forecasting. GODAE: 10 Years of Achievement*, 2009.

[31] H.E. Emara-Shabaik, Y.A. Khulief, and I. Hussaini. A non-linear multiple-model state estimation scheme for pipeline leak detection and isolation. *Journal of Systems and Control Engineering*, pages 497–512, 2002.

[32] M.C. Rochoux, B. Delmotte, B. Cuenot, S. Ricci, and A. Trouvé. Regional-scale simulations of wildland fire spread informed by real-time flame front observations. *Proceedings of the Combustion Institute*, 34:2641–2647, 2012.

[33] M.C. Rochoux, B. Cuenot, S. Ricci, A. Trouvé, B. Delmotte, S. Massart, R. Paoli, and R. Paugam. Data assimilation applied to combustion. *Comptes Rendus de l'Académie des Sciences - Mécanique*, 341:266–276, 2013.

[34] M.C. Rochoux, S. Ricci, D. Lucor, B. Cuenot, A. Trouvé, and J.M Bart. Towards predictive simulation of wildfire spread using a reduced-cost ensemble kalman filter based on polynomial chaos approximation. In *In Proceedings of the 2012 Summer Program, pages 199 - 208, Center for Turbulence Research, NASA AMES, Stanford University, USA, June 24 to July 20 2012*, 2012.

[35] Jan Mandel, Lynn S. Bennethum, Jonathan D. Beezley, Janice L. Coen, Craig C. Douglas, Kim Minjeong, and Anthony Vodacek. A wildland fire model with data assimilation. *Mathematics and Computers in Simulation*, 79:584–606, 2008.

[36] Y.K. Sasaki. Some basic formalisms in numerical variational analysis. *Monthly Weather Review*, 98:875–883, 1970.

[37] O. Talagrand and P. Courtier. Variational assimilation of meteorological observations with the adjoint vorticity equation. part 1:theory. *Quarterly Journal of the Royal Meteorological Society*, 113:1311–1328, 1987.

[38] A. Tarantola. *Inverse Problem Theory: Methods for Data Fitting and Model Parameter Estimation.* Elsevier, 1987.

[39] A. Gelb. *Applied Optimal Estimation.* MIT Press, Cambridge, Massachusetts, 1974.

[40] O. Talagrand. Assimilation of observations, an introduction. *Journal of the Meteorological Society of Japan*, 75:191–209, 1997.

[41] P. Courtier, Thépaut J.-N., and A. Hollingsworth. A strategy for operational implementation of 4d-var using an incremental approach. *Quarterly Journal of the Royal Meteorological Society*, 120:1367–1387, 1994.

[42] F. Bouttier and P. Courtier. *Data Assimilation concepts and methods*, March 1999.

[43] B. Delmotte, S. Ricci, M. Rochoux, and O. Thual. *Introduction to data assimilation. Application to the calibration of a model parameter on 1D diffusion equation.* CERFACS, March 2011.

[44] Rudolph Emil Kalman. A new approach to linear filtering and prediction problems. *Transactions of the ASME–Journal of Basic Engineering*, 82(Series D):35–45, 1960.

[45] Garry A. Einicke. *Smoothing, Filtering and Prediction - Estimating the past, the present and future.* Intech, 2012.

[46] H.E. Rauch, F. Tung, and C.T. Striebel. Maximum likelihood estimates of linear dynamics systems. *The American Institute of Aeronautics and Astronautics*, 3(8):1445–1450, August 1965.

[47] G.J. Bierman. *Factorization Methods for Discrete Sequential Estimation.* Academic Press, 1977.

[48] A. Deckmyn and L. Berre. A wavelet approach to representing background error covariances in a limited-area model. *Monthly Weather Review*, 133:1279–1294, May 2005.

[49] Y. Michel. Estimating deformations of random processes for correlation modelling: methodology and the one-dimensional case. *Quarterly Journal of the Royal Meteotological Society*, 2007.

[50] P. Courtier, E. Andersson, W. Heckley, J. Pailleux, D. Vasiljevic, M. Hamrud, A. Hollingsworth, F. Rabier, and M. Fisher. The ecmwf implementation of three-dimensional variational assimilation. i:formulation. *Quarterly Journal of the Royal Meteotological Society*, 124:1783–1807, 1998.

[51] O. Pannekoucke, L. Berre, and G. Desroziers. Background error correlation length-scale estimates and their sampling statistics. *Quarterly Journal of the Royal Meteotological Society*, 134:497–511, 2008.

[52] Geir Evensen. Sequential data assimilation with nonlinear quasi-geotrophic model using monte carlo methods to forecast error statistcis. *Journal of Geophysical Research*, 99:143–162, 1994.

[53] Geir Evensen. The ensemble kalman filter : theoritical formulation and practical implementation. *Ocean Dynamics*, 53:539–560, 2004.

[54] G. Burgers, P. Van Leeuwen, and G. Evensen. Analysis scheme in the ensemble kalman filter. *Monthly Weather Review*, 126:1719–1724, 1998.

[55] Jasper A. Vrugt, Cees G. H. Diks, Hoshin V. Gupta, Willem Bouten, and Jacobus M. Verstraten. Improved treatment of uncertainty in hydrologic modeling: Combining the strengths of global optimization and data assimilation. *Water Resources Research*, 41, 2005.

[56] Konstantinos M. Andreadis, Elizabeth A. Clark, Dennis P. Lettenmaier, and Alsdorf? Douglas E. Prospects for river discharge and depth estimation through assimilation of swath-altimetry into a raster-based hydrodynamics model. *Geophysical Research Letters*, 34, 2007.

[57] Michael Durand, Konstantinos M. Andreadis, Douglas E. Alsdorf, Dennis P. Lettenmaier, Delwyn Moller, and Matthew Wilson. Estimation of bathymetric depth and slope from data assimilation of swath altimetry into a hydrodynamic model. *Geophysical Research Letters*, 35, 2008.

[58] Jonathan D. Beezley and Jan Mandel. Morphing ensemble kalman filters. In *Tellus 60A*, 2007.

[59] Hamid Moradkhani, Soroosh Sorooshian, Hoshin V. Gupta, and Paul R. Houser. Dual state-parameter estimation of hydrological models using ensemble kalman filter. *Advances in Water Ressources*, 28:135–147, 2005.

[60] R. Paugam, M.J. Wooster, and G. Roberts. Use of handheld thermal imager data for airborne mapping of fire radiative power and energy and flame front rate of spread. *Geoscience and Remote Sensing*, 51:3385–3399, 2013.

[61] M.J. Wooster, G. Roberts, G. Perry, and Y.J. Kaufman. Retrieval of biomass combustion rates and totals from fire radiative power observations: Frp derivation and calibration relationships between biomass consumption and fire radiative energy release. *Journal of Geophysical Research*, 110:D24311, 2005.

[62] S. Buis, A. Piacentini, and D. Déclat. Palm: a computational framework for assembling high-performance computing applications. *Concurrency and Computation: Practice and Experience*, 18:231–245, February 2006.

[63] A. Fouilloux and A. Piacentini. The palm project: Mpmd paradigm for an oceanic data assimilation software. *Lect. Notes Comput. Sci.*, 1685:1423–1430, 1999.

[64] T. Lagarde, A. Piacentini, and O. Thual. A new representation of data assimilation methods: the palm flow charting approach. *Quarterly Journal of the Royal Meteorological Society*, 127:189–207, 2001.

[65] N. Ferrya, E. Rémya, P. Brasseurb, and C. Maesc. The mercator global ocean operational analysis system: Assessment and validation of an 11-year reanalysis. *Journal of Marine Systems*, 65:540–560, March 2007.

[66] S. Massart, C. Clerbaux, D. Cariolle, A. Piacentini, S. Turquety, and J. Hadji-Lazaro. First steps towards the assimilation of iasi ozone data into the mocage-palm system. *Atmospheric Chemistry and Physics*, 9:5073–5091, 2009.

[67] S. Ricci, A. Piacentini, O. Thual, E. Le Pape, and G. Jonville. Correction of upstream flow and hydraulic state with data assimilation in the context of flood forecasting. *Hydrology and Earth System Sciences*, 15:3555–3575, 2011.

[68] D. Poitou, M. ElHafi, and B. Cuenot. Analysis of radiation modeling for turbulent combustion : development of a methodology to couple turbulent combustion and radiative heat transfer in les. *Journal of Heat Transfer*, 133:10 pages, 2011.

[69] D. Poitou, J. Amaya, M. ElHafi, and B. Cuenot. Analysis of the interaction between turbulent combustion and thermal radiation using unsteady coupled les/dom simulations. *Combustion and Flame*, 159:1605–1618, April 2012.

[70] CERFACS The Palm Team. Openpalm, January 2010.

[71] J. Adjizian-Gerard and B. Ambroise. Topographie et précipitatopn locale : approche trigonométrique. *Annales de Géogrpahie*, t. 104, n. 581-582:173–177, 1995.

[72] D. Sharon. The distribution of hydrologically effective rainfall incident on sloping ground. *Journal Of Hydrology*, 46:165–188, 1980.

[73] R. Weise and G.S. Biging. A qualitative comparison of fire spread models incorporating wind and slope effects. *Forest Science*, 43(2):170–180, 1997.

[74] J.J. Sharples. Review of formal methodologies for wind-slope correction of wildfire rate of spread. *International Journal of Wildland Fire*, 17:179–193, 2008.

[75] C. Lautenberger. Wildland fire modelng with an eulerian level set method and automated calibration. submitted to Fire Safety Journal, 2013.

[76] G. Vico and A. Porporato. Probabilistic description of topographic slope and aspect. *Journal of Geophysical Reasearch*, 114:1–13, 2009.

[77] J.-B. Filippi, X. Pialat, and C.B. Clements. Assessment of forefire/meso-nh for wildland fire/atmosphere coupled simulation of the fireflux experiment. *Proceedings of the Combustion Institute*, 24:2633–2640, 2013.

Appendix A

Algorithm to cycle the analysis

The following algorithm is required to construct the progress variable field from the analysis fire front $\mathbf{X}^a$. The reconstruction algorithm to apply to each mesh nodes leans on vectorial analysis and is as follows: a mesh node N is located in the computational domain by its two-dimensional coordinates within the domain frame $N(x_N, y_N)$.

1. Fig. (A.1a): Calculate the set of distances $d_{N,j},\ j = 1 \ldots N_{fr}$ between the current mesh node N and all the points along the analysis front $\mathbf{X}^a$ we need to rebuild c from:

$$\forall j = 1 \ldots N_{fr},\ d_{N,j} = \sqrt{(x_j^a - x_N)^2 + (y_j^a - y_N)^2}$$

2. Fig. (A.1b): Choose the point along the front, denoted as A, which the associated distance $d_{N,A}$ is the minimal among all the distances $d_{N,j},\ j = 1 \ldots N_{fr}$,

3. Fig. (A.1c): Pick the point along the front, this one denoted as B, "neighbor" to A in the front, the closest to N. By "neighbor", it means the discretization point right after (or right before) the considered point in the front description,

4. Fig. (A.1d): Determine the two-dimensional components of the vector passing through N and orthogonally intersecting the line defined by the points A and B. The intersection point between the line (AB) and this vector is denoted by O which coordinates can be calculated using linear algebra properties,

5. Fig. (A.1e): Calculate the two-dimensionnal inner product between the vector $\overrightarrow{ON}$ and the vector $\overrightarrow{n_A}$ defined as the outgoing normal to the front at the point A. The sign of the inner product will set if the mesh node N is or not in the burnt zone:

$$\begin{cases} \overrightarrow{n_A} \cdot \overrightarrow{ON} < 0 & \Rightarrow \quad N \text{ is inside the burnt area,} \\ \overrightarrow{n_A} \cdot \overrightarrow{ON} = 0 & \Rightarrow \quad N \text{ is on the front,} \\ \overrightarrow{n_A} \cdot \overrightarrow{ON} > 0 & \Rightarrow \quad N \text{ is outside the burnt area,} \end{cases}$$

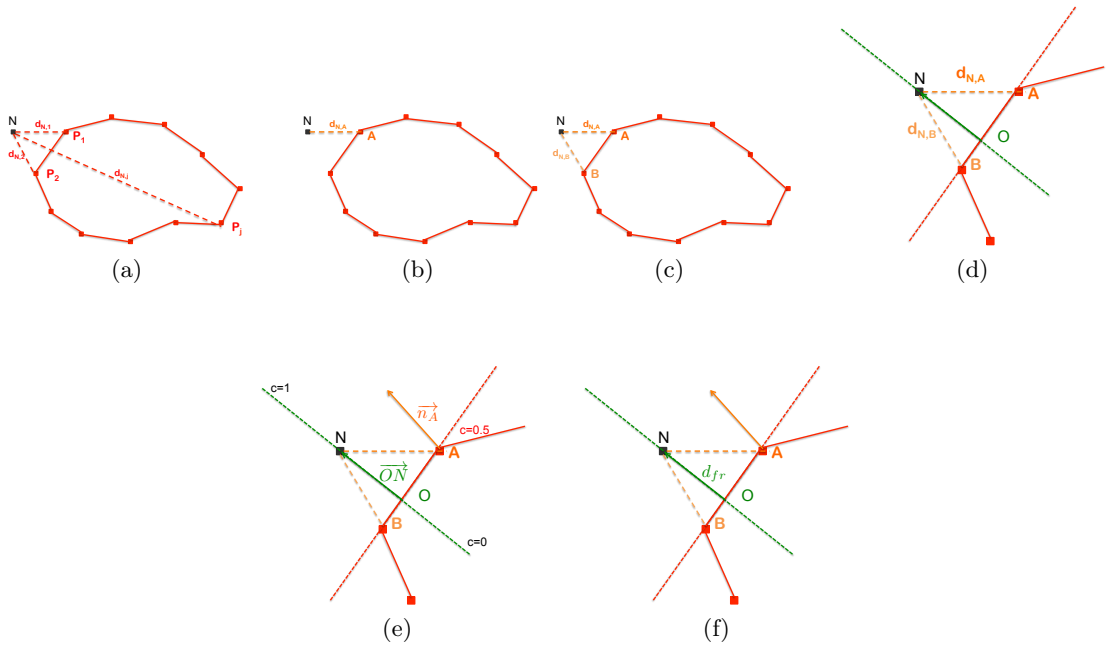


Figure A.1: Details of the steps of the reconstruction algorithm for one mesh node N.

6. Fig. (A.1f): Calculate the value of $c(N)$ with respect to the distance of N to the fire front d_{fr}. This distance is defined as the magnitude of the vector $\overrightarrow{ON}$. If the distance is small enough, the mesh node might be in the transitional part of the field which takes values between 0 and 1. This transition is modeled by a hyperbolic tangent function scaled to take values between 0 and 1 different from the classical limits of the function (i.e. -1 and 1):

$$c(N) = \begin{cases} 0.5 \times (1 - \tanh(-\gamma d_{fr})) & \text{if} \quad \overrightarrow{n_A} \cdot \overrightarrow{ON} < 0, \\ 0.5 & \text{if} \quad \overrightarrow{n_A} \cdot \overrightarrow{ON} = 0, \\ 0.5 \times (1 - \tanh(\gamma d_{fr})) & \text{if} \quad \overrightarrow{n_A} \cdot \overrightarrow{ON} > 0 \end{cases}$$

with γ a scaling constant depending on the mesh grid to make sure that the decrease between 0 and 1 is fast enough.

Appendix B

Reference frames and rotation matrices

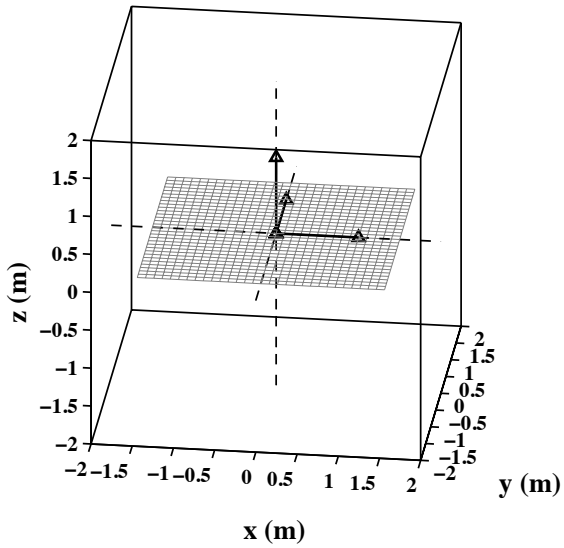
This annexe details the description of the different reference frames and remarquable planes used in the description of the characteristic angles for the topography in Chapter 4 section 1.1. The mathematical description of the axis and rotations is usefull for the calculation of the projections and the definition of the characterization of the planes

The original frame $\mathcal{R}_0$ is the classical $\mathbb{R}^3$ reference frame with the axis $(x_0,\ y_0,\ z_0)$. As illustrated in Fig. (B.1a), x_0 indicates the East direction, y_0 indicates the North direction and z_0 is the vertical direction in the increasing z-values.

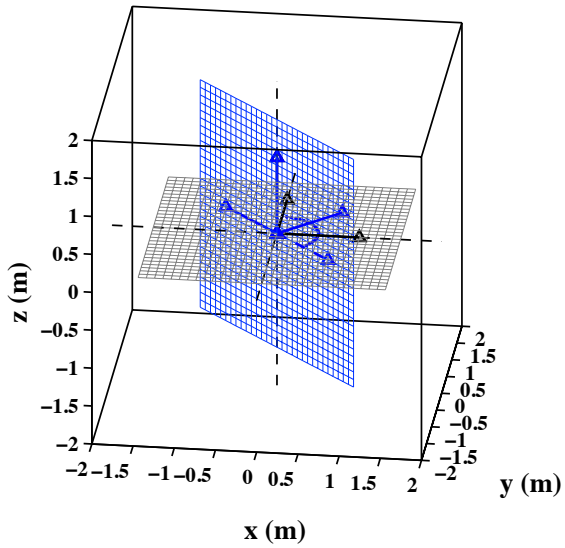
B.1 From the original reference frame to the aspect frame

The aspect angle γ_a is the angle defined in the horizontal plane indicating the downslope direction. Using this angle, we define a second reference frame $\mathcal{R}_a(x_a, y_a, z_0)$ such that the second axis y_a indicates the upslope direction projected in the horizontal plane. This frame is therefore obtained as the rotation of $\mathcal{R}_0$ around z_0 by an angle $\pi - \gamma_a$ in radians (see Fig. (B.1b)). The associated rotation matrix R_0^a is defined as

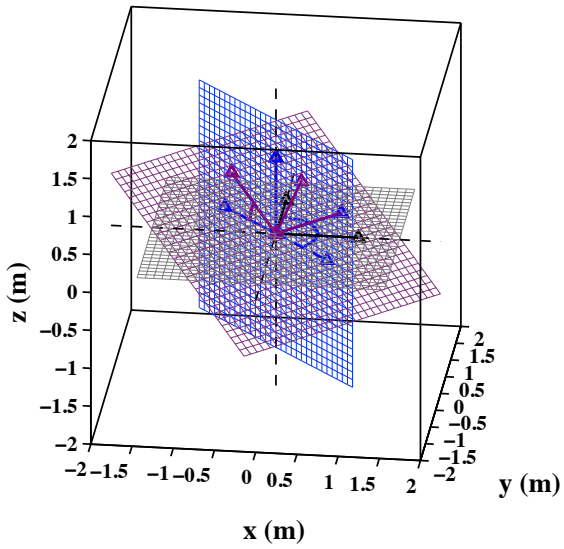
$$\mathsf{R}_0^a = \begin{pmatrix} \cos(\pi - \gamma_a) & -\sin(\pi - \gamma_a) & 0 \\ \sin(\pi - \gamma_a) & \cos(\pi - \gamma_a) & 0 \\ 0 & 0 & 1 \end{pmatrix}. \tag{B.1}$$



(a) $\mathcal{R}_0$



(b) $\mathcal{R}_a$



(c) $\mathcal{R}_s$

Figure B.1: 3D reference frames to describe the slope variables

B.2 From the aspect frame to the slope frame

The slope angle γ_s defines the stepness of the slope. A reference frame adapted to the slope orientation is the one obtained as the rotation of $\mathcal{R}_a$ around x_a of an angle γ_s. As shown in Fig. (B.1c), the final slope frame $\mathcal{R}_s(x_a, y_s, z_s)$ is set such that the axis y_s is in the slope plane and indicates the upslope direction and the axis z_s is the normal direction to the slope plane. The rotation matrix is defined as

$$\mathsf{R}_a^s = \begin{pmatrix} \cos^2(\gamma_a) + \sin^2(\gamma_a)\cos(\gamma_s) & -\cos(\gamma_a)\sin(\gamma_a)(1-\cos(\gamma_s)) & \sin(\gamma_a)\sin(\gamma_s) \\ -\cos(\gamma_a)\sin(\gamma_a)(1-\cos(\gamma_s)) & \sin^2(\gamma_a) + \cos^2(\gamma_a)\cos(\gamma_s) & \cos(\gamma_a)\sin(\gamma_s) \\ -\sin(\gamma_a)\sin(\gamma_s) & -\cos(\gamma_a)\sin(\gamma_s) & \cos(\gamma_s) \end{pmatrix}. \quad \text{(B.2)}$$

Therefore the unitary normal vector to the slope plane is given by the third column of the matrix:

$$z_s = \begin{pmatrix} \sin(\gamma_a)\sin(\gamma_s) \\ \cos(\gamma_a)\sin(\gamma_s) \\ \cos(\gamma_s) \end{pmatrix}. \quad \text{(B.3)}$$

Finally the slope plane $(\mathcal{P}_s)$ is such that:

$$\mathcal{P}_s = \left\{ (x, y, z) \in \mathbb{R}^3, \sin(\gamma_a)\sin(\gamma_s)x + \cos(\gamma_a)\sin(\gamma_s)y + \cos(\gamma_s)z = 0 \right\} \quad \text{(B.4)}$$

Note that this definition of the plane is within a constant. With the above definition, we consider that the plane contains the origin of frame defined as the triplet $(0, 0, 0)$.

Appendix C

Determination of the projection coefficient for the ROS

Let's first recall the definition of some important vectors:

- $\mathbf{n}_{flame,\mathcal{P}_0}$: in the flat terrain case, this vector indicates the normal direction to the fire front so the spreading direction. In the 3D frame $\mathcal{R}_0$ introduced in annexe B, its coordinates are given by

$$\mathbf{n}_{flame,\mathcal{P}_0} \begin{pmatrix} \sin(\theta) \\ \cos(\theta) \\ 0 \end{pmatrix}. \tag{C.1}$$

  We recall that θ is the angle starting from the North direction and defined by turning in the clockwise sense within the horizontal plane.

- $\mathbf{n}_{flame,\mathcal{P}_s}$: this vector is the one we wish to determine here in order to find its projected into the horizontal plane. This 3D vector is in the slope plane ($\mathcal{P}_s$) and is the unitary vector indicating the spreading direction (as $\mathbf{n}_{flame,\mathcal{P}_0}$) in the slope plane. Its coordinates are of the form

$$\mathbf{n}_{flame,\mathcal{P}_s} \begin{pmatrix} \alpha_1 \sin(\theta) \\ \alpha_2 \cos(\theta) \\ \alpha_3 \end{pmatrix}. \tag{C.2}$$

To determine the coefficients α_1, α_2, α_3, we proceed as follows:

1. We determine the 3D coordinates of the points Q_S observed in Fig. (C.1) such that the vector $\mathbf{n}_{flame,\mathcal{P}_0}$ is the projected of $\overrightarrow{OQ_S}$ into the horizontal plane,
2. then $\mathbf{n}_{flame,\mathcal{P}_s}$ is determined by normalizing $\overrightarrow{OQ_S}$:

$$\mathbf{n}_{flame,\mathcal{P}_s} = \frac{\overrightarrow{OQ_S}}{\left\|\overrightarrow{OQ_S}\right\|} \tag{C.3}$$

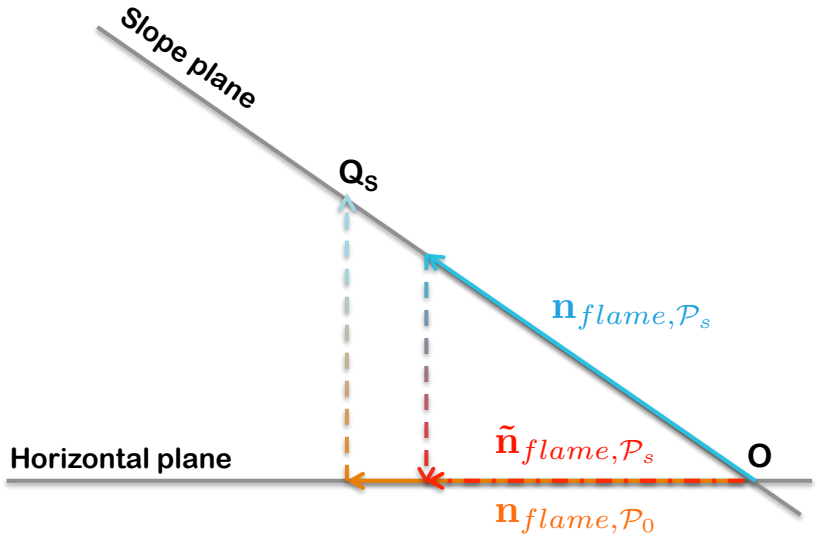


Figure C.1: Visualization of the different flame vectors implied in the determination of the projected ROS

Let's now determine the coordinates of $Q_S(x_{Q_S}, y_{Q_S}, z_{Q_S})$, using the following assumptions:

- $Q_S \in (\mathcal{P}_s)$ giving that

$$\sin(\gamma_a)\sin(\gamma_s)x_{Q_S} + \cos(\gamma_a)\sin(\gamma_s)y_{Q_S} + \cos(\gamma_s)z_{Q_S} = 0 \tag{C.4}$$

- The point with the coordinates $\begin{pmatrix}\sin(\theta) & \cos(\theta) & 0\end{pmatrix}^T$ is the orthogonal projected of Q_S into the horizotal plane. This implies that Q_S is in the line with orientation vector $\begin{pmatrix}0 & 0 & 1\end{pmatrix}^T$ containing the point with coordinates $\begin{pmatrix}\sin(\theta) & \cos(\theta) & 0\end{pmatrix}^T$.

So we need to find the real t such that

$$\begin{aligned}
&\sin(\gamma_a)\sin(\gamma_s)x_{Q_S} + \cos(\gamma_a)\sin(\gamma_s)y_{Q_S} + \cos(\gamma_s)z_{Q_S} = 0 \\
&x_{Q_S} = \sin(\theta) \\
&y_{Q_S} = \cos(\theta) \\
&z_{Q_S} = 0
\end{aligned} \tag{C.5}$$
$$\tag{C.6}$$

We have then

$$\begin{aligned}
& \sin(\gamma_a)\sin(\gamma_s)\sin(\theta) + \cos(\gamma_a)\sin(\gamma_s)\cos(\theta) + \cos(\gamma_s)t &&= && 0 \\
\Leftrightarrow \quad & -\sin(\gamma_s)\left(\cos(\gamma_a)\cos(\theta) + \sin(\gamma_a)\sin(\theta)\right) &&= && \cos(\gamma_s)t \\
\Leftrightarrow \quad & -\tan(\gamma_s)\cos(\gamma_a - \theta) &&= && t
\end{aligned}$$

Therefore, the coordinates of the vector $\overrightarrow{OQ_S}$ are

$$\overrightarrow{OQ_S}\begin{pmatrix}\sin(\theta)\\ \cos(\theta)\\ -\tan(\gamma_s)\cos(\gamma_a-\theta)\end{pmatrix} \tag{C.7}$$

Also,

$$\left\|\overrightarrow{OQ_S}\right\| = \sqrt{1+\tan^2(\gamma_s)\cos^2(\gamma_a-\theta)} \tag{C.8}$$

We deduce then that the unitary vector $\mathbf{n}_{flame,\mathcal{P}_s}$ is defined as

$$\mathbf{n}_{flame,\mathcal{P}_s}\begin{pmatrix}\frac{\sin(\theta)}{\sqrt{1+\tan^2(\gamma_s)\cos^2(\gamma_a-\theta)}}\\ \frac{\cos(\theta)}{\sqrt{1+\tan^2(\gamma_s)\cos^2(\gamma_a-\theta)}}\\ \frac{-\tan(\gamma_s)\cos(\gamma_a-\theta)}{\sqrt{1+\tan^2(\gamma_s)\cos^2(\gamma_a-\theta)}}\end{pmatrix}. \tag{C.9}$$

And the 3D ROS vector in the slope plane is given by $\mathbf{\Gamma} = \Gamma\mathbf{n}_{flame,\mathcal{P}_s}$.

Finally, the projected ROS vector in the horizontal plane has to be of the form

$$\widetilde{\mathbf{\Gamma}} = \widetilde{\Gamma}\mathbf{n}_{flame,\mathcal{P}_0} \tag{C.10}$$

The projected of $n_{flame,\mathcal{P}_s}$ into the horizontal plane is given by

$$\widetilde{\mathbf{n}}_{flame,\mathcal{P}_s}\begin{pmatrix}\frac{\sin(\theta)}{\sqrt{1+\tan^2(\gamma_s)\cos^2(\gamma_a-\theta)}}\\ \frac{\cos(\theta)}{\sqrt{1+\tan^2(\gamma_s)\cos^2(\gamma_a-\theta)}}\\ 0\end{pmatrix} = \frac{n_{flame,\mathcal{P}_0}}{\sqrt{1+\tan^2(\gamma_s)\cos^2(\gamma_a-\theta)}}. \tag{C.11}$$

Therefore, the projected ROS vector is

$$\widetilde{\mathbf{\Gamma}} = \Gamma\frac{n_{flame,\mathcal{P}_0}}{\sqrt{1+\tan^2(\gamma_s)\cos^2(\gamma_a-\theta)}}, \tag{C.12}$$

and the projected value of the ROS is

$$\widetilde{\Gamma}\frac{\Gamma}{\sqrt{1+\tan^2(\gamma_s)\cos^2(\gamma_a-\theta)}}. \tag{C.13}$$

Appendix D

Slope correction on a basic canyon

D.1 Mathematical and numerical description of a canyon

A canyon is the reunion of two non parallel planes with facing aspects. The two faces of the canyon, denoted as $(\mathcal{P}_1)/(\mathcal{P}_2)$, are characterized by one aspect angle each, $\gamma_{a,1}/\gamma_{a,2}$, and a slope angle each, $\gamma_{s,1}/\gamma_{s,2}$. The intersection between the two planes is a line denoted as (d_c) in the 3D-space. Its projection in the horizontal plane, denoted by $(\overline{d_c})$, is also a line pointing in a direction defined by the angle θ_c. As for aspect or slope angles, θ_c is defined from the North direction, by turning in the clockwise direction. The vertical direction of (d_c) is defined by the angle γ_c. To define these planes and lines equation, we need to fix a point $S(x_S, y_S, z_S)$ which is contained by the two planes and the line. For a computational domain defined by the x- and y-limits $x_{\min},\ x_{\max},\ y_{\min},\ y_{\max}$, S is chosen to be the point defined by:

$$S(\frac{x_{\max} + x_{\min}}{2},\ 0,\ 0) \tag{D.1}$$

A summary of these features is presented in Fig. (D.1).

The line (d_c) in the set of points (x, y, z) such that:

$$\left\{ \begin{array}{rcl} x & = & \sin(\theta_c)\cos(\gamma_c)t + x_S, \quad t \in \mathbb{R} \\ y & = & \cos(\theta_c)\cos(\gamma_c)t \\ z & = & \sin(\gamma_c)t \end{array} \right. \tag{D.2}$$

The plane $(\mathcal{P}_1)$ is the of points (x, y, z) such that:

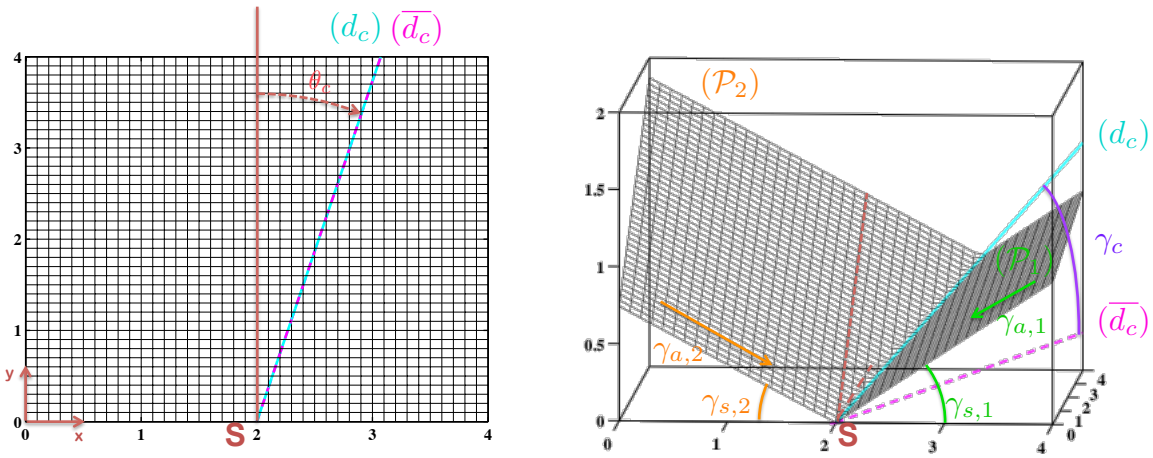
$$\sin(\gamma_{a,1})\sin(\gamma_{s,1})(x - x_S) + \cos(\gamma_{a,1})\sin(\gamma_{s,1})y + \cos(\gamma_{s,1})z = 0 \tag{D.3}$$

Finally, $(\mathcal{P}_2)$ is the of points (x, y, z) such that:

$$\sin(\gamma_{a,2})\sin(\gamma_{s,2})(x - x_S) + \cos(\gamma_{a,2})\sin(\gamma_{s,2})y + \cos(\gamma_{s,2})z = 0 \tag{D.4}$$

To be sure to define realistic canyon in FIREFLY, the user has to choose the values of the angles θ_c, γ_c, $\gamma_{s,1}$ and $\gamma_{s,2}$ given their subset of definition:

$$\theta_c \in]-90°;\ 90°[\quad \gamma_c \in [0°;\ \min(\gamma_{s,1},\ \gamma_{s,2})] \quad \gamma_{s,1}/\gamma_{s,2} \in [0°;\ 90°[\tag{D.5}$$



(a) View in the horizontal plane

(b) 3D view

Figure D.1: Mathematical formalism associated with the definition of a numerical canyon.

Then, FIREFLY will determine the aspect angles $\gamma_{a,1}$ and $\gamma_{a,2}$ given these conditions and also given that

$$\theta_c - \pi \leq \gamma_{a,1} \leq \theta_c \quad \text{and} \quad \theta_c \leq \gamma_{a,2} \leq \theta_c + \pi \tag{D.6}$$

Given the hypothesis (D.5) and (D.6), $(\gamma_{a,1}, \gamma_{a,2})$ are solutions of the following systems:

- $\theta_c - \pi \leq \gamma_{a,1} \leq \theta_c$ and $\gamma_{a,1}$ is solution of the system composed of equations (D.2,D.3),
- $\theta_c \leq \gamma_{a,1} \leq \theta_c + \pi$ and $\gamma_{a,2}$ is solution of the system composed of equations (D.2,D.4).

Note that it has been proven that it exits an unique pair $(\gamma_{a,1}, \gamma_{a,2})$ that verifies (D.5) and (D.6) and are solutions of the above systems. The proof is presented in Annexe E.

D.2 Validation on a limit case

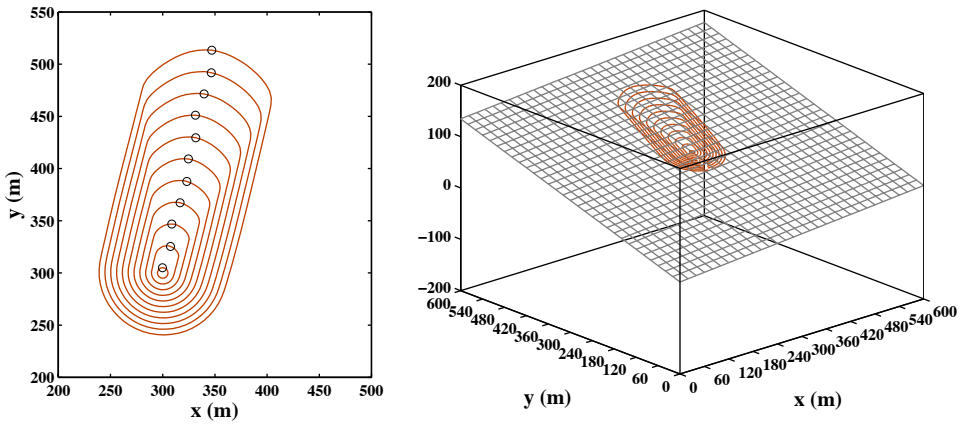
According to those previous definitions, the canyon configuration tends to a plane configuration when $\gamma_{s,1}$ and $\gamma_{s,2}$ tends to γ_c. When these three angles are equal, the resulting topography is a slope plane with slope angle $\gamma_s = \gamma_c$ and aspect angle $\gamma_a = \theta_c \pm 180°$. We need then to verify that the spread on this particular "canyon" configuration is equivalent to a spread on a slope plane defined by the anglesγ_s and γ_a seen above.

The simulation configuration with the different angles is presented in Table (D.1) and the resulting fronts are illustrated in Figure (D.2).

As illustrated in Fig. (D.2b), the topography presents a slope plane configuration and the resulting fire fronts in Fig. (D.2a) are consistent with the fire fronts configuration obtained in Fig. (4.4) when the wildfire is spreading on a simple slope plane. We need then to make sure that the angles and the ROS are correct.

Domain		600 m×600 m, $dx = dy = 1$ m
Time		1000 s, $dt = 0.5$ s
Ignition config		circular
Fuel depth	δ_v	1 (m)
Fuel moisture content	M_v	0.15 (%)
Surface-to-volume ratio	Σ_v	11500 (m^{-1})
Wind velocity	u_w	0 ($\mathrm{m.s}^{-1}$)
Canyon angles	θ_c	15°
	γ_c	15°
	$\gamma_{s,1}$	15°
	$\gamma_{s,2}$	15°

Table D.1: Configuration for the canyon validation



(a) 2D view

(b) 3D view

Figure D.2: Resulting fronts for a propagation on a plane with a zero-slope

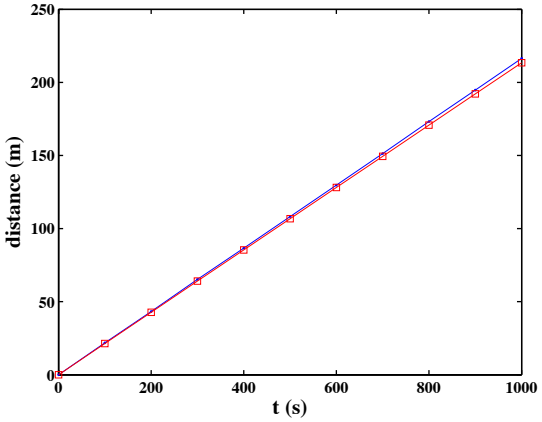


Figure D.3: Comparison between the theoretical and experimental velocities. red-squared line is for the theoretical ROS and blue-dotted line is for the experimental ROS.

From the execution, the calculated aspect angles for each side of the canyon were:

$$\gamma_{a,1} = -2.879 \text{ rad} = -164.955^\circ \simeq \theta_c - 180^\circ \quad \gamma_{a,2} = 3.403 \text{ rad} = 194.97^\circ \simeq \theta_c + 180^\circ \tag{D.7}$$

The simulation was then able to determine the right aspect angle. The determination of the ROS will help us verify the slope angle. In the upslope direction, the theoretical ROS is defined as the optimal ROS with a slope of 15°. Given the fuel characteristics, the projected ROS is equal to $\Gamma_{\text{th}} = 0.2135$ m.s^{-1}. The black circular markers in Fig. (D.2a) are the markers used to determine the experimental covered distance. This latter is compared to the theoretical one with respect to the time in the Figure (D.3). The mean experimental velocity is in accordance with the theoretical velocity and equal to $\Gamma_{\text{th}} = 0.2166$ m.s^{-1}.

D.3 Wildfire spreads on a canyon topography

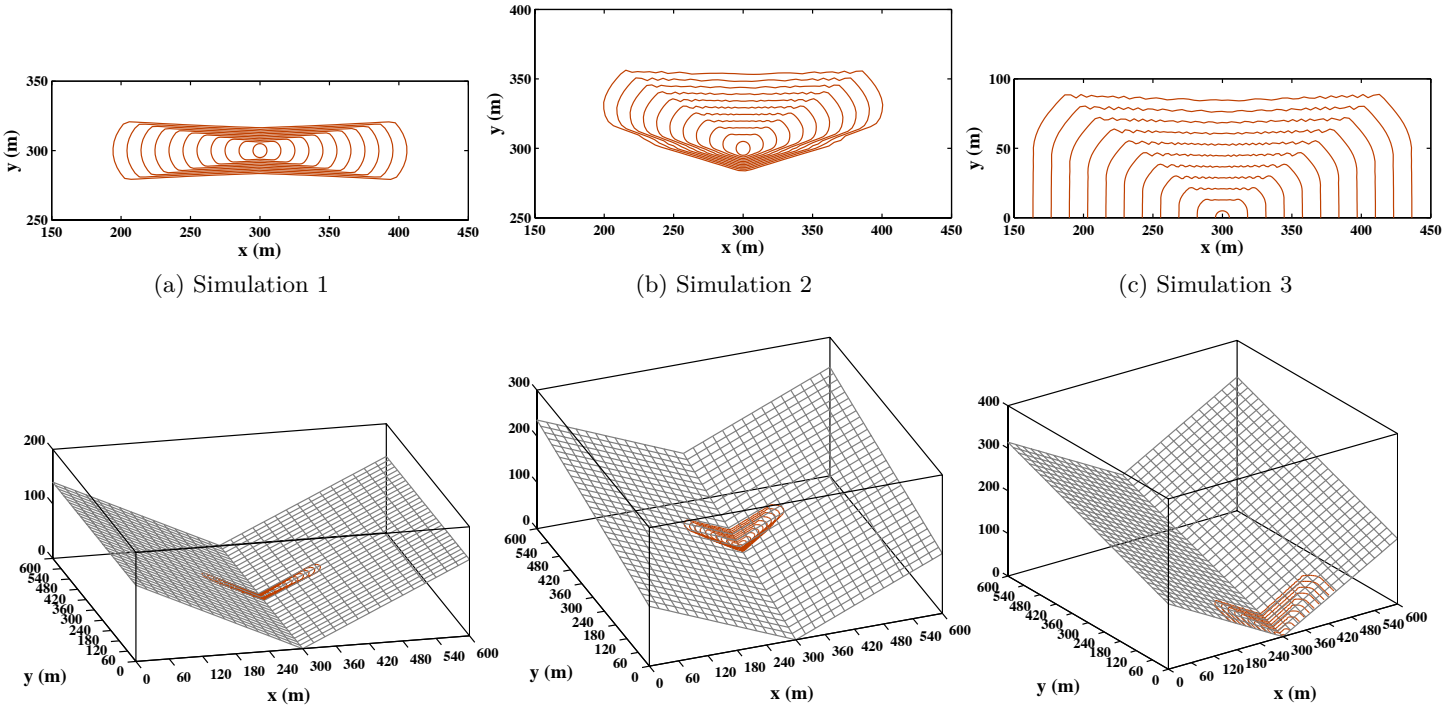
For these simulations, the fuel characteristics are given in Tab. (D.2) and no wind is implied. Three different simulations are run, varying from the initial fire configuration to the values of the angles that define the canyon, the details are given in Tab. (D.3). The resulting fronts are presented in Fig. (D.4) with the fronts observed every 20 s. Visually, the highest ROS are in directions corresponding to the aspect directions given in Tab. (D.3).

Domain		600 m×600 m, $dx = dy = 1$ m
Time		200 s, $dt = 0.5$ s
Fuel depth	δ_v	1 (m)
Fuel moisture content	M_v	0.15 (%)
Surface-to-volume ratio	Σ_v	11500 (m^{-1})

Table D.2: Fuel characteristics for the simulations on a canyon topography.

	Simulation 1	Simulation 2	Simulation 3
Ignition config	circular	circular	cone
θ_c (degree)	0	0	0
γ_c (degree)	0	15	15
$\gamma_{s,1}$ (degree)	25	25	25
$\gamma_{s,2}$ (degree)	25	25	25
$\gamma_{a,1}$ (degree)	-90	-125.0721	-125.0721
$\gamma_{a,2}$ (degree)	90	125.0721	125.0721

Table D.3: Angles characteristics for the simulations on a canyon topography.



(a) Simulation 1 (b) Simulation 2 (c) Simulation 3

(d) Simulation 1 (e) Simulation 2 (f) Simulation 3

Figure D.4: Simulated fronts spreading in a canyon topography, observed every 20 seconds. (a-c) represent the fronts projected in the horizontal plane and (d-f) represent the fronts in the 3D domain.

Appendix E

Existence and uniqueness of a pair $(\gamma_{a,1}, \gamma_{a,2})$ to describe a canyon topography

E.1 Recall on the numerical definition of a canyon topography

A canyon is made of two secant planes with facing aspects. The two slope planes $(\mathcal{P}_1)/(\mathcal{P}_2)$ are characterized by an aspect angles $\gamma_{a,1}/\gamma_{a,2}$ and a slope angles $\gamma_{s,1}/\gamma_{s,2}$. Also, the two planes are secant in a line (d_c) with an orientation angle (in the horizontal plane) θ_c and a slope γ_c. It is chosen that the two slopes are secant in a point with the coordinates:

$$S\left(\frac{x_{\max}+x_{\min}}{2}, 0, 0\right) \tag{E.1}$$

with $x_{\max}$ and $x_{\min}$ the limits of the computational domain along the x-axis. This particular point is required to define an unique canyon. We have then:

$$-\frac{\pi}{2} < \theta_c < \frac{\pi}{2} \quad \text{and} \quad -\frac{\pi}{2} < \gamma_c < \frac{\pi}{2} \tag{E.2}$$

Also:

$$M(x,y,z) \in (d_c) \Longleftrightarrow \exists t \in \mathbb{R} \begin{cases} x &= \cos(\gamma_c)\sin(\theta_c)t + x_S \\ y &= \cos(\gamma_c)\cos(\theta_c)t \\ z &= \sin(\gamma_c)t \end{cases} \tag{E.3}$$

To make sure to define realistic canyon, the user will choose the angle θ_c, γ_c, $\gamma_{s,1}$, $\gamma_{s,2}$ respecting the following bounds:

- $-\frac{\pi}{2} < \theta_c < \frac{\pi}{2}$,
- $-\frac{\pi}{2} < \gamma_c < \frac{\pi}{2}$,
- $\gamma_c \leq \gamma_{s,1} \leq \frac{\pi}{2}$,

- $\gamma_c \leq \gamma_{s,2} \leq \frac{\pi}{2}$.

Then, a particular subroutine will determine the corresponding angles $\gamma_{a,1}$ and $\gamma_{a,2}$ knowing that, according to the configuration in Fig. :

$$\theta_c - \pi \leq \gamma_{a,1} \leq \theta_c \quad \text{and} \quad \theta_c \leq \gamma_{a,2} \leq \theta_c + \pi \tag{E.4}$$

First, we want to determine $\gamma_{a,1}$ such that $\theta_c - \pi \leq \gamma_{a,1} \leq \theta_c$ and $(d_c) \subset (\mathcal{P}_1)$ defined with the slope angle $\gamma_{s,1} \in [\gamma_c, \frac{\pi}{2}]$. According to Eq. (E.3) and the definition of the plane $\mathcal{P}_1$, we have that it exists a real t such that

$$\begin{cases} x = \cos(\gamma_c)\sin(\theta_c)t + x_S \\ y = \cos(\gamma_c)\cos(\theta_c)t \\ z = \sin(\gamma_c)t \\ \text{and also} \\ \sin(\gamma_{a,1})\sin(\gamma_{s,1})(x - x_S) + \cos(\gamma_{a,1})\sin(\gamma_{s,1})y + \cos(\gamma_{s,1})z = 0 \end{cases} \tag{E.5}$$

Eq. (E.5) gives us that:

$$\cos(\gamma_{a,1}) = \frac{-\cos(\gamma_{s,1})\sin(\gamma_c) - \sin(\gamma_{a,1})\sin(\gamma_{s,1})\cos(\gamma_c)\sin(\theta_c)}{\sin(\gamma_{s,1})\cos(\gamma_c)\cos(\theta_c)} \tag{E.6}$$

In this expression, $\gamma_{s,1} \neq 0$, else $(\mathcal{P}_1)$ is flat and it unnecessary to determine $\gamma_{a,1}$. The same expression is found to determine $\gamma_{a,2}$ only the difference is the domain of definition of this other angle.

Let's now define the function f defined for $x \in [\theta_c - \pi; \theta_c]$ as

$$f(x) = \cos(x) + \frac{\cos(\gamma_{s,1})\sin(\gamma_c) + \sin(x)\sin(\gamma_{s,1})\cos(\gamma_c)\sin(\theta_c)}{\sin(\gamma_{s,1})\cos(\gamma_c)\cos(\theta_c)} \tag{E.7}$$

$\gamma_{a,1}$ is the solution of the equation

$$f(x) = 0 \tag{E.8}$$

The following study will be to prove that the solution of the Eq. (E.8) is unique over the set $[\theta_c - \pi; \theta_c]$.

E.2 Study of the function f over $[\theta_c - \pi; \theta_c]$

f is continuous and so differentiable over $[\theta_c - \pi; \theta_c]$. We have then that

$$f'(x) = -\sin(x) + \tan(\theta_c)\cos(x) \quad f''(x) = -\tan(\theta_c)\cos(x) - \sin(x) \tag{E.9}$$

The study of the sign of f' and f'' gives us that:

$$f'(\theta_c - \pi) = 0 \quad f'(\theta_c) = 0 \quad f'(\theta_c - \pi) > 0 \quad f''(\theta_c) < 0 \tag{E.10}$$

So f is strictly increasing over $[\theta_c - \pi; \theta_c]$. We need then to verify if f changes its sign in the studied interval by determining the sign of $f(\theta_c - \pi)f(\theta_c)$:

$$f(\theta_c-\pi)f(\theta_c)=\left[-\cos(\theta_c)+\frac{\cos(\gamma_{s,1})\sin(\gamma_c)}{\sin(\gamma_{s,1})\cos(\gamma_c)\cos(\theta_c)}-\frac{\sin^2(\theta_c)}{\cos(\theta_c)}\right].\left[\cos(\theta_c)+\frac{\cos(\gamma_{s,1})\sin(\gamma_c)}{\sin(\gamma_{s,1})\cos(\gamma_c)\cos(\theta_c)}+\frac{\sin^2(\theta_c)}{\cos(\theta_c)}\right] \quad \text{(E.11)}$$

After developing the product in Eq. (E.11) and simplifying, we find that:

$$\begin{aligned}
& f(\theta_c-\pi)f(\theta_c)=-\cos^2(\theta_c)-2\sin^2(\theta_c)-\frac{\sin^4(\theta_c)}{\cos^2(\theta_c)}+\frac{\cot^2(\gamma_{s,1})\tan^2(\gamma_c)}{\cos^2(\theta_c)} \\
\iff & f(\theta_c-\pi)f(\theta_c)=-1-\sin^2(\theta_c)\left(1+\tan^2(\theta_c)\right)+\cot^2(\gamma_{s,1})\tan^2(\gamma_c)\left(1+\tan^2(\theta_c)\right) \\
\iff & f(\theta_c-\pi)f(\theta_c)=-1-\frac{\sin^2(\theta_c)}{\cos^2(\theta_c)}+\cot^2(\gamma_{s,1})\tan^2(\gamma_c)\left(1+\tan^2(\theta_c)\right) \\
\iff & f(\theta_c-\pi)f(\theta_c)=\left(1+\tan^2(\theta_c)\right)\left(\cot^2(\gamma_{s,1})\tan^2(\gamma_c)-1\right) \qquad \text{(E.12)}
\end{aligned}$$

From the beginning, we have supposed that $0<\gamma_c<\gamma_{s,1}<\frac{\pi}{2}$, then:

$$0<\cos(\gamma_{s,1})<\cos(\gamma_c)<1 \quad \text{and} \quad 0<\sin(\gamma_c)<\sin(\gamma_{s,1})<1 \qquad \text{(E.13)}$$

This implies that:

$$0<\frac{\cos(\gamma_{s,1})}{\cos(\gamma_c)}<1 \quad \text{and} \quad 0<\frac{\sin(\gamma_c)}{\sin(\gamma_{s,1})}<1 \qquad \text{(E.14)}$$

Then we find that:

$$0<\cot(\gamma_{s,1})\tan(\gamma_c)<1 \Rightarrow 0<\cot^2(\gamma_{s,1})\tan^2(\gamma_c)<1 \Rightarrow \cot^2(\gamma_{s,1})\tan^2(\gamma_c)-1<0 \qquad \text{(E.15)}$$

Eq. (E.15), added to the fact that $1+\tan^2(\theta_c)>0$, gives us that

$$f(\theta_c-\pi)f(\theta_c)<0 \qquad \text{(E.16)}$$

We can conclude that there is an only real $\gamma_{a,1}\in[\theta_c-\pi;\theta_c]$ such that $f(\gamma_{a,1})=0$ and also that verifies Eq. (E.6). Then $\gamma_{a,1}$ can be numerically determined by the bissection method with a chosen precision.