Universitat Autònoma de Barcelona

Departament d Arquitectura de Computadors i Sistemes Operatius

# A comprehensive methodology to predict forest fires behavior using complementary models

PhD thesis submitted by Carlos Brun Sorribas in fulfillment of the requirements for the PhD degree from Universitat Autònoma de Barcelona advised by Dr. Tomàs Margalef Burrull.

Barcelona (Spain), May 29, 2014

Departament d Arquitectura de Computadors i Sistemes Operatius

PhD degree from Universitat Autònoma de Barcelona advised by Dr. Tomàs Margalef Burrull

# A comprehensive methodology to predict forest fires behavior using complementary models

*Author*
Carlos Brun Sorribas

*Advisor*
Tomàs Margalef Burrull

Advisor signature Author signature

# Acknowledgements

This PhD Thesis would not have been possible without the unconditional support of Dr. Tomàs Margalef and Dr. Ana Cortés. I appreciate your involvement and support in this work, and especially your extraordinary human quality.

I also want to thank all the teammates of the CAOS fire group during these four years: Moni, Darío, Hisham, Kerstin, Andrés, Tomás and Gemma. Although not being a  pyromaniacal , I must thank Abel his help during all these years.

I do not want to forget all teachers, fellows, and technical and administrative staff of the Computer Architecture and Operating Systems department, who make the day-to-day work easier.

I also want to highlight the role of all those who love me and support me outside of work. Family and friends have helped me to continue in difficult times and meeting this challenge.

Finally, I just need to mention the key person in my life. Núria, thank you for your love, perseverance and patience during all these years. Now I have to reward you for putting up with me these past few months.

# Abstract

Large scale forest fires suppose a great challenge due to their impact on society on many levels. This physical phenomenon is an interdisciplinary problem that requires the efforts of researchers across various fields. Therefore, predicting the behavior of forest fires and minimizing the effects of this hazard is the main goal of this work. One of the major problems with this kind of hazard is the uncertainty and imprecision of the input parameters. We rely on a prediction strategy based on calibration techniques that try to tune these parameters to produce a more accurate prediction and reduce the uncertainty of the parameters. In this work, we propose strategies to solve some of the restrictions associated with this technique. The spatial distribution of the parameters has been considered uniform along the terrain, and we introduce a complementary model to simulate the wind behavior over complex terrains, due to the leading role of wind in fire spreading. We also suggest coupling a meteorological model to enrich the knowledge of the system of the temporal evolution of the weather variables. All these components are integrated in a simulation and prediction framework, and a methodology to simulate real forest fire scenarios is defined in which the steps, processes, models and tools to build an effective real-time forest fire assessment system are specified. This system relies on High Performance Computing resources and paradigms to provide a response in an acceptable time due to the high computational cost of evaluating several scenarios and integrating complementary models.

## Keywords

Forest fire spread prediction, model coupling, wind field model, meteorological model, MPI, OpenMP, two-stage prediction.

# Resumen

Los incendios a gran escala suponen un gran reto debido al impacto que tienen en la sociedad a múltiples niveles. Este fenómeno físico es un problema multidisciplinar que requiere del esfuerzo de investigadores de campos diversos. Por lo tanto, predecir el comportamiento de los incendios forestales y minimizar sus efectos son las principales metas de este trabajo. Uno de los problemas principales de este tipo de fenómenos es la incertidumbre e imprecisión de los parámetros de entrada. Para abordar este problema, nos basamos en una estrategia de predicción basada en técnicas de calibración, que trata de sintonizar estos parámetros para proporcionar una predicción más certera, y reducir la incertidumbre sobre los parámetros. En este trabajo, proponemos estrategias para resolver ciertas restricciones asociadas a esta técnica. La distribución espacial de los parámetros ha sido considerada uniforme a lo largo del terreno, por lo que introducimos un modelo complementario para simular el efecto del viento sobre terrenos complejos, debido al papel fundamental del viento en la propagación del fuego. Además, se sugiere acoplar un modelo meteorológico para enriquecer el conocimiento del sistema sobre la evolución temporal de las variables meteorológicas. Todos estos componentes se integran en un entorno de simulación y predicción, y se define una metodología para simular incendios forestales reales donde se especifican los pasos, procedimientos, modelos, y herramientas necesarias para construir un sistema de evaluación del riesgo de incendios forestales en tiempo real. Este sistema se sirve de recursos y paradigmas propios de la Computación de Altas Prestaciones para proporcionar una respuesta en un tiempo aceptable debido al alto coste computacional de evaluar múltiples escenarios, e integrar modelos complementarios.

### Palabras clave

Predicción de incendios forestales, acoplamiento de modelos, campo de vientos, modelo meteorológico, MPI, OpenMP, predicción en dos etapas.

# Resum

Els incendis a gran escala suposen un gran repte degut a l impacte que tenen en la societat a diversos nivells. Aquest fenomen físic és un problema multidisciplinar que requereix de l esforç d investigadors de diferents camps. Així doncs, predir el comportament dels incendis forestals i minimitzar els seus efectes són els principals objectius d aquest treball. Un dels principals problemes d aquests tipus de fenòmens és la incertesa i la imprecisió dels paràmetres d entrada. Per abordar aquest problema, ens basem en una estratègia de predicció basada en tècniques de calibració, que tracten de sintonitzar aquests paràmetres per tal de proporcionar una predicció més precisa, i reduir la incertesa sobre els paràmetres. En aquest treball, proposem estratègies per resoldre certes restriccions associades a aquesta tècnica. La distribució espacial dels paràmetres ha estat considerada uniforme en tot el terreny, per la qual cosa s introdueix un model complementari que simula l efecte del vent sobre terrenys complexes, degut al paper fonamental del vent en la propagació del foc. A més a més, es suggereix acoblar un model meteorològic, per enriquir el coneixement del sistema sobre l evolució temporal de les variables meteorològiques. Tots aquests components s integren en un entorn de simulació i predicció, i es defineix una metodologia per simular incendis forestals reals, en la qual s especifiquen els passos, procediments, models i eines necessàries per a construir un sistema d avaluació del risc d incendis forestals a temps real. Aquest sistema utilitza recursos i paradigmes propis de la Computació d Altes Prestacions per proporcionar una resposta en un temps acceptable, degut a l elevat cost computacional de l avaluació de múltiples escenaris i de la integració de models complementaris.

## Paraules clau

Predicció d incendis forestals, acoblament de models, camp de vents, model meteorològic, MPI, OpenMP, predicció en dues etapes.

# Contents

# List of Tables

# List of Figures

# Chapter 1

# Introduction

Computers have drastically changed the way we do science. Until the emergence of the first computers, scientific theory and experiments were limited by human ability to solve problems. This fact was a barrier to facing complex problems where the numerical calculations required were intractable for us. Computer Science provided the tools and theory necessary to implement processes or algorithms using computers and opened a new era in which computation is present in all areas of our lives.

Computational Science is an interdisciplinary science that focuses its efforts on studying, modeling and solving problems related to science and engineering, using a mathematical background and Computer Science tools and algorithms. Figure 1.1 represents this confluence of fields.

Natural hazards such as floods, hurricanes or forest fires - the focus of our research - are phenomena characterized by a stochastic behavior, the large amount of variables involved and complex physical systems. It is necessary to simplify these systems to be able to model them and perform simulations using computational resources. Despite this simplification, we must maintain a trade-off between model accuracy and ease.

High Performance Computing (HPC) provides the resources, tools and programming paradigms to model and simulate complex applications in an efficient, fast and reliable way. This computing paradigm opens the door to new challenges and allows us to tackle very complex problems that cannot

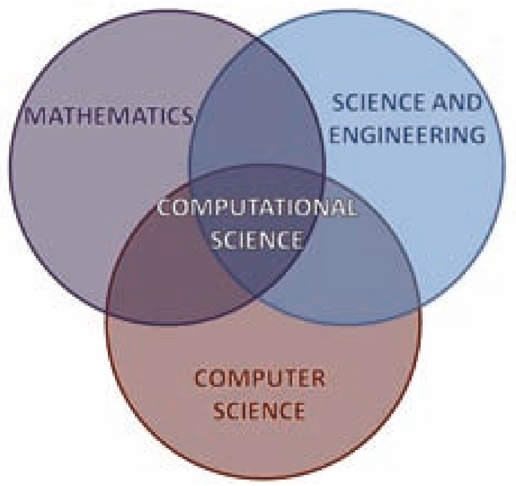


Figure 1.1: Computational Science

be solved in an acceptable time by the classical computing view.

## 1.1 Motivation

The occurrence of forest fires is directly related to the features of the region. There are zones that are more susceptible to suffering from forest fires due to their climatology, vegetation or the management policy of their forests. Forest fires have a global impact and cause severe damage on both environmental and socioeconomic levels. The Joint Research Center, a European research institution under European Commission control, publishes an annual report about forest fires in Europe. The latest report [1], shows the forest fire statistics in Europe, analyzing their impact country by country. Southern European countries (Portugal, Spain, France, Italy, and Greece) are the countries most affected by this hazard due to their Mediterranean climatology. They are characterized by a Mediterranean climate with hot, dry summers and mild winters. These features make these countries critical regions during the hottest seasons.

Figure 1.2 shows the burnt area and the number of fire occurrences since 1980. Focusing on the particular case of Spain, it is the country with the highest average of burnt area decade to decade. Only in the last decade

(1999-2009) was it overcome by Portugal, due to the positive trend in Spain that has allowed for the reduction in the burnt area from an average of 244,788 ha per year in 1980-1989 to 125,239 ha per year in 2000-2009.

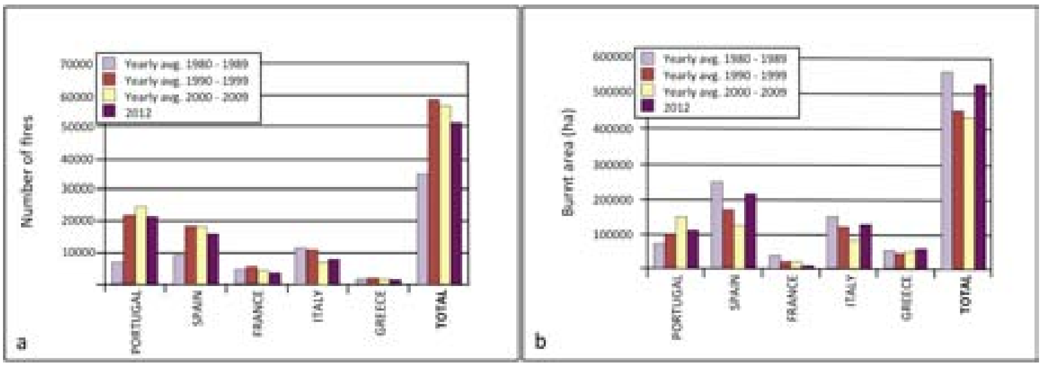


Figure 1.2: Burnt Area and number of fires in Southern European countries

Despite this positive evolution, medium and long-term climatic predictions show that the average temperature will increase, and, consequently, fire risk will be greater year after year. One of the scales that measures fire risk is the SSR (Seasonal Severity Rating). It is derived from the Canadian Fire Weather Index System [2], and offers an objective comparison of fire danger over time in a certain region. Figure 1.3 is an example of a map where SSR is used and shows the average forest fire danger in Europe from 1981 to 2010, as well as the estimated behavior of this measurement over the next several years. These predictions foretell a growth in fire risk throughout most of Europe, so it is necessary to remain vigilant and develop strategies to mitigate the effects of forest fires.

The most visible effects of forest fires are the loss of forest area and human lives. These factors are not negligible, and, every year, we can see examples of large and devastating fires which cause severe damage.

In 2003, Portugal suffered one of the worst fire seasons ever seen [3]. More than 400,000 ha. burnt an area making up 58% of total burnt area in the five most affected countries in Europe (Portugal, Spain, France, Italy, and Greece). The burnt area was almost four times the average recorded from 1980 to 2003. In addition, 21 people died due to forest fires over that season, and the estimated amount required to mitigate the damages was 1 billion

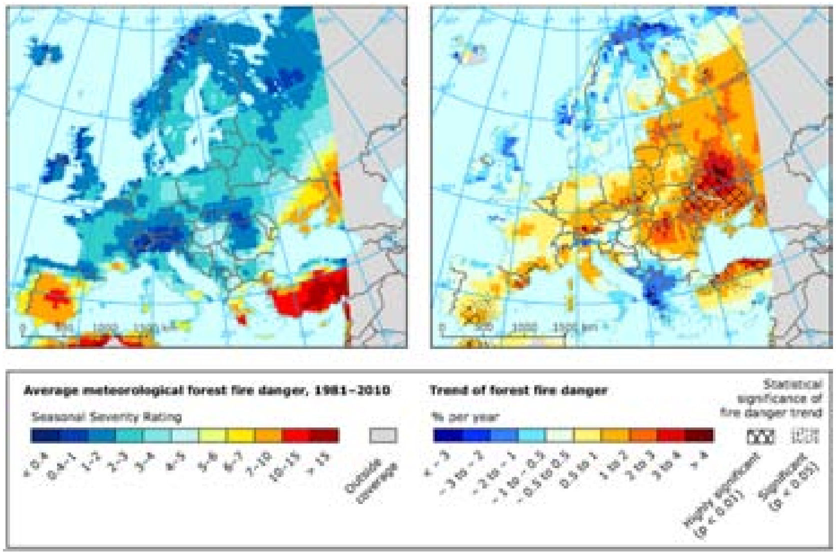


Figure 1.3: Average forest fire danger and trend

euros.

In 2007, Greece was also severely punished by wildfires [4], and 78 people died, most of them civilians (69). Approximately 225,000 ha. were burnt, comprising almost 40% of the burnt area of the Southern countries. These facts cannot hide the severe fire season in Italy that year. In that case, the worst since 1981, the amount of burnt area was similar, accounting for slightly over 40% of the five countries, and the number of victims was 23.

The state of California (US) also suffered a difficult fire season in 2007 [5]. More than 400,000 ha. were burnt by several big fires during that season. The biggest was the Zaca fire in Santa Barbara County that burnt almost 100,000 ha. and lasted 58 days. During that season, 17 people died, and that year became one of the worst fire seasons in California, along with the 2003 season.

Australia is a high-risk country due to the favorable conditions that wildfires find when they occur. These conditions led to one of the worst episodes ever experienced. It is remembered as the Black Saturday Bushfires, and it burnt around 450,000 ha., with over 400 fires detected. Temperatures reached 46 degrees Celsius, and windspeed topped 100 Km/h. In Fig.1.4, the map with the burnt areas from that disaster is shown. It is considered one of the top-ten worst wildfires in history, and 173 people died because of those fires.

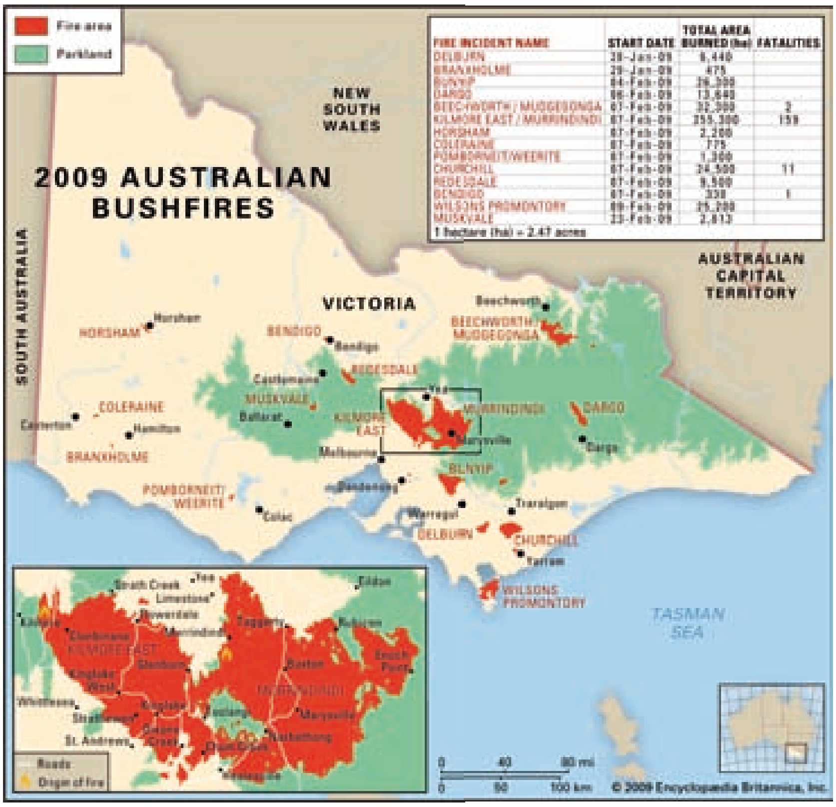


Figure 1.4: Black Saturday Bushfires map, Australia

As we have previously commented, Spain has followed a positive trend in forest fire burnt area. However, in spite of this, 2012 was the worst year since 1994. That year, more than 200,000 ha. were burnt, 10 people died and 30,000 were evacuated. The budget directly related to forest fire management (extinction and prevention) was 75 million euros.

Therefore, developing assessment strategies and tools to reduce the impact of forest fires is a common objective shared by institutions, scientists and society in general. From the point of view of scientists, the main objective in this field has been focused on developing models and simulators to predict fire behavior. Fire spreading is a multidisciplinary issue and requires the combined efforts of physicians, mathematicians, fire analysts and computer scientists, among others.

Fire spread behavior models are equation systems, based on physical and/or empiric knowledge that defines the dynamics of fire. These models are a simplification of the real system due to the high complexity and the amount of variables involved in these kinds of natural hazards. Because of this, one of the error sources when a fire simulation is performed comes from the incompleteness of the model.

This is only one of the many reasons. The input parameters required to feed the model suffer from a high degree of uncertainty and imprecision. Input data sources rely on measurement tools that usually have a certain error. Sometimes this error cannot be determined in advance, and we must accept it and deal with it. There are several factors that cause these errors, and they may start in the manufacturing step of the measurement tool. Its design and manufacturing determine its precision, and it is not exempt from imperfections. In addition, these tools are subject to wear over time, and this influences their accuracy. The environmental factors also influence the quality of the measurements, and, sometimes, the measurements may present a high error or be corrupted due to these factors.

Some input parameters can be generated by weather forecast services that rely on meteorological models, and, as is well known, these predictions may have a certain deviation from reality. Taking all of these considerations into account, developing strategies to minimize these errors seemed to be a promising way to improve forest fire prediction.

Based on the idea of minimizing the impact of the uncertainty of the input parameters, the two-stage prediction method that establishes a methodology to reduce the errors arising from the imprecision of the parameters was developed. This strategy relies on calibration techniques to find the set of parameters that best describes the fire behavior at a certain instant, using the knowledge of the spread of fire in a previous time interval.

The two-stage prediction method has some restrictions because of its original implementation and its working hypothesis. The latter states that the conditions should remain quite stable throughout the interval in which the calibration is performed, as well as the prediction interval. Therefore, this assumption restricts the possible fire scenarios where this methodology could provide successful predictions.

The main reason why this method is not sensitive to sudden changes is because of the use of a single value of each parameter to define the fire behavior over a whole interval. This is an advantage because it allows for a reduced set of parameters to calibrate, but it supposes a hard restriction in many scenarios. A similar situation occurs with the spatial distribution of the parameters. A single value of each parameter is calibrated for the whole terrain. This does not suppose a problem in small and flat terrains, but when we deal with real fires that take place in complex and rough surfaces, considering a unique value of certain parameters does not fit with the real behavior of such parameters.

## 1.2 Objectives

Taking the ideas previously exposed as a reference, our specific objectives are:

- Introducing complementary models to tackle the restrictions of the two-stage prediction method and extend the methodology in order to be applicable to real and complex forest fires with changing conditions. That means that these complementary models should solve the problems related to the temporal and spatial uniformity that the methodology presents.

- Once the models have been selected, they must be coupled in an efficient way in the two-stage prediction system. It is necessary to study the input and output parameters that take part in every model and implement interconnection modules that allow for communication between the system and these new models.

- It is also necessary to analyze the computational requirements of each model and to study how to introduce them without compromising the system s response time or ability to give a prediction in an acceptable time. To ensure an adequate response time, we must also define the requirements to incorporate real-time data in an operational system.

- Therefore, we want to develop a complete framework that allows for a fast response, using the available resources efficiently. We rely on HPC systems and parallel programming paradigms to guarantee these requirements.
- Finally, all these improvements must be validated using different fire scenarios and conditions.

Summarizing all these objectives, we want to provide a simulation and prediction system that aids in taking decisions during a real-time emergency. Obviously, this is an ambitious target, so we establish a methodology, prediction strategies, and a parallel framework. We have validated this work starting with prescribed fires, synthetic fires, and moving up to real scenarios. The reference fires are used to compare our simulations, analyze the error, and implement strategies to minimize this error.

Prescribed fires (see Fig. 1.5) are real fires evolved in a reduced and uniform terrain, where the conditions are monitored, and it is possible to study the spreading of the fire in detail. These fires are useful for validating fire spread models, studying vegetation features, and, in our case, comparing our strategies with these small study cases.

When we want to study bigger fires, with non-uniform terrains and conditions, and we do not have a real fire at our disposal, we can experiment with synthetic fires such as the one shown in Fig. 1.6. These fires are evolved using a fire spread simulator, and the features of the terrain and the conditions may be real or generated depending on our needs. This kind of fires are helpful when we want to reproduce certain terrain properties or conditions and study the fire behavior in those cases.

Finally, real fires (see Fig. 1.7) are the key scenarios where we want to provide our knowledge. In these simulations, we use the real perimeters of the fire provided by the fire analysts or the corresponding authorities.

These reference fires are compared to the simulations provided by our prediction strategies. The basis from which we start is the two-stage pre-

Figure 1.5: Prescribed burning

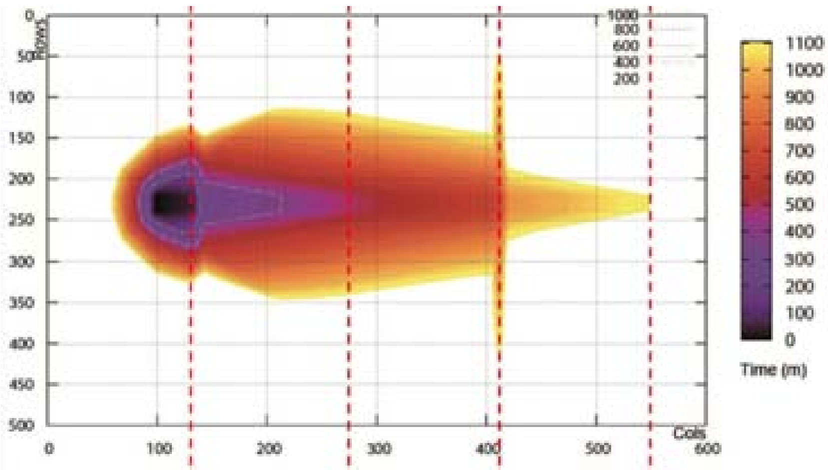


Figure 1.6: A reference fire evolved using a simulator

diction method, which will be explained more deeply in the next section. Briefly, this method performs a calibration stage that tries to tune the input parameters to be closer to the real fire behavior. It requires a previous real fire front to adjust the parameters and then provide a simulation.

We have focused our efforts on improving the predictions of the two-stage method and coupling complementary models with our system in order to increase the accuracy of the input parameters. How to improve the most sensitive parameters behavior has been studied, both from a spatial point of

Figure 1.7: An example of a real fire scenario

view as well as temporally.

It is well known that when studying large forest fires that burn for long intervals, meteorological conditions significantly affect fire behavior. These conditions cannot be considered uniform along the terrain and constant over time due to the high complexity involved in big fires. Up to now, considering these variables uniform and constant was possible because the homogeneity present in prescribed fires. When we deal with large scale fires, this assumption becomes a hard restriction, and new strategies must be applied.

The first restriction is related to the spatial distribution of certain parameters. We highlight the case of wind components due to their high significance in fire propagation. In this case, assuming a single wind direction and speed for the whole terrain is not a realistic situation. Wind suffers changes because of terrain irregularities that must be taken into account when we model a fire over a complex surface.

The second restriction is directly related to the temporal evolution of the variables over time. The most relevant cases are the weather variables. When we study the evolution of a fire during long intervals, considering the first value of each parameter as a constant for the whole interval generates considerable prediction errors due to the large variability present in these types of parameters.

Therefore, we propose adding a spatial wind field modeler to simulate the effect of wind - one of the most influential parameters - over a certain terrain. Additionally, we suggest increasing the number of meteorological variables samples over time, using the information provided by a meteorological forecast model.

Finally, this work proposes a comprehensive methodology to simulate and predict forest fires, beginning with the data acquisition and processing, the models and simulators used, the prediction schemes proposed, and lastly, the output analysis. This methodology also establishes a parallel solving scheme, using programming paradigms such as MPI or OpenMP, in order to speed up the response of the system.

## 1.3 Organization

This work is organized as follows:

Chapter 1 introduces the research topic, the motivation of this work and the main goals that will be tackled.

Chapter 2 is intended to analyze the background of this work. In our case, the most relevant research work in this field is presented, analyzing the existing fire spread models and the simulators and tools used by the researchers and fire analysts. At the end, we present the previous work done in our research group that has culminated in this thesis.

Chapter 3 explains the complementary models, how to integrate them in the two-stage prediction method, the computational requirements and depicts the parallel system proposed, its inputs and outputs, and the programming paradigms used.

Chapter 4 presents the methodology that we propose to treat this issue. The data acquisition and processing, the topographic characterization of a region and the modules of the system are detailed.

Chapter 5 shows the experimental work that validates our proposal. It is organized chronologically, from the earlier experiments using small synthetic fires to the latest experiments using real fires.

Finally, Chapter 6 points out the conclusions extracted from this work and the open lines that must be discussed and kept in mind in the future.

# Chapter 2

# Forest fire spread prediction

## 2.1 Models, simulators and frameworks

Forest fires and other natural hazards are phenomena widely studied by the scientific community due to the huge environmental, economic and social impact that they produce every year around the world.

In the case of forest fires, many preventive actions can be applied to reduce the forest fire risk, but unfortunately, they do not completely remove the multiple factors unleashed by a fire. Because of this, our effort is focused on mitigating the consequences when the fire is detected.

To study forest fires, many fire spread models based on physical laws and empiric observations have been developed in order to predict the behavior of forest fires [6] as closely as possible. Fire models can be divided into three categories:

- Physical models: Theoretical models based on physical laws of heat transference and fluid mechanics. They are represented by mathematical equations [7][8][9].

- Empiric models: This kind of model is based on experimentation and probability factors. It is tightly linked to the zone where the data has been collected, so it is hardly portable [10].

- Semi-physic models: It comes from mixing both ideas, a hybrid model based on physical laws and introducing empirical observations [11][12].

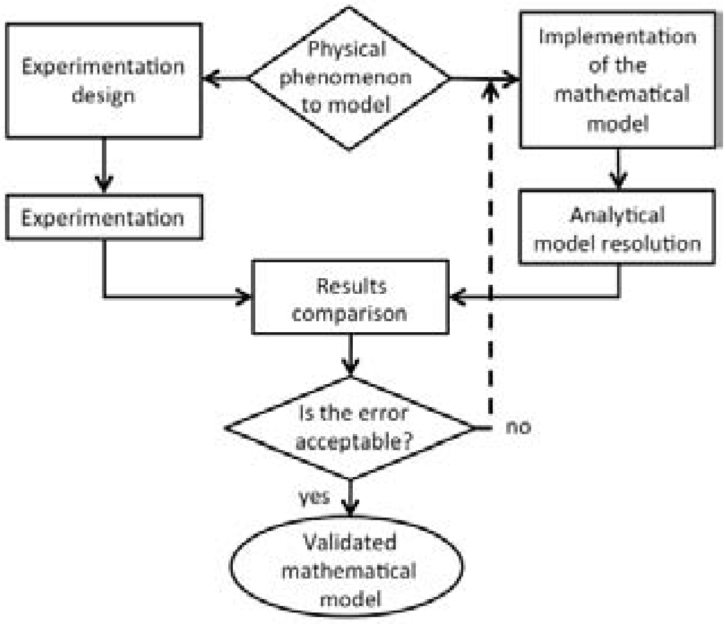


Figure 2.1: Semi-physical model process creation.

### 2.1.1 Fire spread models

Besides the nature of their equations, fire models can be classified by the fire type that we want to study. There are four main types of fires, and each one has different behavior. Therefore, the physical systems and equations vary from one to another. In Pastor et al. work [13], an extensive list of forest fire models is analyzed and classified according to their typology. Basically, we can find four types: surface fires, crown fires, spotting fires and ground fires.

Surface fire models focus their study on the fires that burn the vegetation near the surface, such as small trees, brush, or herbaceous plants. Crown fire models are somewhat complementary to surface fire models and study how the fire spreads over the canopy of trees in a forest. Spotting fire models provide the equations to analyze those new fires caused by incandescent pieces of the main fire transported out of the main fire perimeter. Finally, ground fire models concentrate their attention on those physical processes that occur in the substrate of the soil when a fire takes place. Usually, fire prediction tools use a combination of a surface fire model, a crown fire model, and a spotting fire model inside the simulation core.

The first attempt to model surface fire was done by Fons [14], who proposed the first theoretical fire spread model. McArthur [10] developed an experimental work that defined grassland fire behavior tables. These tables have helped to calculate fire risk indexes in herbaceous terrains, predict the rate of spread in these terrains, and implement a public alarm system in Australia. Anderson [15] proposed a semi-physical model that described the heat transfer processes in a fire analytically, using experimental data. This model has the limitation that it assumes a scenario without wind and analyzes how certain parameters (fuel variables, terrain features and some atmospheric variables) influence the fire spread rate. The most common shape for approximating how a fire spreads is the ellipse. Van Wagner [16] proposed an analytical model based on the mathematical equation that defines the ellipse (see Fig.2.2). The size of the ellipse (a and b segments) depends on the rates of spread of the head, the flanks and the rear. These rates will be determined by the wind components and other fire behavior parameters, and the area of the ellipse is a function of time, so it will increase at every step, maintaining the elliptical properties.

Later, Rothermel [17] presented the most widely used fire spread model, based on a mathematical basis and empiric observations. This model will be explained in detail in subsequent sections. Rothermel also defines 11 fuel models and their parameters required as input to the model. Albini later refined these fuel models and added two more. These 13 standard fuel models were described by Anderson [18] and advice about how to select a model was given. Anderson et al. [11] proposed an alternative mathematical model for grasslands based on Huygen s principle that was intended to predict fire perimeters. Huygen s principle is also used in Rothermel s model and is depicted in Figure 2.3. Considering an initial fire front $t_0$, the future fire front $t_1$ will be equal to the curve that envelops the elliptic spreads, resulting in the application of the fire spread model at different points of the initial perimeter.

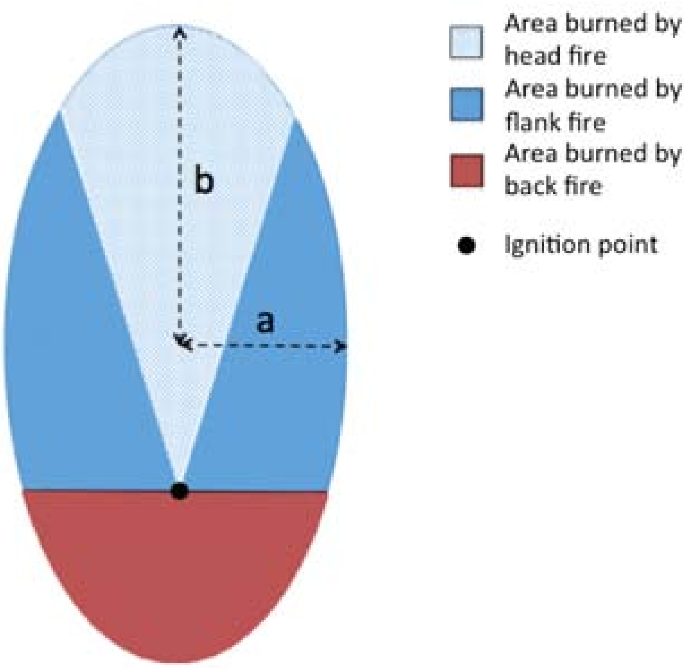


Figure 2.2: Elliptical fire spread scheme.

Other later models tried to improve fire spread modeling by including more complex physical interactions and profiting from the continuous improvement in computational resources [7][19][20].

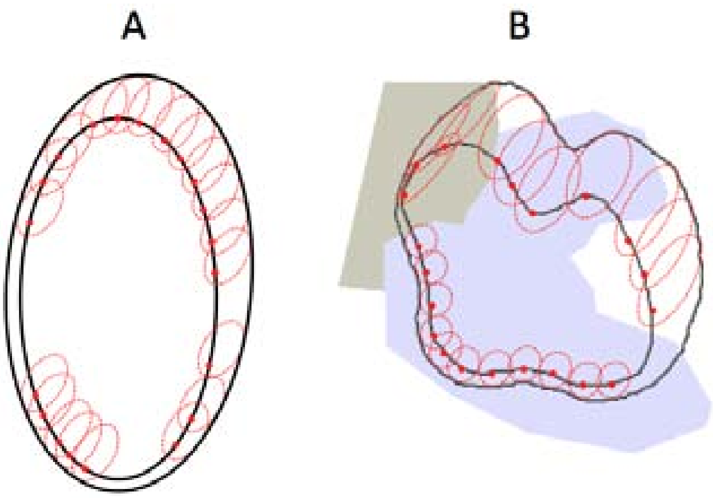


Figure 2.3: Huygen s principle.

There are several models that study how crown fires behave. Some of them analyze the initial conditions and the factors required for a crown fire [21] and how a surface fire can become a crown fire. Other models [22][23] propose equations characterizing the spread of these fires. Later works presented models that include both initiation and spread models [24][25]. In all of these cases, it is important to note that each model is intended to study a specific zone. Therefore, the models have intrinsic errors due to the difficulty of extrapolating them to other scenarios.

In the case of spotting fires, the most relevant and prolific author was Albini [26][27][28]. According to him, the most important factor to know is the maximum spot fire distance. If we can accurately predict this value, it will be easier to place emergency services and firefighters correctly.

### Rothermel s model

Rothermel s fire spread model [17] is one of the most widely used models for the study, modeling and prediction of forest fires. It can be considered a semi-physical model because it is based on mathematical equations that represent physical processes and relies on certain factors that have been inferred experimentally. This is a surface fire model that defines the equations that calculate the rate of spread and the intensity of a single point of the fire. The underlying physical principle is the conservation of energy applied to a unit volume of fuel ahead of a spreading fire in a homogeneous fuel bed. Equation (2.1) represents the rate of spread in Rothermel s model, and it is the ratio between the heat generated by a unit volume of fuel and the heat required to bring a unit volume of fuel to ignition temperature. A unit volume of fuel (also seen in the literature as fuel unit, unit fuel cell, or unit cell) is the minimum unit of fuel used in these models instead of the particle.

$$R = \frac{heat\ source}{heat\ sink} = \frac{I_r \xi \left(1 + \Phi_w + \Phi_s\right)}{\rho_b \varepsilon Q_{ij}} \; (ft/min) \tag{2.1}$$

where:

$I_r$ = Reaction intensity

$\xi$ = Propagating flux ratio

$\Phi_w$ = Wind coefficient
$\Phi_s$ = Slope factor
$\rho_b$ = Fuelbed bulk density
$\varepsilon$ = Effective heating number
$Q_{ij}$ = Heat of preignition

The reaction intensity is the energy released per fuel unit while a fire is burning. It depends on fuel characteristics such as particle size, bulk density, heat content, moisture of the fuel and chemical composition. These parameters will influence how the fire spreads, i.e. when the moisture of the fuel is high, the fire spreads slower than in the opposite case, and when the heat content or the density is high, the reaction intensity grows.

The propagating flux ratio defines how much of the reaction intensity is forwarded to the atmosphere and how much the rate of spread value is directly affected.

Wind coefficient and slope factor are empiric values that represent the added effect of wind and slope over the rate of spread.

The factor corresponding to fuelbed bulk density is different for every vegetation type (or fuel model). This parameter determines the amount of fuel mass that a given fuel unit contains.

The effective heating number is a value from 0 to 1, which represents the percentage of fuel load that needs to be heated to ignition for each vegetation type. When we deal with vegetation which is predominantly grass, this value is close to 1, because it needs to be heated to ignition completely. However, trees or big branches have a lesser effective heating number, because only a portion of their mass must be heated to ignition.

The last element of the equation corresponds with the heat of pre-ignition. It is the energy required per fuel unit for ignition and depends on the ignition temperature, the moisture and the load of the fuel type.

### 2.1.2 Fire simulators and tools

Many simulators have been developed using a combination of a surface, crown and spot models as underlying fire models. We can find simulators and tools such as BEHAVE, FireLib, BehavePlus, FlamMap, FSPro, FSIM, FMAPlus, NEXUS, FireStation, or Farsite. Most of them belong to the big family of Rothermel s model-based tools. Out of all of these applications, Farsite is the most used fire growth simulator. Some of them have different purposes, complementary or different to Farsite. Let us analyze some features of these tools:

BEHAVE [29] is the simulator on which the rest of fire growth simulators hang. It relies on Rothermel s surface model, Van Wagner s initiation crown model, Rothermel s spread crown model and Albini s spotting fire model. The other tools use the same models but they are different in how they treat space and time. Behave, and its successor BehavePlus [30], assume that initial conditions remain stable and uniform both in space and time.

FireLib [31] is a simple fire library for predicting the spread rate and the intensity of fires, directly derived from BEHAVE. It uses a cellular automation technique to simulate the spread. The most common types of simulators, depending on the way they calculate the spread, are those based on cellular automation and those that use an elliptical wave propagation [32]. The first ones are based on cell maps representing the landscape, and every cell has a parameter set and a state. When an ignition is placed, the state transition will be determined by a set of common rules that will affect the ignition cells and their neighbors. In elliptical wave propagation simulators the landscape is seen as a continuous surface (although its real representation is discrete). The calculation of the next perimeter is done by choosing certain points of the initial perimeter, applying the spread equations at every point and building the envelope curve (Huygen s principle).

FlamMap [33] has many similarities to Farsite, but eliminates the temporal evolution. FlamMap is intended to study fire behavior under landscape features (terrain and fuels) at a certain instant. For this reason, it is interesting to understand the fire behavior and changing fuel parameters. All the fire spread equations are applied independently to each cell on the landscape, and the simulator offers a wide output information (fire line intensity, flame length, rate of spread, minimum travel time paths, etc.).

FSPro [34] follows a statistical approach and builds burning probability maps for a given time period. It uses historical information about weather and fires. It evaluates several weather scenarios to determine the fire risk for each cell of the terrain. Unlike other fire tools, FSPro is not oriented to desktop computers, but it requires more computational power to evaluate all the scenarios and perform the statistical analysis.

FSIM [35] is a tool to determine the burn probability over large areas (millions of hectares). It includes a fire growth model, a weather generation module and fire occurrence and suppression systems. It can simulate the fire growth for large periods (in the order of years) to generate burn probabilities that can help in mid- and long-term fire-fighting planning or ecological research.

FMAPlus [36] or Fuels Management Analyst Plus (FMA+) is not a fire spread simulator, but a set of tools designed to store and estimate some terrain features such as surface and canopy fuel loading.

Another fire analyst tool is NEXUS [37], a system that couples a surface with a crown model to assess the potential for crown fires at the stand level and allows for the visualization of surface and crown model interactions.

FireStation [38] is a simulator based on the cellular automation technique that also includes the fire growth models of the BEHAVE system. One of the major contributions of this simulator is the fact that it includes a wind field modeler. This task is relayed in two wind models, NUATMOS and CANYON, and wind data coming from weather stations near the hazard focus.

Farsite [39] is a fire behavior and fire growth simulator that incorporates both spatial and temporal information on topography, fuels and weather. Unlike BehavePlus, it includes temporal variation in fire conditions. Farsite is an elliptical wave propagation simulator and avoids a typical problem of cell based simulators of reproducing the fire shape in heterogeneous conditions, due to their reduced number of propagation paths.

## 2.2 Forest Fire simulation

In this section, we will describe the input and output data that take part in the forest fire simulation process and some prediction methods that try to reduce the prediction errors. The classical prediction will be described and compared with the two-stage prediction method. We will also show some approaches that follow the two-stage methodology and have been the basis of this work.

### 2.2.1 Input and output data

Although it is the main step, fire simulation is the last phase of the process. Once we have collected all the information about the fire, the simulation can take place. All the input data has to be adapted to the fire simulator. The static and dynamic data come from several sources, with specific formats and different resolutions. These heterogeneous data must be unified and standardized to be useful for the fire simulation system.

Most fire behavior simulators require GIS input layers to provide fire spread forecast as reliably as possible. Although the sensitivity of the model to the input data clearly depends on the nature of each required parameter ([40]), the precision and quality of all of them are not dismissible. In general, the data needed to perform the predictions can be divided into two main groups: *static* and *dynamic* data. The static input data is the one that remains constant throughout the whole prediction interval and the dynamic data experiment changes during the fire spread simulation.

### 2.2.2 Static input data

As has been commented above, static data do not vary during the simulation, so they can be collected and processed once and are then available for all the forest fires that take place in the same topographic area. In particular, those features that can be considered fixed in a given area are: topographic information and vegetation maps.

### Topographic information

The three input layers required to properly describe the forest topographic area where the fire occurs are the so-called *elevation*, *aspect* and *slope* maps. The *elevation* or DEM (Digital Elevation Map) is a simple, regularly spaced grid of elevation points which provides a discretization of a continuous surface, taking into account measurements at certain points of the terrain (see Fig. 2.4). The maximum resolution of the imaging resources defines how many measurements we can achieve in a certain area.

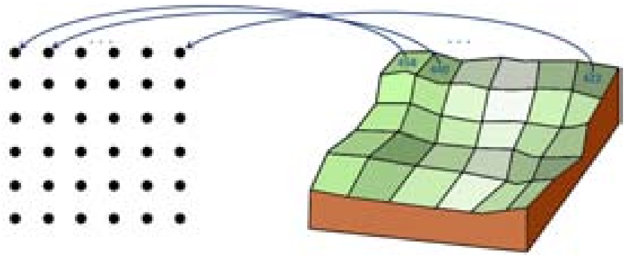

Figure 2.4: Elevation sampling process

Slope and aspect can be calculated using the elevation ([41][42]) and applying certain equations. In DEM files based on regular square cells, slope and aspect of each cell is determined by the altitude of some specific neighbors, depending on the method chosen (see Figure 2.5). These parameters determine the roughness of the terrain and have a direct influence on fire spreading.

### Vegetation maps

The vegetation map, also known as the fuel map, shows the vegetation diversity of the area. Each vegetation or fuel type has its own parameters such as moisture content, flammability, fuelbed load and density, heat content, etc. These kinds of maps consider a limited group of standard fuels that represent the great diversity of possible fuels and parameters.

In fire spread prediction, every one of the standard fuels present on a map has a list of parameters about the content moisture of that fuel. Fuel moistures are those parameters that define the content of water of live and

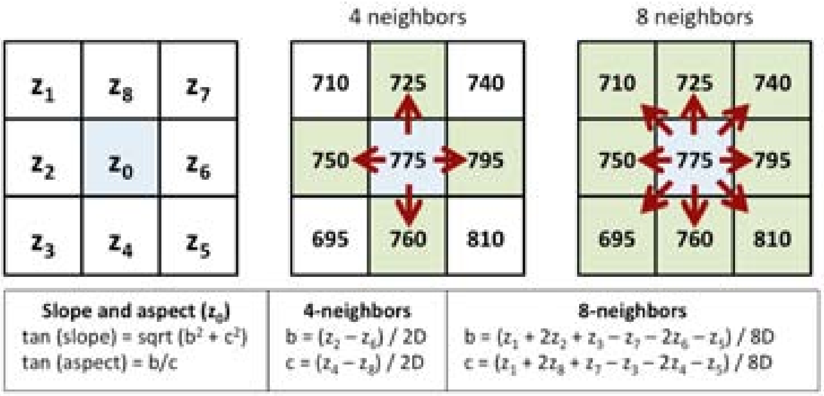


Figure 2.5: Slope and aspect calculation equations

dead fuel. The concept of live fuel is related to the live vegetation of the area (trees, bushes, grass, etc.). Dead fuel is the vegetation that remains lying in the litter of the forest floor. These moistures have a direct impact on the fire spread rate.

There is another kind of map concerning vegetation called the canopy cover map. This map shows the percentage of tree crowns present in each terrain division. This information will affect the fire typology, since it does not spread in the same way as a surface fire or a crown fire.

### 2.2.3 Dynamic input data

Dynamic data is probably the most influencing input and its accuracy is essential in producing a good prediction. We define two main groups: GIS fire perimeters and meteorological data.

#### GIS fire perimeters

The first group are those images that allow us to fit the forest fire simulator with an initial fire perimeter and, furthermore, to observe the evolution of the fire and to compare our predictions with the real fire behavior. These perimeters are really difficult to obtain and their precision depends on satellite image quality, which will be affected by fire and weather conditions. It is worth mentioning that the image resolution also has a great impact on the results.

In time terms, fire perimeters are possibly the hardest data to obtain. In our experience, the maximum fire perimeter frequency is up to two per day. This value corresponds to the number of satellite images that we can achieve of a certain region. But, in some cases, these images are not clear and certain factors (clouds, smoke, etc.) make it impossible to interpret the fire perimeter. Therefore, it is difficult to ensure when a perimeter will be ready, and sometimes it is not possible to compare our predictions with the real fire behavior.

### Meteorological variables

The meteorological variables are a key factor in a real-time forest fire spread forecast. These parameters have a direct impact on fire spread direction and intensity. We only take a representative subset of these variables, those which have more relevance in these kinds of hazards.

Temperature and humidity have an influence on fire intensity, but the most important variable is wind. Wind direction usually determines the direction of maximum spread in a forest fire. It also depends on vegetation and terrain features (fire spreads more over an upslope than a downslope), but if wind speed is high, fire tends to spread in the same direction.

Fortunately, weather variables are easier to obtain than fire perimeters. In this case, meteorological services usually perform short-term and mid-term daily predictions. Obtaining this data is almost immediate, but it must be processed, and passed many times through a complementary model to increase the resolution. In addition, we are interested in the latest prediction, because the accuracy of these models depends on the time-window selected. Therefore, we look for a trade-off between data accuracy (newest predictions) and data processing time.

### Output data

Each fire simulator provides its own information about the fire spread. The rate of spread, the time of arrival of fire, the intensity, the perimeters in vector files and other fire parameters are delivered by the simulators and give detailed information about the forest fire.

## 2.3 Forest Fire prediction

Below, we briefly present the classical way of predicting forest fires (and other phenomena). We also describe the two-stage prediction methodology and the different approaches that have been developed.

### 2.3.1 Classical prediction

The classical way of predicting forest fires relies on a single execution of the fire simulator and is depicted in Fig. 2.6. In this prediction scheme, the input parameters that define the fire evolution in an initial instant are collected, along with the ignition point or perimeter, as well as simulation parameters, such as the time to be simulated and other parameters specific to each simulator.

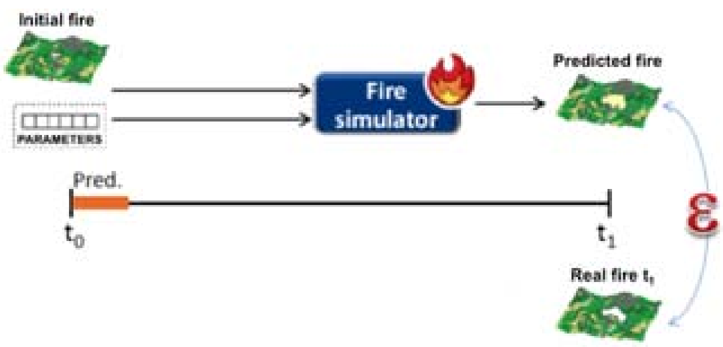


Figure 2.6: Classical prediction scheme.

Once we have defined the environmental and simulation features, we carry out the simulation, and the system returns a prediction for the next time instant. The output will depend on the simulator chosen and it can give back a time-to-arrival map with the instant time when the fire burns a cell, a shape file with the output perimeter, a file indicating the rate of spread at every perimeter point, the maximum spread direction and any information related to the future fire behavior.

### 2.3.2 Uncertainty and imprecision of input parameters

The major problems of classical prediction method are probably the uncertainty and imprecision that the input parameters of a complex physical system as a forest fire, or any natural hazard, present. This imprecision generates errors in the input parameters that are generated and propagated from the data sources to the fire simulator. The data sources and the common issues that generate this error are shown in Fig. 2.7.

Given this situation, we can address the problem from two opposing positions. We can assume this intrinsic error that generates this uncertainty and suppose that it will not influence prediction errors too much, or we can face the problem and develop strategies to reduce this lack of knowledge or minimize the error due to the uncertainty of the parameters.

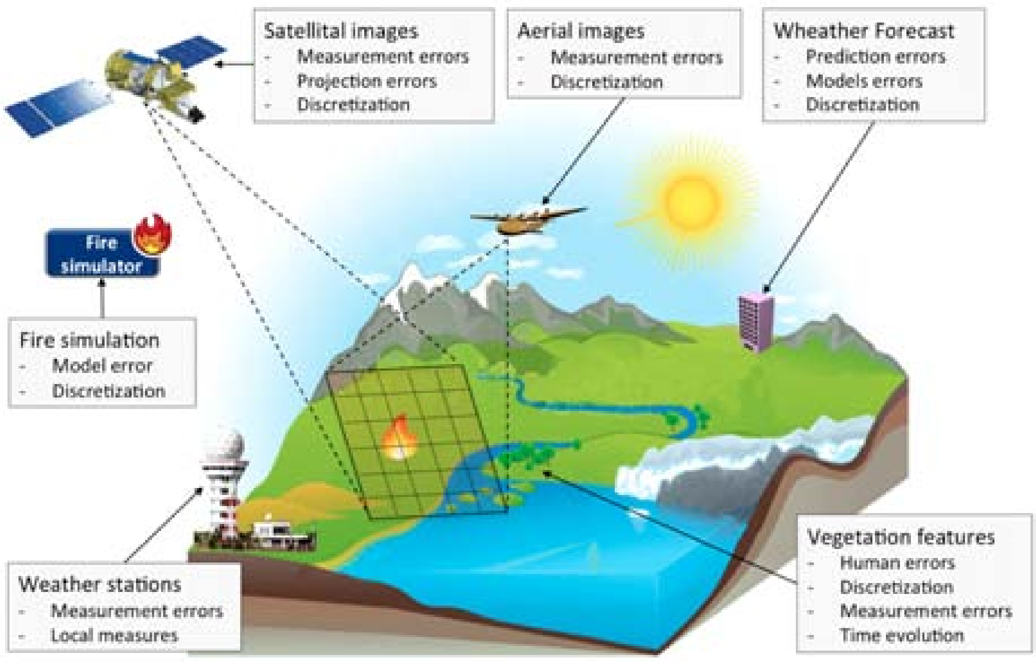


Figure 2.7: Error sources in forest fire input parameters.

In our research group, a prediction method was developed that attempts to deal with this problem using parameter calibration techniques, which allow for the minimization of errors under certain conditions. Besides reducing the uncertainty, this method also slightly minimizes the fire model error itself.

### 2.3.3 Two-stage prediction method

Two-stage prediction method is so named because of the inclusion of a new stage called the calibration stage before performing the prediction. Following the scheme shown in Fig. 2.8, we are going to explain this methodology step by step.

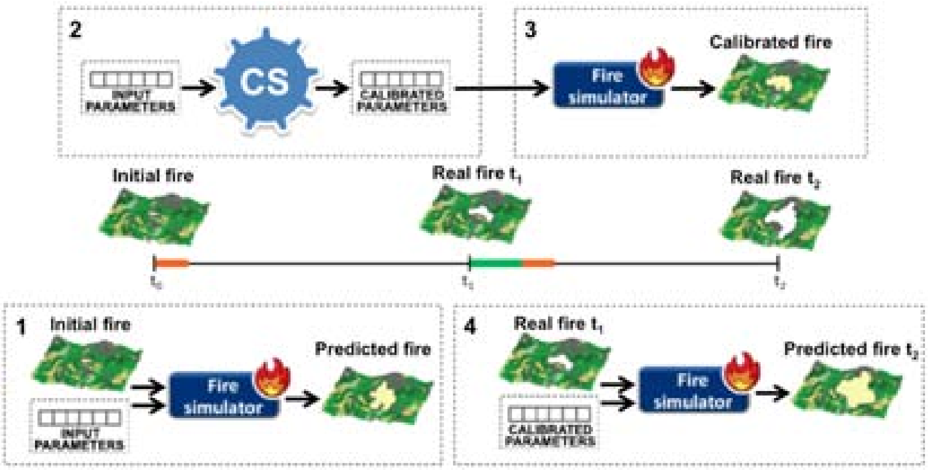


Figure 2.8: Two-stage prediction scheme.

A first interval ($t_0 - t_1$) is required to train the method. In this process (step 1 in the figure), several single predictions are carried out and the result is compared with the real fire behavior at the time instant $t_1$ ($RFt_1$). This provides valuable knowledge about the fire to the system.

Once we have the comparison between the real perimeter and the training set, the parameters are sent to the calibration module (step 2 in the figure). This module applies optimization techniques to achieve an improved set of parameters. This is an iterative process that tries to minimize the error produced by the input parameters. The process will end when the error reaches an acceptable value or after a fixed number of iterations, when we obtain the parameters that best reproduce the recent behavior of the fire (step 3 in the figure).

This optimized parameter set along with the real perimeter at $t_1$ will serve as input to the fire simulator to perform the prediction for the next time instant $t_2$. Under certain restrictions, these parameters should produce a better prediction than the non-optimized ones (step 4). These restrictions are summarized in the working hypothesis of this method, which states that the conditions under which the fire evolves do not vary drastically between calibration and prediction stages. Without the assurance of this assumption, we could not ensure the success of this methodology.

#### Evolutionary method

The first approach of the two-stage was using an evolutionary method based on a genetic algorithm. Genetic Algorithms [43] allows for multiple-problem resolution using the idea of how species evolve in the natural world. The two main ideas are natural selection and sexual reproduction, with the first being the basis of survival and offspring of individuals, and the second the fact that allows for the combination of the individuals that survive in the population. Therefore, selection ensures that the individuals best adapted to the environment will survive. In addition, these individuals will combine their genetic features to generate new individuals that will cause the population to evolve generation after generation. In this process, some individuals can mutate some genes so that the resulting individuals are not exactly the result of crossing their predecessors.

Transferring this concept to the computational world, a population can be seen as a set of possible solutions to a problem. Each solution, in genetic jargon, is an individual. To evaluate these individuals, an error or fitness function is necessary that calculates the goodness of an individual. Once we have ranked all the individuals or solutions, we can perform a series of genetic operations to generate a new optimized population. First, some of the best individuals (a percentage or a fixed number) are selected, and they remain unchanged for the next generation (elitism). Then, a subset of the population is combined in pairs to generate new individuals (crossover). And, finally, a percentage of the population (usually low) modify their genetic features or genes to introduce more heterogeneity in the population (mutation).

But how is this idea applied to forest fire parameters calibration? The calibration method implemented was called the evolutionary method and based on a genetic algorithm [44] where we start with a random population where every individual is a set of parameters that describes the status of the fire at a certain instant. These individuals are introduced into the fire simulator with the initial fire front ($RFt_0$), and a predicted fire front is obtained for every individual for the next time instant $t_1$ (see Fig. 2.9).

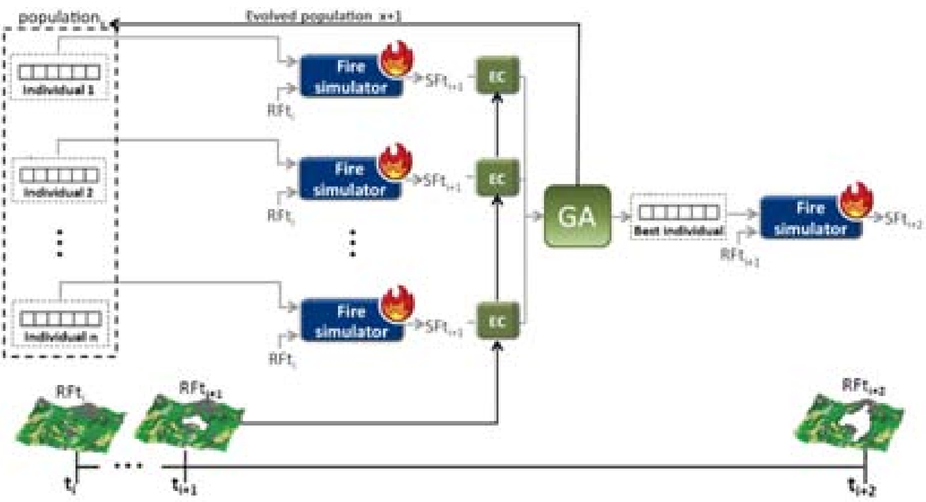


Figure 2.9: Two-stage prediction scheme based on Genetic Algorithm.

These fronts are compared with the real one at $t_1$ ($RFt_1$) and the difference or error of each individual (EC module, Error Calculation) is calculated. The population is ordered according to this error and the genetic operations (elitism, crossover, and mutation) are applied to generate a new population.

The process is repeated for a fixed number of iterations (generations), at which point the individual with the best parameter set is selected and is responsible for performing the prediction for the instant $t_2$.

The most important parameters of the algorithm are the number of individuals, the number of generations, the elitism rate, the crossover rate and the mutation rate. It is easy to implement and it is a powerful parameter optimization technique that usually does not require a large number of generations to reach a good solution. Despite this, it may happen that the algorithm does not converge and never ensures an optimal solution, so we must properly configure the parameters to obtain the expected results.

#### Guided evolutionary method

To improve the response of the method to the changes in meteorological conditions, some techniques to add knowledge to the system are developed. The objective is to identify these changes in advance and to not always be a step behind the fire.

The idea is simple but useful. Rothermel s model calculates spread direction depending on wind components and slope. This method [45] intends to go the opposite way and proposes achieving the wind components that best fit the actual fire behavior from the slope and the spread direction observed. This relationship is obtained using a knowledge database where every entry has a slope, spread speed, spread direction, vegetation model, and its corresponding wind components (direction and speed).

This added knowledge about wind will be applied to those individuals that take part in the mutation process, delimiting the variation range of wind speed and direction genes. To reduce the wind range (direction 0-360 degrees and speed 0-30 mph), the value found is placed in the center of the range of variation and a confidence interval for each component is defined. Thus, individuals chosen to mutate will do so in a reduced range.

Using this method, the system usually converges faster and we improve the prediction quality in most cases. In addition, the data-driven evolutionary method performs better under changing meteorological conditions. It is noteworthy that the prediction quality depends on the completeness of the knowledge database.

#### Statistic method

The statistic or probabilistic method [46] pursues the same objective of the previous methods, but differs a lot in how it reaches the target of calibrating the parameters. To better understand the method, we will support the explanation in Fig. 2.10.

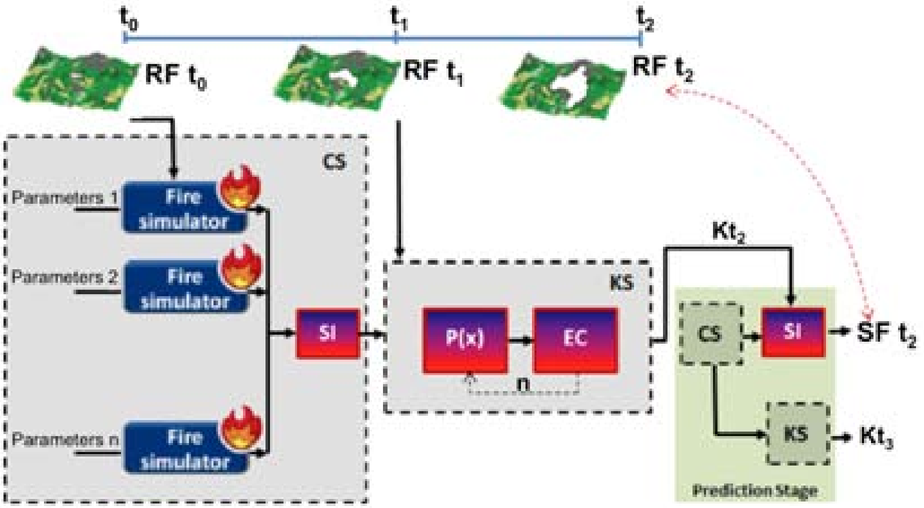


Figure 2.10: Two-stage prediction scheme based on Statistical method.

The first consideration is that it does not start from a limited number of parameter sets, but it performs a combinatorial explosion from which we obtain all the possible parameter combinations. To achieve this goal, defining a minimum and maximum bound for all the parameters is mandatory, based on the observation and study of each parameter. Therefore, the population is much greater than in the evolutionary approach, because we cover all the possible combinations and eliminate the possibility of having poor initial populations at the cost of exploring the entire search space.

All the predicted fire fronts are introduced in the Statistics Integration module (SI). This module generates a map where the number of times that a cell is burned is accumulated, sweeping the predicted maps. All the cell values of the accumulated map are divided by the total number of scenarios, which will produce a burning probability map with values within the range [0-1] (see example in Fig. 2.11).

Once the probability map is built, it begins the search for the factor $k_{ign}$ that corresponds to the probability that best describes the real fire behavior at $t_1$ (see example in Fig. 2.12). For this reason, this step receives the real

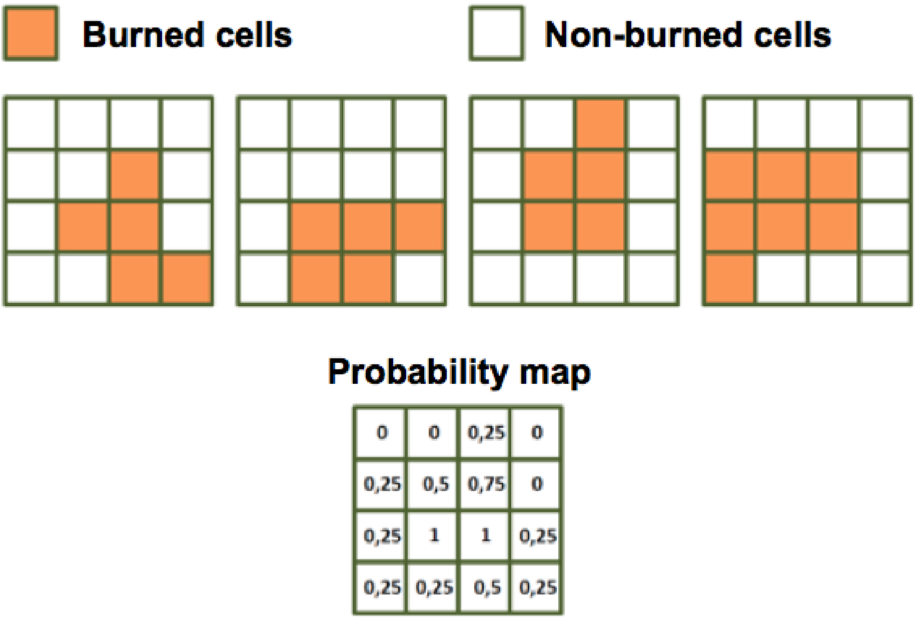


Figure 2.11: Probability map for four scenarios.

front at that time ($RFt_1$) that is used to compare with the maps corresponding to each probability value.

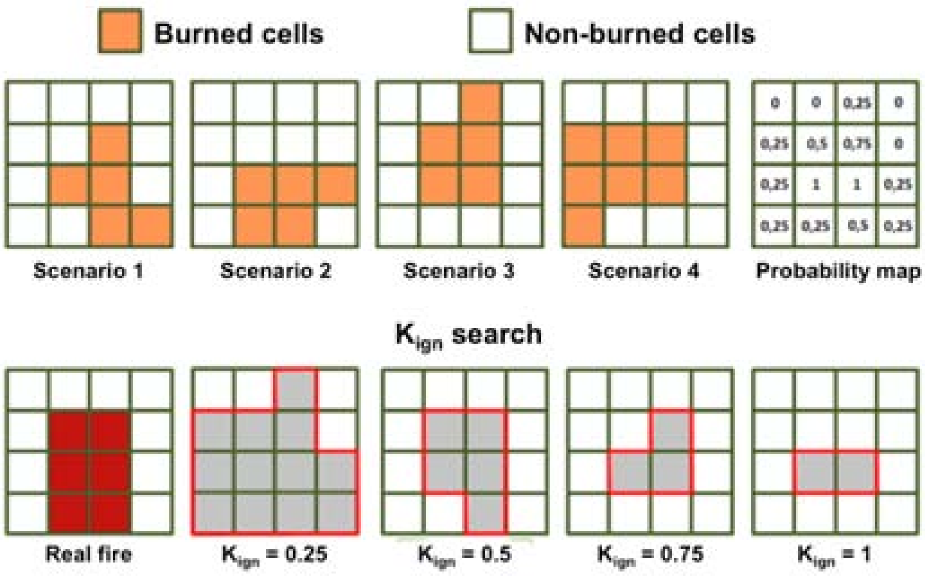


Figure 2.12: Kign factor search process.

The probability value that best matches the real fire is then assigned to the factor $k_{ign}$ ($Kt_2$ in Fig. 2.10) and the system will be able to give a prediction for the time instant $t_2$. At this moment, the search for the factor for the subsequent time $t_3$ ($Kt_3$) can be initiated. In this way, the predictions will overlap.

The main benefit of this method is also its main weakness. As has been commented, combining all the parameters themselves means that we cover all the potential solutions. In contrast, this method requires a huge computing power to simulate all the possible scenarios. Nevertheless, and in comparison with the evolutionary method, the statistic method adapts better and faster to sudden changes in meteorological conditions. Obviously, this method is highly parallelizable and the workload can be distributed among the available resources.

#### Hybrid statistic-evolutionary method ($SAPIFE_3$)

After seeing the strengths and weaknesses of the previous strategies, a new hybrid method is developed that combines some benefits of both and reduces their main problems. The method is called $SAPIFE_3$ [47] (Adaptive System for Forest Fire Prediction based on Statistical-Evolutionary Strategies, translated from Spanish). The main idea is using the genetic algorithm not only to generate a single solution (or individual), but also a set of possible scenarios that will be processed by the statistic method (Fig. 2.13). This significantly reduces the computational requirements of the statistic algorithm due to the reduction of the search space. This could violate the nature of the statistic method based on the exhaustive search, but it has been demonstrated that using a reduced (but optimized) set of parameters does not adversely affect the method and significantly improves the prediction times.

It was necessary to modify the genetic algorithm in order to implement this method. Instead of returning the best individual, it will deliver a population as input to the statistic modules of the system. These modules no longer perform a combinatorial explosion covering all the parameters configurations but only focus the effort on the reduced population optimized by

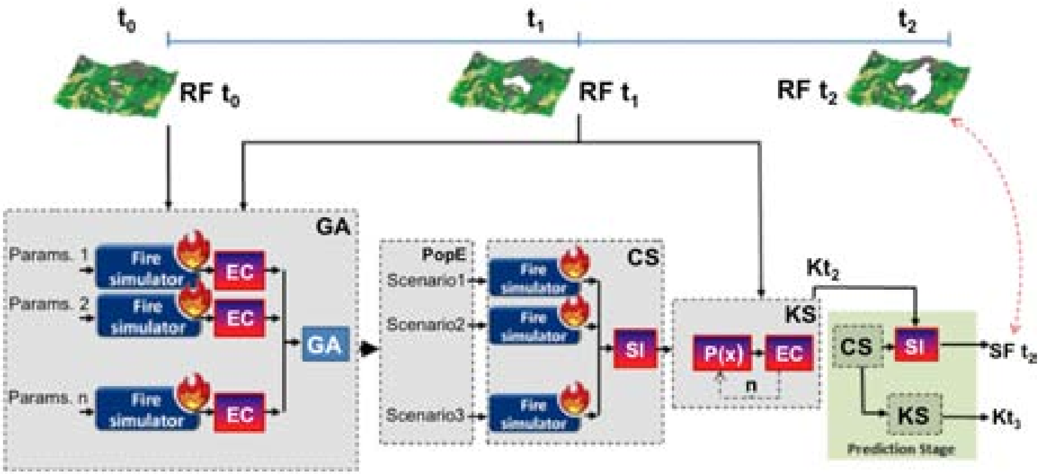


Figure 2.13: Two-stage prediction scheme based on hybrid statistical-evolutionary method.

the genetic algorithm.

The statistic part builds the probability map, searches for the $k_{ign}$ factor and returns the predicted map for the next time instant, using the same methodology previously described.

The benefits of this method are the result of coupling the two previous methods and taking advantage of the fast and efficient parameter optimization that the evolutionary method offers along with the better adaptation to changes in weather conditions of the statistic method. The results of this work show that when weather conditions remain quite stable, the errors are close to those that are achieved by the simple evolutionary method. When a change in conditions between stages occurs, the evolutionary method generates high prediction errors compared with the statistic, and the hybrid method behaves similarly to the statistic. In addition, it meets its objective in a much shorter time. These results are based on an experiment where a synthetic fire is simulated. The weather conditions are fixed during the calibration stage from 0 to 8 minutes (wind of 5 mph and 180 degrees) and changed at this moment to 10 mph and 270 degrees. As has been commented, $SAPIFE_3$ performs similarly to the evolutionary method during the first 8 minutes. When conditions change the evolutionary method increases in error while the statistical and the hybrid cushion the error and achieve better predictions.

### $SAPIFE_3rt$ method

The $SAPIFE_3$ method was improved and renamed $SAPIFE_3rt$ (real time). The objective of this change was using real data sources (meteorological stations, sensors, etc.) in order to detect sudden changes in the environment and to act accordingly.

To achieve this goal, a new module is added which is responsible for acquiring data, detecting changes in conditions and replacing a certain number of individuals that take part in the genetic algorithm. This work establishes a methodology for dealing with injected data and suggests the creation of an injection policy, because injecting all the real data is not always recommended. Therefore, the pending work is defining a policy of how and when to inject real-data during a simulation. It is also important to define when a change is abrupt enough because first experiments hypothesize that, in those cases, it is highly recommended.

## 2.4 Evaluation of quality prediction

In order to compare our predictions with real fire behavior, we must use metrics to determine the quality of our simulations and be able to rank them. Several metrics exist to compare real and predicted values [48] and each one weighs the factors involved differently. The notation used in this kind of error functions is depicted in Fig. 2.14.

The cells around the map that have not been burnt by either the real fire or the predicted map are considered *Correct Negatives* (CN). Those cells that have been burnt in both maps are called *Hits*. The ones that are only present in the real fire and the predicted fire does not consider that will be burnt are the *Misses*. Finally, in the opposite case, the cells that the simulator predicts will be burnt and the real fire does not actually reach are called *False Alarms* (FA). Besides these factors, some equations take into account the real map

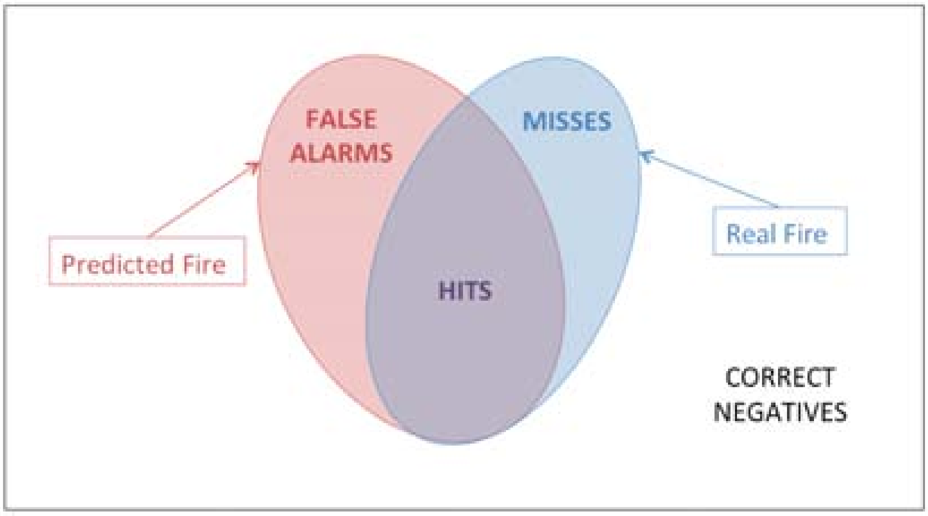


Figure 2.14: Factors that take part in error comparison equations.

(Real), the simulated or predicted map (Sim), the ignition map (Ini) or the total number of cells of the terrain (Total). This notation may vary but the meaning remains the same. For better ease of understanding and to simplify the equations, the initial fire is considered a point and can be removed from the equations.

In our research work we have been using the symmetric difference between maps. The values given by this error function are positive, but not in a closed interval, with the best value being 0 without an upper limit. We use the concept of union ( ) and intersection ( ) as factors in the equation but, as can be seen in Eq. 2.2, they can be translated to the notation presented previously.

$$\begin{aligned} Error &= \frac{\;-\;}{Real} = \frac{(Hits + Misses + FA) - (Hits)}{Real} \\ &= \frac{Misses + FA}{Real} \end{aligned} \tag{2.2}$$

In fact, both real and simulated maps can also be transformed as a combination of the Hits, Misses, and FA factors, and the equation could be reformulated replacing the following elements:

$$\begin{aligned} Real &= Hits + Misses \\ Sim &= Hits + FA \end{aligned} \tag{2.3}$$

This metric equally penalizes the misses and the false alarms. Another metric that was used to rank individuals was the critical source index (CSI), which gives us the rate of hits achieved from 0 to 1, with 1 being the perfect match between maps. It also weighs misses and false alarms in the same way.

$$Fitness = - = \frac{Hits}{Hits + Misses + FA} \tag{2.4}$$

The fact of equally penalizing both factors is not desirable in our research field. It is much more important to minimize the misses than to reduce the false alarms. The consequences of misses can cause severe damage, both to the environment and in human lives, while the false positives may represent an extra effort in fire-fighting resources.

The main problem of these metrics using our methodology is focused on the calibration stage. In this part of the methodology, we evaluate several scenarios, rank them using the error function, and then select the best parameter set after the calibration process to perform the prediction. In many real fire cases, the individuals that almost do not spread give back the best error values. Analyzing the shape of the other individuals, we saw that potential good predictions were discarded from the calibration process due to the high penalty generated by the false alarms. In order to solve this undesired effect, we changed the first equation in order to minimize the effect of false alarms. The new error function is depicted in Eq. 2.5.

$$Error = \frac{\frac{\cup - \cap}{Real} + \frac{\cup - \cap}{Sim}}{2} = \frac{\frac{Misses+FA}{Real} + \frac{Misses+FA}{Sim}}{2} \tag{2.5}$$

The latest equation has shown better behavior in the calibration stage than the other metrics. The individuals that overestimate have a better error than those individuals that underestimate, when the area is equal by excess and by defect.

## 2.5 Working hypothesis limitations

As will be widely analyzed in the next chapter, the working hypothesis of the two-stage prediction method supposes a hard restriction in many scenarios. It relies on the fact that the weather conditions will not suffer a noticeable change between the calibration and the prediction stage. Sometimes this assumption is hardly achievable. It is especially evident in large fires, when the terrain is rough and weather conditions undergo continuous changes during the simulation. Under these parameters, the two-stage method previously delivered poor predictions due to the high dynamism of such scenarios.

# Chapter 3

# Coupling complementary models

When we deal with real complex fires new challenges and problems must be taken into account to adapt the fire prediction system to this new situation. In these scenarios the uncertainty about the input parameters grows exponentially and the reaction must be fast and efficient in order to minimize the effects of such hazards.

Therefore, the information about the environmental conditions must be as accurate and reliable as possible, describing the particular conditions that affect the fire spreading at any point of the region under study. Usually, the available data about the conditions concerns certain points where the weather variables have been measured or describes an atmospheric value that is global to the whole area. Directly injecting these values produces a lack of reliability because we assume a global and uniform behavior in a heterogeneous terrain. Additionally, we also assume a continuous behavior of these parameters over time, which is uncommon in real scenarios. The temporal evolution of the input parameters changes dynamically depending on the time of day, the season or the specific features of the region.

To build an effective fire prediction system, we should take these considerations into account and provide a complete knowledge of the environment where the fire is taking place. Most of this information cannot be known

| **Type** | **Real fire** | **Prescribed fire** |
|---|---|---|
| **Area** | Hundreds of $m^2$ | Hundreds of ha. |
| **Duration** | minutes/hours | hours/days |
| **Conditions** | controlled | not controlled |

Table 3.1: Prescribed fires vs. real fires.

in advance or its resolution is not high enough to consider the topography irregularities. These issues can be faced using complementary models that can calculate or predict the behavior of the weather variables. These models can be divided into diagnostic and prognostic models. Diagnostic models can develop the global behavior of a parameter and deliver its behavior in detail, which is not time dependent. Prognostic models predict the behavior of one or more parameters using mathematical equations that define the dynamics that drive their evolution under certain initial conditions.

## 3.1 Complementary models

Some parameters present a spatial and temporal distribution that make the calibration process more difficult, since only an average value for the parameter along map and time can be selected. This assumption can be made when we deal with prescribed fires where the size, the duration, and the conditions are bound and known but in real fires this restriction is too hard because the background changes drastically (see Tab. 3.1).

The original Two-Stage prediction scheme suffers from two main handicaps. This scheme considers a uniform distribution of the parameters along the whole terrain and it does not consider prognostic models to enable dynamic parameters changes over time. Both restrictions have a direct impact on the quality of the prediction results. Thus, the original scheme was modified to be a multi-model prediction framework, where different complementary models were easy to couple in order to reduce that negative impact. Therefore, we focus on the meteorological conditions and, in particular, on the wind components, since these are the parameters that most affect fire spread.

In order to equip the two-stage prediction scheme with the capacity to react to sudden changes in environmental conditions, it becomes mandatory to fit environmental data coming from prognostic models, such as weather forecasting models, into the prediction scheme. Both prognostic models and wind field models are computationally expensive. So, any approach to coupling those models into a system (Two-Stage prediction scheme in our case) will need to carefully analyze the implications of the total execution time and resources needed, because when dealing with natural hazards such as forest fire, any fire spread prediction must be delivered faster than real-time fire evolution in order to be useful.

### 3.1.1 Wind field model

When complex terrains are considered, new features arise that must be taken into account from a prediction accuracy point of view. Local terrain features such as drainages, ridges and other topographical characteristics generate flow effects that can only be captured in high resolution models. The meteorological wind is modified by the topography, resulting in rapid changes in fire intensity on a small scale that can have significant influence on fire growth on a larger scale. Therefore, a single value for the wind representing the wind on each cell of the terrain is a very restrictive simplification. It is necessary to evaluate or estimate the wind on each cell of the terrain. The benefits of using wind fields in forest fire spread prediction have been discussed in many other works [38][49], and it has been tested in many scenarios with significant results. Therefore, to tackle this problem, a wind field model must be introduced to obtain the effective wind at the required level of detail.

The wind can be measured by meteorological stations, but the value measured in one meteorological station is a measurement at a single point, while at other points of the terrain, the value (speed and direction) of the wind can be different due to the orography of the terrain. The hills, mountains, valleys and canyons of the terrain modify the meteorological wind, creating a wind field with different values at each point of the terrain.

Moreover, the wind is one of the input parameters that most affects the fire propagation. So, it is necessary to introduce a diagnostic model that calculates the wind speed and direction at any point of the terrain, given a meteorological wind. In Figure 3.1, there is a scheme of a homogeneous wind map on the left and the corresponding wind field map generated by a wind field modeler is on the right. The selected wind field model is WindNinja, because it has a direct connection to FARSITE. It does not deliver a prediction of the wind behavior over time but WindNinja performs a surface wind diagnosis at a certain time instant.

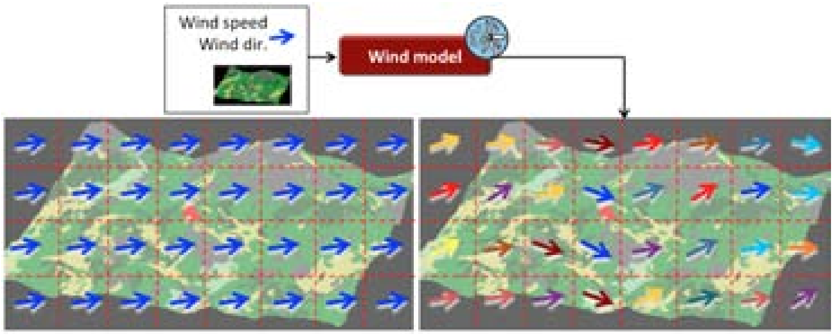


Figure 3.1: Scheme of a homogeneous wind map (left) and a wind field map given atmospheric wind components (right)

### 3.1.2 Meteorological model

The parameters calibrated by the 2st-BASIC scheme were considered constant during both time intervals and a single value for each parameter was used for the calibrating process and for the prediction stage. This methodology fits the DDDAS paradigm [50] since the prediction is dynamically driven by the system evolution.

However, there are several parameters that are not constant over time and they may vary dynamically. In the case of forest fires, a typical example is wind. In some cases, when the time interval is short, an average value for the wind can be a feasible value but, when the time interval is longer, in most cases a single value cannot represent the variability of the wind. We can estimate wind behavior by applying a complementary model.

We are going a step further in considering the dynamic behavior of such dynamic parameters. We propose an extension of the existing prediction scheme that takes into account the dynamically changing parameters by coupling a weather prediction system with a DDDAS Forest Fire Propagation Prediction system.

An additional advantage of the introduction of the wind parameters at the calibration stage instead of calibrating them as another gene is that the search space for the calibration techniques is significantly reduced and, therefore, the other parameters considered can reach better values, and this fact allows for smaller calibration errors in less time.

In the prediction stage, it is not possible to introduce the exact dynamic values of the weather parameters beforehand. To overcome this limitation a numerical weather prediction (NWP) model can be used to predict the dynamic behavior coupling the previously described forest fire spread prediction system with an NWP. In this case, the quality of the forest fire propagation prediction significantly depends on the quality of the weather parameters prediction obtained from the NWP. A similar idea has recently been proposed in [51][52][53]. These works show the benefits of considering the influence of the heat flux generated by the fire itself into the surface wind of the meteorological model. However, those approaches are focused on interfacing intra-models for executing a single fire simulation evolution. In our work, as has been stated, we do not rely on a single simulation, but on the execution of thousands of them. The way we propose to couple both models is a pipeline in which the values obtained at each NWP step are fed into the corresponding fire simulation step.

One of the most extended NWP models is the Weather Research and Forecasting (WRF) [54]. It is a mesoscale weather prediction system that is used for several meteorological applications worldwide. It can work with real data or can reflect ideal conditions, depending on the purpose of the simulation (weather research, forecasting, etc.), and provides a resolution that ranges from meters to kilometers. The WRF model was designed to be a parallelizable and extensible software. Most meteorological services rely on WRF as one of the main weather models to perform their simulations and predictions.

## 3.2 Two-Stage prediction schemes coupling complementary models

In this section, we want to introduce the prediction schemes resulting from the addition of the wind field modeler and the data coming from an NWP model. The impact of including these models in the two-stage prediction system is also analyzed, as well as the benefits that they introduce.

### 3.2.1 Coupling Wind field Model to the two-stage method (2ST-WF)

The original two-stage prediction scheme was designed to only accept environmental input data at a low resolution. In particular, wind components were considered at a mesoscale resolution, that is, a single wind direction and wind speed for the whole terrain was used. In order to overcome this constraint, a wind field modeler (WindNinja) has been included in the two-stage prediction scheme enabling the system to deliver spread predictions, which reflect the influence of the terrain at a high resolution level. WindNinja is able to generate these wind fields, but, depending on the terrain size, it has a high computational cost.

In this scheme, each worker process receives the parameters representing one particular scenario, and it is then necessary to run the WindNinja wind modeler followed by the Farsite fire propagation simulator. This pipelined worker scheme is depicted in Figure 3.2.

It can be observed that this strategy implies that each individual of the GA population, which represents a possible fire scenario, has to execute an instance of WindNinja and one of the fire spread simulator. This approach represents hundreds of simulations for every prediction interval. For example, if the population size of the GA is set to 50 individuals and the GA is evolved 5 generations, the number of total simulated scenarios will be 250.

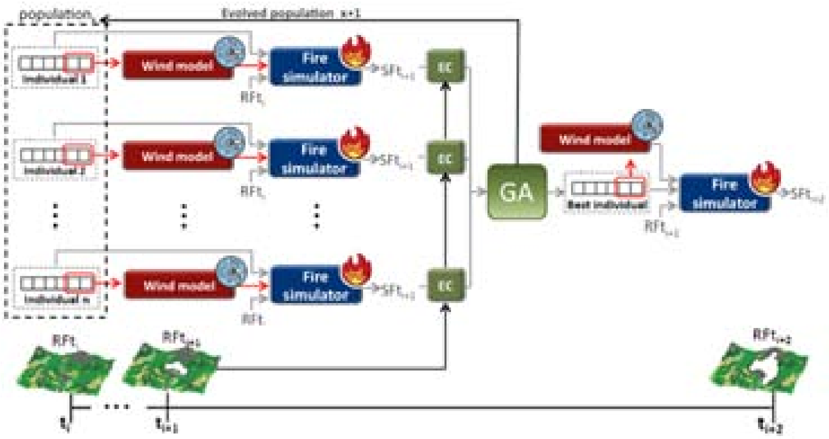


Figure 3.2: 2ST-WF prediction scheme

Therefore, the number of times that WindNinja should be executed will also be 250. This approach is more realistic, but the computational time required by WindNinja is quite long. The execution time of the WindNinja modeler depends on the map size and topography, but it usually takes some minutes on a single core. If there are not enough computational resources, the provided prediction may be achieved too late.

### 3.2.2 Coupling Meteorological Model to the two-stage method (2ST-MM)

It is well known that wind can change suddenly in speed and direction. During the calibration stage, it is feasible to receive information from meteorological stations frequently (every 30 minutes or even more frequently). In this case, the wind speed and direction do not need to be calibrated since they are received from direct measurements. However, during the prediction stage, such values are not available beforehand. So, it is necessary to introduce a meteorological model that can provide the expected values for the meteorological wind speed and direction. These values can be used during the prediction stage [55]. We assume that meteorological predictions are available from a meteorological service, so it is not necessary to compute the meteorological forecast on the fire spread prediction platform. It implies that the computational cost of the 2ST-BASIC forest fire propagation prediction (see Fig. 3.3) does not increase.

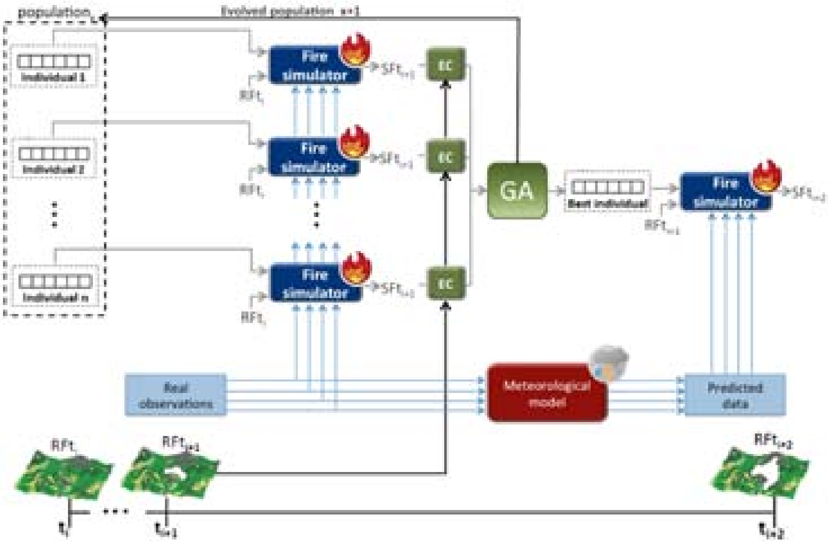


Figure 3.3: 2ST-MM prediction scheme

It can be stated that including wind dynamic behavior in the two-stage prediction process must improve fire spread prediction quality when dealing with large-scale forest fires. Such wind dynamic behavior must be considered in both stages, calibration and prediction.

In the calibration stage, the data concerning the time interval $t_i$ - $t_{i+1}$ is available. So, in this stage, the simulations executed to reproduce the behavior during the interval $t_i$ - $t_{i+1}$, can be fed the real measured values of the wind parameters during that interval. It implies that the simulation of the forest fire propagation does not consider constant values for the wind parameters, but the measured value for each time subinterval is injected in the simulator. So, these wind parameters are not introduced in the individuals of the GA and are not calibrated. The other parameters concerning moisture contents and vegetation features are calibrated in the calibration stage.

In the prediction stage, the best individuals coming from the calibration stage are used to deliver a prediction for the time instant $t_{i+2}$. All the parameters of the individual remain the same except the forecasted parameters given by the meteorological service. These parameters will be replaced by the old ones, used in the calibration stage.

### 3.2.3 Coupling Wind field Model and Meteorological model to 2ST-BASIC (2ST-WF-MM)

In order to join the improvements of the previous prediction strategies in a single approach, we also propose a hybrid scheme, which is a trade-off between the accuracy obtained in the prediction results and the time spent to reach them. In the scheme described in 2ST-MM, the system can be fit with weather data provided from real observations during the calibration stage and, at the prediction stage, the injected data comes from a weather forecast model. In this hybrid scheme, we interpose the wind field model just before injecting the wind components to the prediction system. Therefore, for each observed wind speed and wind direction and for each predicted wind component, one wind field simulation will be run. In the calibration stage, that means that the number of wind field evaluations is drastically reduced because each individual of the population will use the same wind fields. This effect compensates the slight increase in computational time introduced at the prediction stage due to the evaluation of the wind fields in this case. For example, if the meteorological model gives one prediction per hour and the prediction interval is 12 hours, it will be necessary to generate only 12 wind fields.

## 3.3 Computational requirements of the proposed schemes

The two-stage prediction methodology is by itself a platform with a high computational cost. Evaluating hundreds or thousands of scenarios in order to find the one that best matches the current fire behavior supposes a challenge that current personal computers cannot solve in an acceptable time. the use of High Performance Computing resources becomes mandatory to deliver our predictions. Furthermore, adding new models to the system introduces

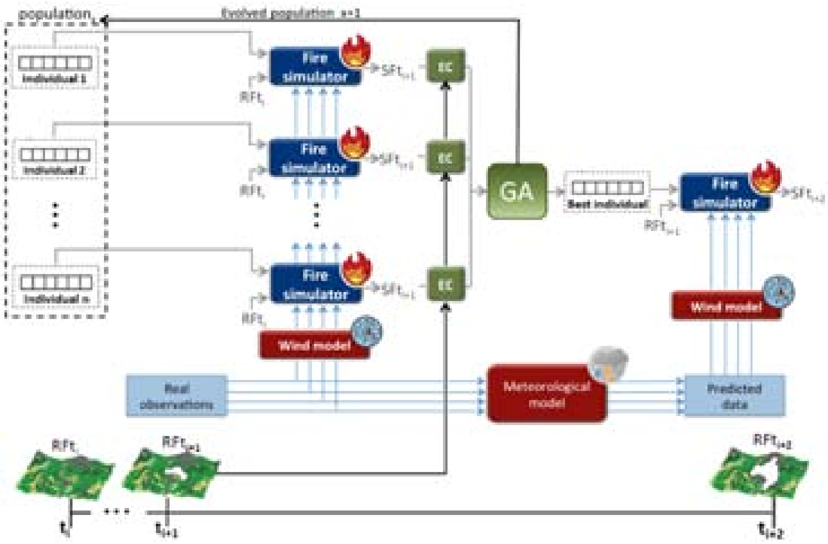


Figure 3.4: 2ST-WF-MM prediction scheme

a computational overhead that must be tackled. In our case, the calculation of the wind field becomes a bottleneck as the size of the terrain increases. As has been commented, the computation of the meteorological model is done by external services, so it does not significantly increase the computational cost. However, this detailed information about the weather conditions may increase the fire simulation time. Therefore, we propose a hybrid parallel framework that takes advantage of parallel programming paradigms as MPI and OpenMP and eases the process of reproducing and simulating a fire scenario.

## 3.4 Hybrid MPI-OpenMP FFSS prediction framework

The FFSS (Forest Fire Simulation System) is a multi-model forest fire simulation and prediction system, which is highly parallelizable due to its design pattern (a Master-Worker) and the parallel programming paradigms used (MPI and OpenMP). This framework (see Fig. 3.5) is intended to be a simulator-independent system that considers the fire simulator a black box.

That means that not all of the operations that the system performs depend on the simulator chosen. There are certain constraints and considerations that must be taken into account. The simulator chosen should be based on Rothermel s fire spread model to be fully supported by the system. The simulator must also deliver output maps with the values representing the time of arrival of the fire. These maps are necessary to compare the output with the real fire behavior. Finally, some internal functions must be extended or overloaded to consider the peculiarities of each simulator. Usually, the simulation parameters or the input parameters are not exactly the same from one simulator to another. This forces us to extend, modify or overload the binding functions that enable the proper data swapping between the fire simulator and the rest of the system.

The FFSS framework relies the simulation settings on a single file. This text file contains the paths of the files and the folders where the input and output files must be found or created, the global parameters of the simulation and the specific parameters of every simulator or module of the framework.

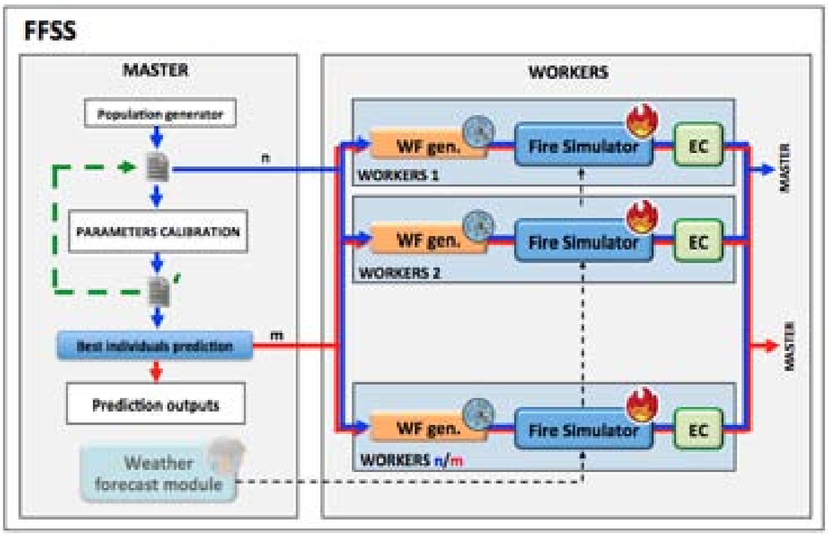


Figure 3.5: Master-Worker scheme of the FFSS system

The multi-model schemes described in the previous section rely on Genetic Algorithms for the calibration stage. The GA fits very well in the Master/Worker programming paradigm. In this case, the Master process generates the initial random population. Then, it distributes the individuals to the Worker processes that can be executed independently in different cores.

This scheme was first implemented using MPI where the master process sends a set of scenarios to all the workers using MPI blocking directives. Each worker process was mapped into an MPI process, which executes WindNinja (wind field evaluation) in a pipeline fashion, Farsite (fire spread simulator) and the symmetric difference between the real and simulated burnt area (error evaluation). Finally, each worker returns the error obtained to the master process that ranks the individuals and applies the genetic operators to generate the next population. This process is repeated for a fixed number of iterations. In this case, each core in the underlying platform executes one worker process, so the maximum number of worker processes is limited to the number of available cores in the platform. Therefore, if there are enough available cores, the execution time for each iteration is limited by the execution time of the scenario whose simulation lasts longer. The quality of the calibration depends on the available elapsed time to provide the propagation prediction and the number of individuals on each iteration, but it must be considered that, in these emergency situations, response time is a critical issue [56]. For the previous studies [45], it was stated that 5 iterations usually provide a successful calibration.

However, to cope with those strict real-time constraints, a hybrid MPI-OpenMP scheme was proposed to reduce GA s iteration time. This hybrid approach includes a second-level parallelization on the worker processes by using OpenMP within both the wind field model (WindNinja) and the fire spread simulator (Farsite). WindNinja was originally implemented using OpenMP, so it is capable of benefiting from multi-core processors. It is worth mentioning that WindNinja is a very time-consuming element of this scheme, so an approach to be analyzed will be the possibility of dividing the working domain of WindNinja into subdomains, which will be run on GPU architectures. However, Farsite was not a parallel simulator. Artes et al. describe a parallel version of Farsite in [57], which exploits the potential internal loop parallelisms to benefit from multi-core platforms.

### 3.4.1 FFSS framework initialization file

The FFSS initialization file defines the whole background where the simulation will run. It is divided into four sections:

- Section 1: Global simulation parameters
- Section 2: Calibration technique parameters
- Section 3: Fire simulation parameters
- Section 4: Wind model parameters

#### Global simulation parameters

This set of parameters defines the main features of the simulation. In this part, the path of the initial population to enhance the calibration, the number of individuals, the calibration technique used, the output traces of the framework, the models that will be enabled during the simulation, and some parameters about the computational resources that will be available (number of cores per MPI thread, threads used by the fire simulator, etc.) must all be depicted. It is also necessary to determine the ranges in which the parameters involved in the calibration stage will vary. For now, only a calibration technique based on a Genetic Algorithm is included, but it is possible to attach new techniques without disturbing the rest of the system.

#### Calibration technique parameters

Here, the specific values for the chosen calibration technique will be defined. As has been commented above, we include a genetic algorithm as the single calibration technique available for now. In this case, the main parameters concerning this technique are the number of generations, the number of individuals that will pass to the next generation by elitism and the crossover and mutation rates. It is also possible to define your own crossover pattern if the standard crossover does not meet your expectations, and, in this case, the chosen method must be stated.

#### Fire simulation parameters

This is the widest section since it involves the majority of the inputs of the system, and most of the parameters required to simulate a fire. A first group consists of the definition of the paths of the input and output, and the names of the files that will be used or created during the simulation. The landscape, wind, weather, ignition and shape output files are examples of this group.

In a second group, we can find those parameters that define certain actions that the fire simulator can perform, as well as some specific behavior parameters. In this case, we can define the internal time step and the resolution of the simulator, the output files to be created (raster, shape, and/or vector) and the maximum time of a single execution, among others.

The last group is related to the duration and times to be simulated, both in the calibration and in the prediction stages. Here, the initial and the final dates (month, day and hour) for each stage are established, and we provide the comparison perimeters of each stage.

#### Wind model parameters

This brief section contains the wind field simulator parameters such as the elevation or landscape file, resolution, wind height capture and the wind initialization method, among others. The wind simulator adds a considerable overhead to the system, so these parameters should be carefully modified depending on our requirements and available resources.

## 3.5 FFSS framework input and output files

In this section, the main input and output files will be detailed. The system as it is currently composed will be taken as a reference, with Farsite as the fire simulator, a genetic algorithm as the calibration method, WindNinja as the wind field simulator and the dynamic injection of meteorological data provided by an external meteorological service.

### 3.5.1 Input files

Using the categorization depicted in section 3.4.1, the input files corresponding to the global simulation parameters are the initial population, the range file and the fuels used file.

The initial population contains the parameters set for every individual that will be evaluated in the calibration stage. These populations must be generated using the information contained in the range file. This file specifies the maximum and minimum values that define the variation interval of each parameter. It is also essential to generate new valid values to mutate a parameter when the GA is executed. Finally, the fuels used file defines the different fuel models involved in the fire simulation. It is an auto-generated file that can be used to adjust the spread factor of each vegetation. Optionally, a file where the core affinity is specified can be introduced, to force specific mapping between workers and cores or nodes.

In fire simulation parameters group, the files that should be defined are the terrain files, the base settings files, the ignition file, the calibration perimeter file and the prediction perimeter file. In our case, the terrain files are included in a single .lcp (Farsite landscape file) that contains the features of the area (elevation, slope, aspect, fuel and canopy cover). The base settings files are auxiliary files used to insert the information of each individual in the files supported by the simulator. They contain tags that will be replaced by the proper values. The ignition file defines the point or the perimeter from which the fire will spread. When each individual is executed, it must be compared with the real fire perimeter at a certain instant, and its path must be indicated. Likewise, if we want to evaluate our prediction, the real perimeter should be stated.

As wind model input files, the same landscape file used to simulate the fire can be used or specifying the elevation file.

### 3.5.2 Output files

The framework delivers much information about the simulation process. Besides the intermediate files that correspond to each individual execution for every generation, the system provides some global information. It generates a trace file that annotates the time of each generation and the global time. It also generates a file per generation that shows the calibration error for each individual. It also gives the prediction errors of n individuals, as specified in the configuration file. If we include the *gnuplot* library, the system is capable of generating some graphics related to time or quality. The framework also stores the shapes of the best individuals, both in calibration and prediction.

## 3.6 MPI execution scheme of FFSS framework

As has been commented in the introductory part of this chapter, the MPI scheme of the FFSS framework is based on a Master-Worker pattern. The master process generates the individuals and distributes them among the available workers. Fig. 3.6 shows schematically a scenario where there are as many workers as individuals, so the generation time will be determined by the slower individual.

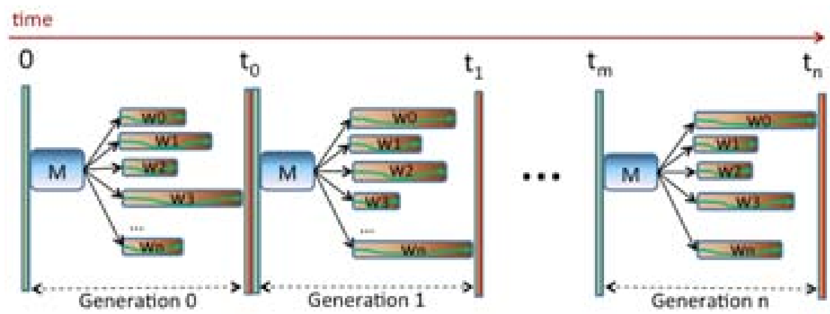


Figure 3.6: Master-Worker generation time

In this case, the master process delivers all the individuals and waits for the response of each of them to generate a new population. If these conditions cannot be supplied, and the number of workers is lesser than the individuals to evaluate, the master process sends individuals to the workers on demand.

One of the problems of both alternatives is the possible load imbalance between workers. An example of this issue can be observed in Fig. 3.7. From this figure, we see that longer individuals cause many workers to spend too much time waiting. This is an undesired situation (poor efficiency), because HPC resource time must be exploited as much as possible.

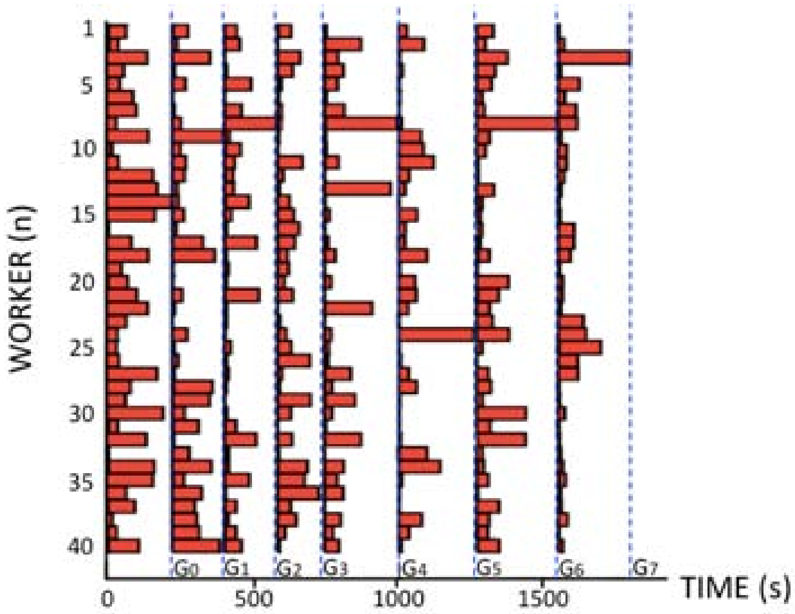


Figure 3.7: Workers load in a scenario with one individual per worker

In the basic implementation of the framework, there was not any policy for choosing which individual would be sent next. The latest works in this group have developed a methodology to classify the individuals according to their estimated execution time. Based on this knowledge, it is possible to define a policy to favor a good load balancing. This methodology is based on executing thousands of executions, defining a time classification, and, using decision trees, being able to predict the class of a certain individual.

## 3.7 OpenMP parallelism in FFSS framework

Another option to reduce the load imbalance is to reduce the execution time of each individual. In keeping with this idea, we profit, when possible, from the multi-core and many-core architectures to speed up the execution of the worker processes (see Fig. 3.8).

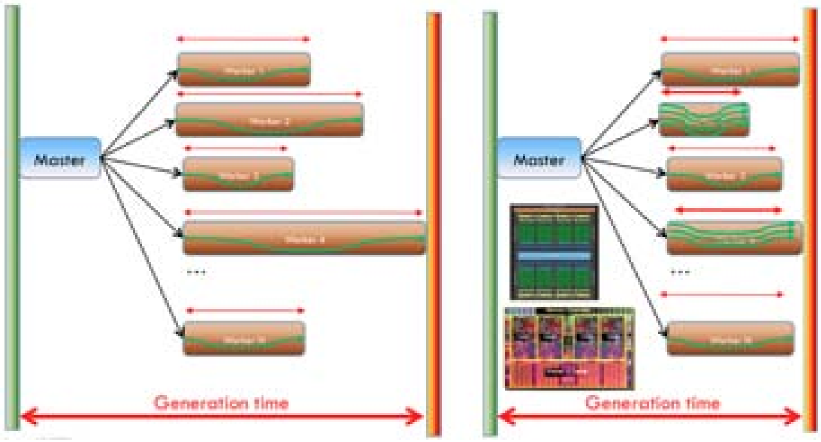


Figure 3.8: Worker reduction example scheme

As has been commented at the beginning of this chapter, the worker executes at least one instance of the fire simulator and, depending on the prediction scheme selected, a wind field calculation. Therefore, by properly allocating the available cores among the workers, the individual time and, consequently, the generation time and the global execution time, can be reduced.

Returning to the above idea of classifying the individuals according to their execution time, a policy to assign resources dynamically is being implemented and tested. Those individuals that fall in slower classes will receive more resources (cores) than faster individuals. By performing a study of the expected time reduction per class depending on the number of cores provided, we will be able to guarantee a significant reduction in execution time.

In Figures 3.9 and 3.10 two histograms are shown with 12,000 Farsite instances stacked by their execution time. In the first case, each instance is executed in a single core. In the second, 4 cores per Farsite instance have been allocated.

They show a clear tendency to compress the histogram to the left, which means that the longest executions have notably reduced their time. The shortest individuals do not have a significant reduction, so, to be efficient, the system should identify the different classes and distribute the resources accordingly.

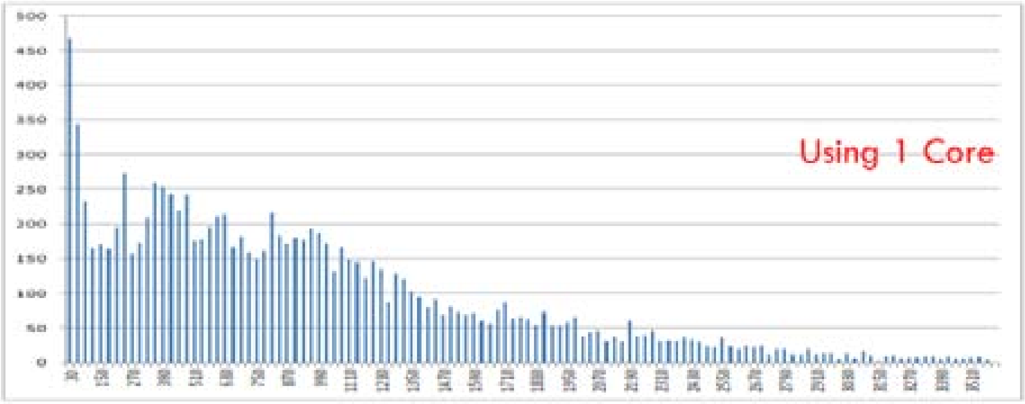


Figure 3.9: Histogram showing 12,000 Farsite executions stacked by execution time (1 core)

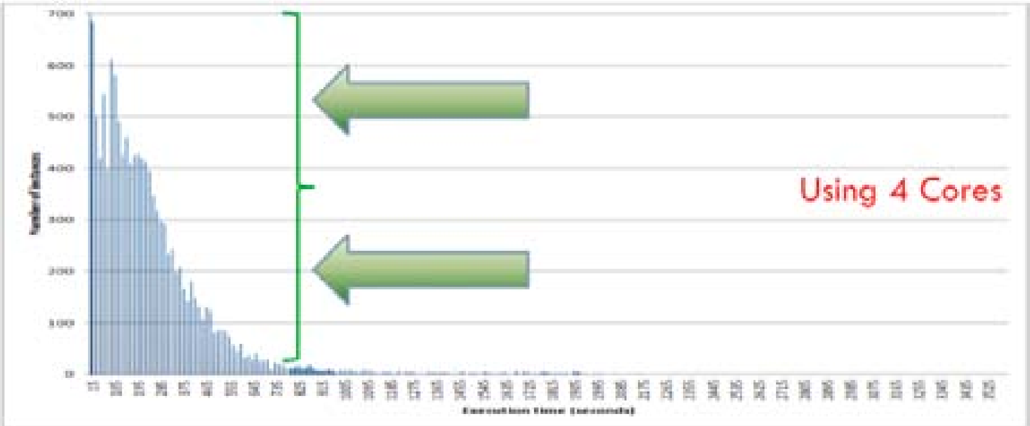


Figure 3.10: Histogram showing 12,000 Farsite executions stacked by execution time (4 core)

## 3.8 Wind field acceleration

The calculation of the wind field is, in many cases, one of the most time demanding models in our parallel system. The schemes that include this model (2ST-WF and 2ST-WF-MM) perform the wind field execution differently. The 2ST-WF scheme needs to compute a wind field simulation for each individual in the GA and for each generation. The 2ST-WF-MM scheme analyzes the weather predictions provided by the weather forecasting service and executes a wind field simulation for each wind prediction. The first scheme implies the execution of hundreds of instances of the wind field simulator, and they must be performed while the GA is being carried out. Instead, the second scheme usually requires tens of executions, and they can be performed just when we get the weather forecasting corresponding to the desired time interval. WindNinja, the wind field simulator used in this work, requires a certain execution time that basically depends on the map size. This simulator includes OpenMP directives to parallelize some parts of its code due to its high memory requirements. In a typical computing node, it lasts too long (around an hour for a 1500x1500 cell map with 30m resolution) and requires a high amount of main memory (12GB for the last example). These restrictions mean that the prediction schemes previously seen require an unaffordable time, especially the 2ST-WF scheme.

The first approach to tackling this problem is based on the partitioning of the map to reduce the size of the terrain that will be introduced to WindNinja. This idea is depicted in Fig. 3.11. The master process divides the map and distributes each part to the workers, which compute the wind fields and give the results back to the master. Then, the master recomposes the global wind field using the specific wind fields corresponding to each part. Although it is a simple concept, partitioning the map introduces new issues that must be taken into account to maintain the quality of the partitioned wind fields.

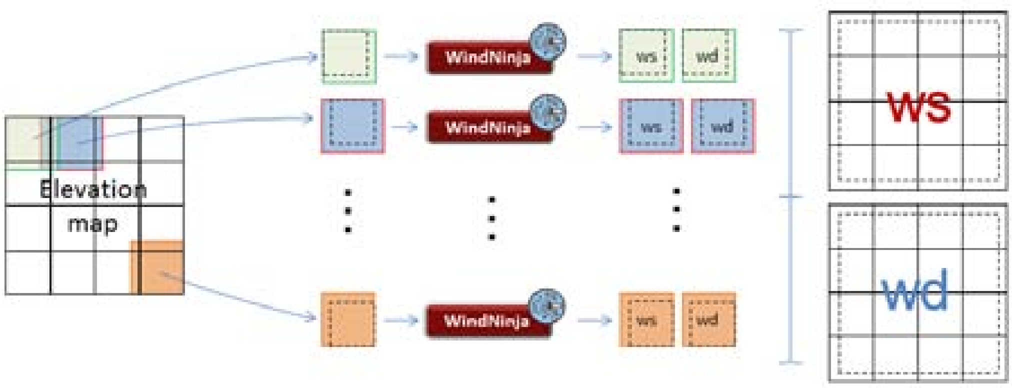


Figure 3.11: Wind field partitioning scheme

WindNinja is based on mass conservation equations and delimited boundary conditions. The results close to the borders of the map are not reliable because the system needs some cells to stabilize the values. Therefore, when we divide the map, we indirectly introduce new boundary errors to the final wind field. These boundary effects can be seen in Fig. 3.12, where three vertical parts have been evaluated using WindNinja, and the results have been merged in a single map.

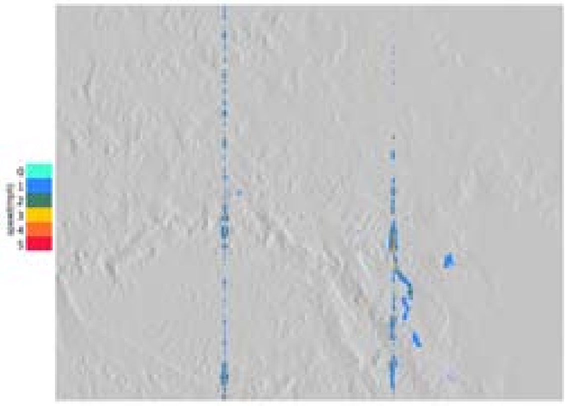

Figure 3.12: Boundary effects produced by map partitioning

To avoid these effects, it is necessary to introduce a certain overlap between parts, discarding those unstable values that appear near the borders of each division. Applying a certain overlapping rate, we can minimize the boundary effects (see Fig. 3.13).

Using this technique, we can significantly reduce the execution time, and the speed up is improved when we increase the number of parts, as can be seen in Fig. 3.14. The RMSE (Root Mean Square Error) error that shows the difference (in this case the wind speed) between the global map and the partitioned map in average is also depicted. The graphic shows that the error grows as the number of parts is increased.

Therefore, it is important to find a trade-off between the number of parts and the RMSE error, to reduce the execution time without compromising the quality of the partitioned wind field. It has been tested that using the resulting wind field coming from the partitioning technique does not signifi-

Figure 3.13: Boundary effects reduction using a 14% overlap

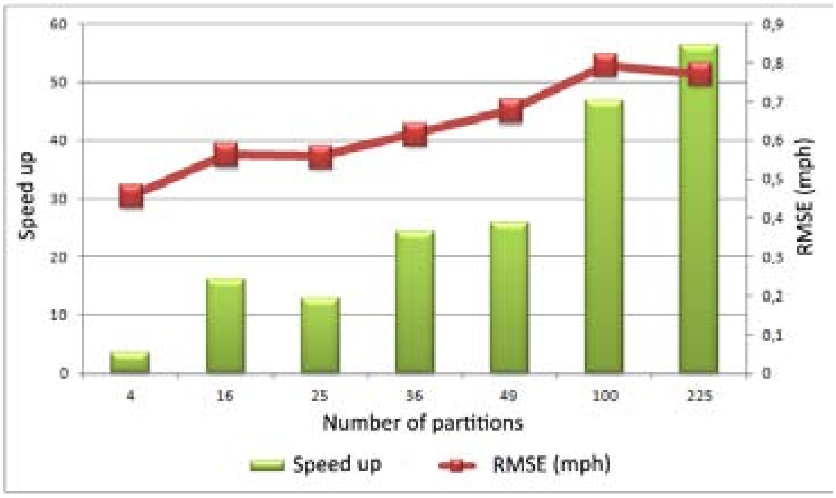


Figure 3.14: Speed up and RMSE error for each number of parts

cantly affect the accuracy of the forest fire spread prediction, compared with the system behavior using the non-partitioned wind field.

# Chapter 4

# An operational framework to predict forest fire behavior using complementary models

In this chapter, we want to introduce our methodology to predict and reproduce forest fires and dissect each one of its modules. The centerpiece is the fire simulator, which, in our case, is Farsite, previously analyzed in Chapter 2. The simulator is fed by a set of parameters that must be acquired and processed to be compatible with the expected simulator inputs.

Another module of the methodology is responsible for calibrating the input parameters based on the two-stage prediction technique that has been widely analyzed previously. This module includes a Genetic Algorithm that performs the evolutionary operations required to tune the parameters generation after generation.

We rely on some prediction schemes that gather and inject the information coming from complementary models in the system, in order to increase the knowledge that we have about the fire environment. Every scheme uses or discards the information from these models depending on our needs. It is not always beneficial to inject information, because, in some cases, data may be corrupt, come from unreliable data sources, or be acquired from too far from the seat of the fire. In those cases, it could be preferable to calibrate these parameters instead of injecting the data provided by the complementary models. Therefore, the decision must be taken by the expertise staff depending on the information quality.

Finally, the system collects all the output data about the simulation and provides some useful information about time and quality factors. Fig. 4.1 shows the structure of the system, and the different modules that correspond to each of the methodology parts can be seen.

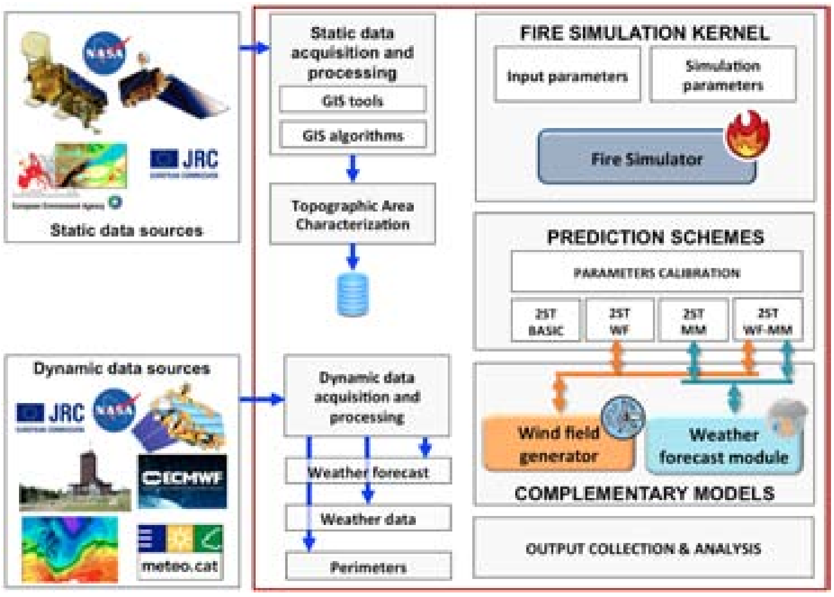


Figure 4.1: Scheme of the operational framework

As has been stated in the previous chapter, parameters can be classified according to their behavior in static and dynamic parameters. According to this feature (static or dynamic), the ways of gathering and processing the information to obtain the corresponding GIS input files are quite different. In the case of static data, the pre-processing and organization of the required layers in the proper format could be done prior to the hazard occurrence. There are certain constraints related to the terrain dimension that should be considered accurately, but the process of homogenizing the precision, projection and datum could be done off-line, and this data could be characterized and ready to be used when a crisis occurs.

If the static data of a region is available before a fire occurrence, the effort must be focused on those parameters that vary dynamically during the simulation or that depend on the fire scenario studied. In this case, it is necessary to collect them in real-time, thanks to the different data sources and services that can provide this information. Obviously, this case is the most critical since we depend on third-party frequency of data arrival and data format. Therefore, the conversions and the injection must be done in an on-line mode while the simulation is being carried out.

In the following sections, the relevant aspects and acquisition processes for both static and dynamic input data shall be stated.

## 4.1 Topographic Area Characterization

This section is intended to describe the steps needed to perform a forest fire simulation in a real-time scenario from scratch. That means that the system must respond quickly in order to be an effective decision support system, and it is fully self-sufficient if we provide it with all the data described in this section. All required input, their sources and the handling process will be defined.

Performing a whole topographic characterization of a region requires time that cannot be assumed in a real-time emergency. It makes no sense to characterize areas of little or no interest because resources are limited and must be focused on providing efficient service when a real hazard occurs. For these reasons, it is highly recommended to establish a sensible criterion when selecting areas of special interest.

When we deal with large topographic areas, it is necessary to reduce the complexity of the problem by dividing the area into partitions. These partitions should cover the region under study and must have a computationally treatable size. There is not a fixed or recommended partition size, so the decision may be based on political arguments or the availability of computational resources. In addition, some common sense factors should be considered. The area should be big enough to be able to simulate a large fire, critical zones must have overlapping partitions and non-burnable areas can be discarded. In our research group, we have focused our study on the

European region. Using exposure hazard risk maps (see Fig. 4.2), historical fire occurrences, and forest fires risk trends across Europe, we can decide which areas must be fully characterized.

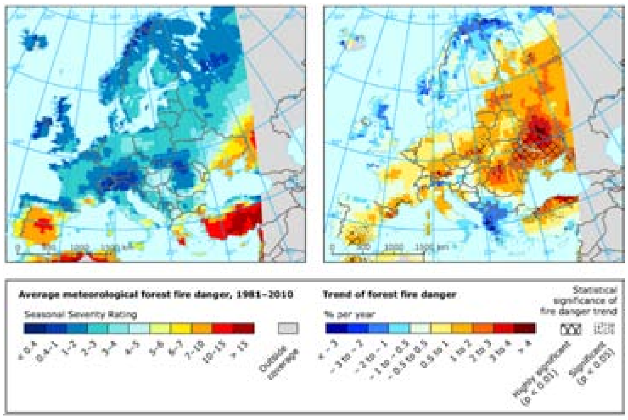


Figure 4.2: Average forest fire meteorological danger and future estimated trend in Europe

In Figure 4.2, the average forest fire meteorological danger is depicted, as well as the predicted behavior of this danger for the next years. This risk is measured using the Seasonal Severity Rating (SSR), which allows us to compare the differences in fire danger in a region over time. As we can observe, the southern countries are historically the most affected by forest fires. But the Mediterranean region is a huge area that must be considered separately for forest fire management purposes. Thus, what is the best map partitioning scheme for large topographic areas like this? We try to imitate the EFFIS manners in order to cope with European standards. Therefore, the characterization methodology subsequently described relies on EFFIS aspects to determine the *Map Partitioning Scheme* and the *GIS Input Files Generation*, which are the two basic steps of the proposed topographic area characterization methodology.

### 4.1.1 Map Partitioning Scheme

As mentioned above, we trust European standards to define our *Map Partitioning Scheme*. EFFIS rises as the standard European system for centralizing all forest fire related data. Therefore, since EFFIS uses NUTS levels as information domains, we have selected the NUTS 3 divisions as starting domains to deploy the proposed characterization methodology. Figure 4.3 shows the European NUT3 division.

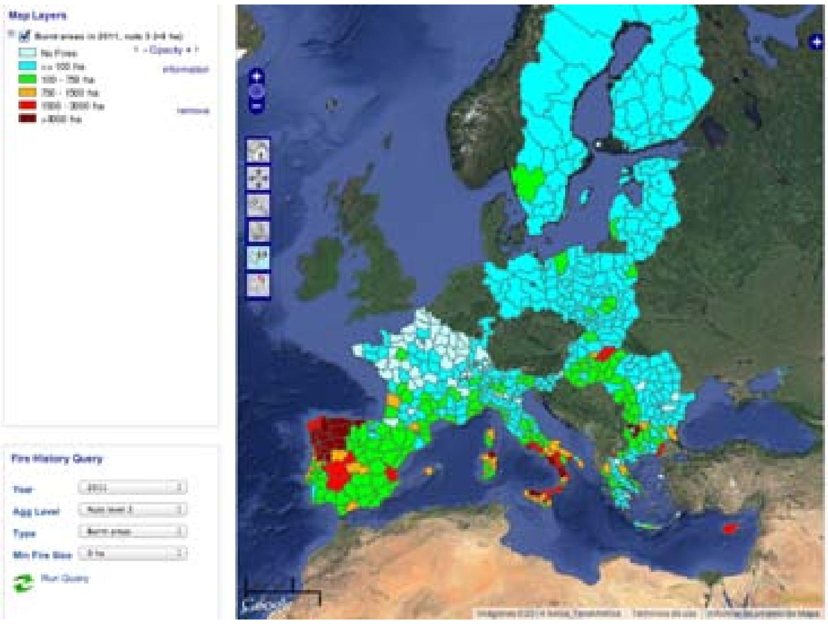

Figure 4.3: NUTS 3 Europe division, with burnt area information in 2011

NUTS 3 regions are described as small regions such as provinces with sizes varying from 10 to 100 thousands square kilometers. Obviously, NUTS 3 division generates regions that are too big when talking about forest fire spread simulations. For that reason, we introduce the concept of *fine grain tile (fgtile)*. An *fgtile* is a topographic area obtained by downscaling NUT3 domains to square, equal-size portions with a maximum dimension of around three thousand square kilometers (50km x 50km). Afterwards, for each *fgtile*, one can gather the static data needed for forest fire simulation purposes, co-register to the same resolution, projection, extension and datum and store it in a standard repository database. However, in order to be efficient when gathering the data of all defined *fgtiles*, we propose a non-uniform *fgtile* distribution across a certain NUT3 region. For that reason, we rely on forest fire occurrence maps to determine the areas with a higher rate of occurrences as the so-called Areas of Interest (AI) where the *fgtiles* will have higher density. This fact allows for the selection of the *fgtile* which best fits with the fire occurrence under simulation.

Therefore, the proposed methodology can be described as follows:

- Step 1: Selecting NUT3 region.
- Step 2: Analyzing NUT3 occurrence forest fire map.
- Step 3: Determining the Areas of Interest (AI) according to their rate of fire occurrence.
- Step 4: Creating and distributing the *fgtiles* within the NUT3 region, increasing the density on the AI obtained in Step 3.
- Step 5: Generating co-registered GIS files for all *fgtiles*, and aggregating them to the database (reported in the next section).

### 4.1.2 Generating GIS Input Files: Static data

First, it is necessary to acquire the static data that defines the terrain features. As commented in Section 4.1, this input can be built before the hazard occurs.

#### Elevation map

The elevation map is generated using satellite images from ASTER, which is a system embedded in NASA Terra satellite. ASTER ([58]) is capable of capturing detailed images of Earth for different purposes (surface temperature, reflectance and elevation). This system can take images at a resolution from 15 to 90 meters. In 2009, the Global Digital Elevation Map (GDEM) was released which makes it possible to acquire the elevation map of any zone of the Earth - it covers 99% of the Earth. It is composed of 23,000 tiles, and more than 1.3 million images from ASTER were necessary to build it. GDEM has a resolution of 30 meters, and this is the resolution that we usually work with to study forest fire scenarios. The output of this system is a set of GeoTIFF images.

The Farsite fire simulator uses ESRI ASCII raster files to generate the encapsulated landscape file. This format has two sections, the header and the data. The format is detailed in Figure 4.4.

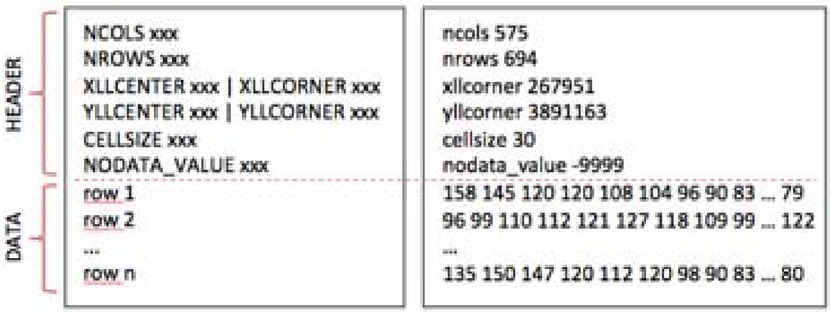


Figure 4.4: ESRI ASCII file format

### Slope and aspect maps

Besides the elevation file, it is necessary to provide the slope and aspect files. This information can be obtained from the elevation file, but Farsite cannot extract it. Many GIS tools (QGIS, Miramon, ArcGIS, etc) allow us to perform most of the transformations and conversions needed, and we can achieve the slope and the aspect from a certain elevation map. They use methods based on calculating the slope and the aspect values of a cell, depending on the elevation values of its neighbors (4 or 8). An elevation, slope and aspect map can be seen in Figure 4.5.

The easiest way to calculate slope and aspect is to use the gradient. It can be defined with two components as can be seen in the equation (4.1).

$$\vec{g} = \left( \frac{\partial z}{\partial x}, \frac{\partial z}{\partial y} \right) = (b, c) \tag{4.1}$$

Using this definition, the slope will be obtained by applying the equation depicted in Figure 4.5 from Section 4.1.2, where $b$ and $c$ are the components of the gradient.

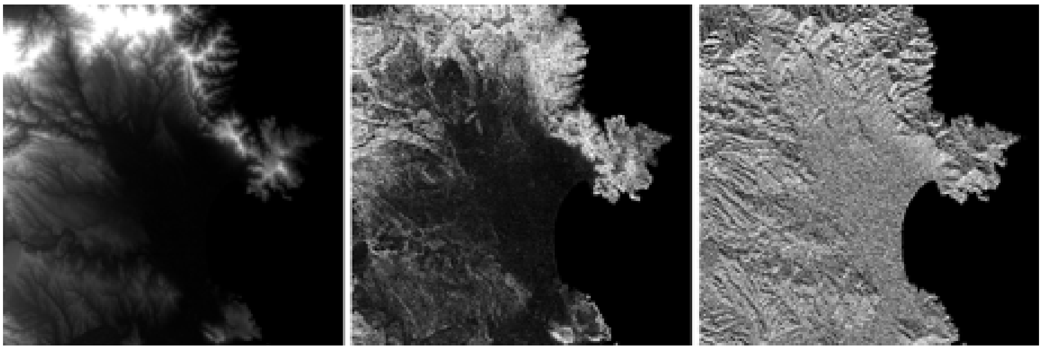

Figure 4.5: Elevation map (left), slope map (center) and aspect map (right).

**Vegetation map**

The vegetation map (or fuel map) is a raster file that describes the predominant vegetation in each cell. This information is provided by the Corine Land Cover project [59] of the European Environment Agency. This project studies the land use (not only vegetation) of European countries and generates maps that distinguish 44 classes divided into three levels. The most general level has four groups: artificial surfaces, agricultural areas, forests and semi-natural areas and wetlands. We do not use this map directly, because the Farsite fire simulator does not understand this categorization. Farsite only admits the 13 models defined by Rothermel and Albini [18] and those custom models previously defined by the user. In this work, we have used the Corine Land Cover for the Rothermel & Albini conversion done by the Joint Research Centre. This institution is under European Comission control and, among many other activities, it also studies forest fire effects and management and generates detailed annual reports of fires in Europe. These vegetation maps influence the way the fire spreads and give us more information about the terrain under study. Figure 4.6 shows a vegetation map where each color corresponds to a vegetation type.

**Fuel moistures**

Estimating fuel moisture values is not an easy task, and the most accurate way to perform this task is to directly take measurements over the whole area

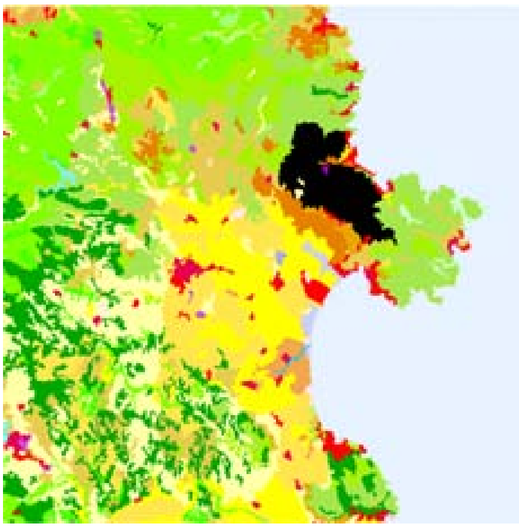

Figure 4.6: Fuel map

being studied. This is extremely demanding in terms of economical costs and time required. Other approaches try to achieve them using remote sensing and satellite imagery ([60] [61]). These works only study live fuel moisture contents because they are much more variable. Dead fuel moisture content is usually closely related to meteorological conditions.

**Canopy cover map**

The canopy cover map used in this work is a homogeneous map with a single value representing the tree crown coverage. There are techniques that estimate canopy cover from satellite images, but we do not have enough information in relation to the studied area. The lack of reliability about this parameter is minimized by the calibration techniques that tune the parameters to reduce the uncertainty.

## 4.2 Dynamic data acquisition and processing

### 4.2.1 Fire perimeters

At the European level, we obtain perimeters from Terra and Aqua satellites, which have an instrument aboard called MODIS (or *Moderate Resolution Imaging Spectroradiometer*) that takes images at a resolution from 250m to 1Km. Terra and Aqua pass through a particular area twice a day. The images provided by these satellites can be altered by clouds or the smoke produced by the fire. These factors may make the task of interpreting the perimeter difficult. This is not an automatic process; it is performed by expertise staff that try to define the perimeter with maximum precision, based on their knowledge and experience. In Figure 4.7, an example of images from MODIS at different time steps can be seen, as well as the corresponding perimeters processed by expertise staff. For this task, we rely on experts from the Joint Research Centre (JRC), an institution under control of the European Commission that studies forest fires in Europe - among other topics -, and develops and updates the European Forest Fire Information System ([62]).

Their research in this field covers the following topics:

Fire Danger Rating

Active fire detection from remote sensing

Rapid damage assessment from remote sensing (medium spatial resolution imagery)

Detailed fire damage assessment from remote sensing (high-spatial resolution imagery)

European Fire Database and fire statistics

Post-fire soil erosion

Long-term forest fire risk

Analysis of post-fire severity from remote sensing

Analysis of post-fire vegetation recovery from remote sensing

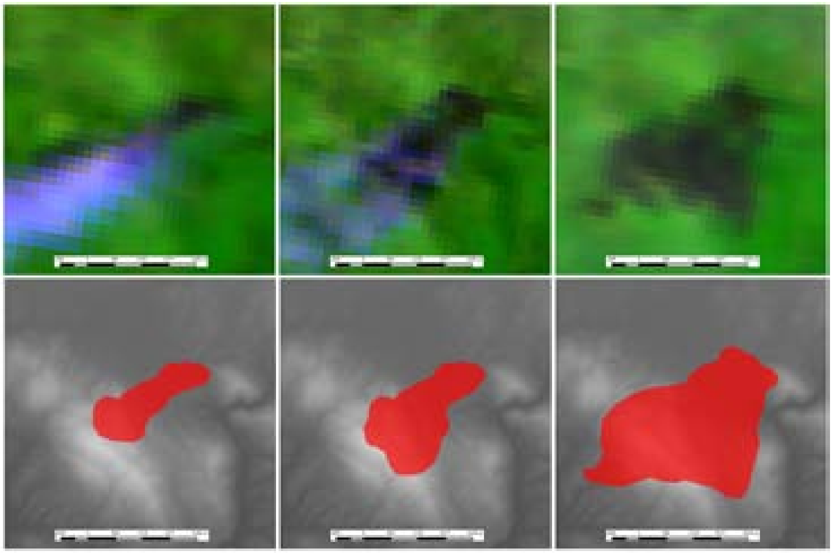

Figure 4.7: Images from MODIS instrument (top) and perimeters defined by JRC expertise staff (bottom).

In some cases, when we deal with fires that take place in Catalonia, we rely on data provided by the local government, the emergency management staff and its forest agents. They provide perimeters coming from routes on foot with GPS and aerial images.

### 4.2.2 Meteorological variables

Meteorological services such as the European Centre for Medium-Range Weather Forecasts (ECMWF) or the Meteorological Service of Catalonia (SMC [63] ) provide predictions of weather parameters, but the resolution is low (normally about 4 Km). In some cases, they use additional models to downscale these values. Both services, the European Centre and our local service, use the Weather Research and Forecast model (WRF) ([54]) to generate their predictions. The output of this model can be introduced in another model such as Calmet ([64]), which is capable of producing high resolution output (about 400m).

The terrain features influence wind behavior, so, sometimes, when topography is rough and abrupt, a 400m resolution may be inadequate. In these cases, we can add another diagnostic model only for wind components in order to reach higher final resolutions.

The meteorological services carry out the execution of weather forecast models (WRF and Calmet) every day for many purposes. We profit from their output and adapt it to our system. In addition, if the scenario requires more accurate wind input, we carry out the WindNinja executions to improve the resolution.

## 4.3 Fire simulation kernel

The fire simulation kernel is the core of the system and is composed of the fire simulator and various submodules responsible for building the data structures required to launch the simulation and configuring the execution parameters. The fire simulator has been widely discussed in previous chapters.

The integration submodules receive the data coming from other modules of the FFSS prediction framework, mainly from the module that implements the different prediction schemes and the calibration process. This module will be shown in the next section. The GA individual settings must be converted to the weather, wind, fuel moistures and fuel adjustment files. Depending on the prediction scheme chosen, these files will also receive information coming from the dynamic data sources instead of the calibration process. In addition, the static data will be packed in a landscape file, which collects all the topographic features.

The execution parameters are common to all the individuals, and they are extracted from the data initialization file. These settings only change from the calibration stage to the prediction stage, where the ignition perimeter, the duration and other simulation parameters are different.

## 4.4 Prediction schemes and Complementary Models

This part of the framework has been widely explained in previous chapters, and we will only detail the specific features that are required to include the schemes and the models in an operational framework. The prediction schemes and the complementary models allow the system to perform more accurate predictions, injecting real knowledge and high resolution data.

In fact, there are only two modules that do the required actions to perform the calibration stage using a certain prediction scheme. The main module is the calibration kernel module. It contains the calibration algorithm responsible for tuning the parameters and delivers the individual that will take part in the prediction stage.

The second (and last) module is the calibration setup module. It is responsible for reading the settings of the initialization file and collecting the input parameters and files that will feed the calibration process. Depending on the scheme selected, it will build the required files using information coming from the individual parameters, or produced by the complementary models.

Besides this task, when the scheme selected requires the generation of the wind fields for every individual, it will provide the executable path of the wind field simulator to the fire simulation kernel. If the selected scheme requires the generation of a set of wind fields corresponding to the forecasted weather conditions, the module will schedule the execution of these wind fields in the platform. When the module receives the generated wind fields, it will give the control to the calibration module.

# Chapter 5

# Experimental Results

This chapter is intended to show the experimental results that we have been performing throughout the evolution of this work. One of the main goals was testing the benefits of including complementary models in the original two-stage prediction method. In addition, we wanted to build an operational framework that would allow us to reproduce real forest fire scenarios, taking into account the specific features of this kind of fire. To be able to simulate and predict real fires, it was first necessary to test the framework with simpler cases that validated our prediction strategies.

The calibration method was a Genetic Algorithm (hereinafter, GA), although the number of genes involved and the number of generations and individuals have been increased as the complexity of the problem has grown. From now on, we will call the prediction scheme implementing the two-stage prediction method 2ST-BASIC, and 2ST-WF will be the scheme that includes the wind field model with the 2ST-BASIC. Likewise, the 2ST-MM includes dynamic weather data injection from weather stations and weather forecasting models and, finally, the 2ST-WF-MM is the complete scheme that encompasses both models.

The initial experiments were focused on analyzing the potential benefits of including a wind field modeler in the original system. These tests were developed using a fire evolved with the simulator in a synthetic terrain as comparison, with weather conditions adapted to our requirements. In parallel, using the same synthetic terrain, we carried out a set of experiments to check the system behavior when weather data was introduced during the simulation. In this case, the conditions with which the reference fire was created were dismissed, and we compared the results of the 2ST-BASIC with the ones introducing weather data (2ST-MM), resulting from introducing distortions to the real conditions.

After this, we performed a set of experiments replacing the synthetic terrains with real terrains. The objective was to use complex and rough terrains to better see the wind behavior over these terrains and to compare the introduction of wind fields in the fire simulator with the global homogeneous wind case. We continued using synthetic fires and the conditions that may be generated by us or come from stations near the terrain studied to introduce more realistic conditions. In the same experiments, besides testing the behavior of the 2ST-WF scheme compared with the 2ST-BASIC, we completed the experiments performing the 2ST-MM and 2ST-WF-MM cases. Thereby, we tested all the schemes in terms of quality.

This work has been mainly focused on evaluating and improving the quality of our predictions, but we show some experiments oriented towards showing the impact of the inclusion of complementary models in the original system. This is a key problem that is being tackled in our research group, but it is not the main objective of the present work.

Finally, we present our first results experimenting with real forest fires. These scenarios have great complexity due to the high uncertainty in the input parameters. Despite this, the results show that we can achieve good predictions, and new research lines can be opened to solve the new problems that appear in this kind of fires.

## 5.1 Experiments using synthetic fires

### 5.1.1 Early experimentation

The initial experimentation was aimed at trying out the inclusion of a wind field model in the two-stage prediction scheme. In that moment, the simulator used was Firelib, a much simpler fire simulator than Farsite.

These first tests correspond with an early experimentation to compare the 2ST-BASIC with the 2ST-WF under certain conditions. The main restriction was not having the wind simulator kernel uncoupled from the graphical interface. This made us reduce the number of individuals in the calibration stage, because the wind field calculations must be done between generations. As it is impossible to automate the process, the solution was to reduce the number of possible solutions to evaluate. Obviously, this restriction produced a negative impact on the GA. The populations tended to homogenize generation after generation, and thus increased the probability to fall to a local minimum.

The GA settings chosen for these experiments were an elitism of two individuals, a crossover rate of 20% (0.2) and a mutation rate of 1%. A population of 10 individuals has been generated that will evolve over 5 generations. The error of the best individual of this population in the calibration and prediction stages has been calculated. To calculate the errors, it is necessary to compare it with a reference fire. In this case, we built a synthetic fire (based on ideal values of wind) over a synthetic terrain with uniform vegetation consisting of shrubs (fuel model 7).

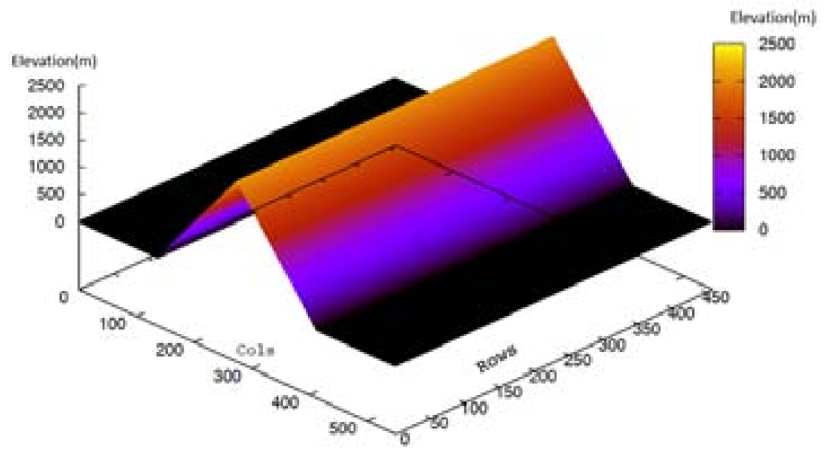


Figure 5.1: Synthetic reference fire

This terrain can be divided into four parts. The first and last part are flat areas with no slope. The second and third are opposite, with an upslope and a downslope of 50%. The initial ignition perimeter was located in the middle of the first part.

The wind conditions of the reference fire and the corresponding perimeters are depicted in Fig. 5.2. Five intervals (0-6h, 6-12h, 12-18h, 18-24h, and 24-30h) had been defined. The wind was set uniform (7mph and 315 degrees) during the three first intervals. At 18 hours, a sudden change in wind conditions (15mph and 270 degrees) was introduced that was maintained until the simulation ended.

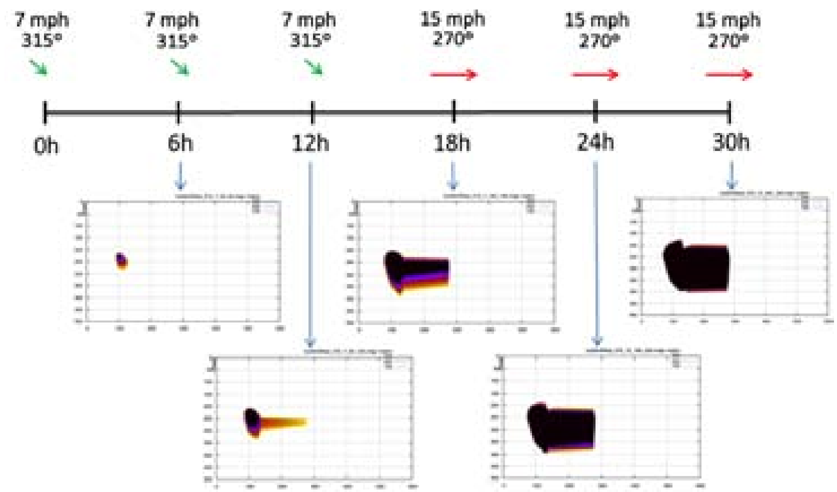


Figure 5.2: Synthetic fire conditions

The propagation results provided by each strategy have been compared using the error function stated in Equation 5.1, which evaluates the symmetric difference between cell maps (simulated map and real map). Each element of the equation is expressed in a number of cells. If the number of cells of the initial fire is considered negligible, the error is the number of cells that belong to the union of the maps minus the cells of the intersection, both of them divided by the number of cells of the real fire. This value is not within the interval 0-1, but the error can be greater than 1.

$$Error = \frac{(Cells(\cup) - Cells(ini)) - (Cells(\cap) - Cells(ini))}{Cells(Real) - Cells(ini)} \tag{5.1}$$

The objective was to find the best calibrated individual at 6, 12, 18, and 24 hours and to provide a prediction using these individuals for 12, 18, 24, and 30. Each calibrated individual at a certain time instant provided the prediction for the next interval. The last calibrated individual (corresponding to calibration stage from 18 to 24 hours) was used to produce the prediction at 30 hours and was also used to supply a prediction at 36 hours.

The calibration errors for each interval are shown in Fig. 5.3

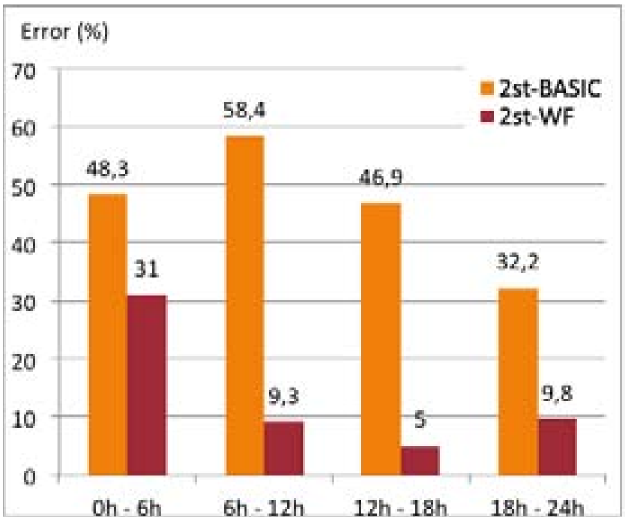


Figure 5.3: Calibration stage errors

As a rule, calibration errors were significantly less using the 2ST-WF scheme and, despite an interval with a considerable error (probably a local minimum), the errors were under 10%. In the 2ST-BASIC case, these errors were quite a bit higher, with the best around 30%.

Focusing on the prediction stage (see Fig. 5.4), although in the first interval the 2ST-BASIC was slightly better, in the following intervals, 2ST-WF achieved better predictions. The behavior of the schemes was also noteworthy when the wind conditions suffered a sudden change (at 18 hours). In this particular case, we observed that two-stage methods could not adapt well to this change, reaching high errors. Despite this, 2ST-WF was more capable of smoothing the error when compared with 2ST-BASIC. In the next intervals, the conditions remain stable, and both methods decrease in error, from

216% to 35.5% for the 2ST-BASIC, and, in the case of 2ST-WF, from 70% to 11.1%. In addition, both methods delivered good predictions at 36 hours (using the calibrated individual from 18 to 24 hours) due to the stability of the wind.

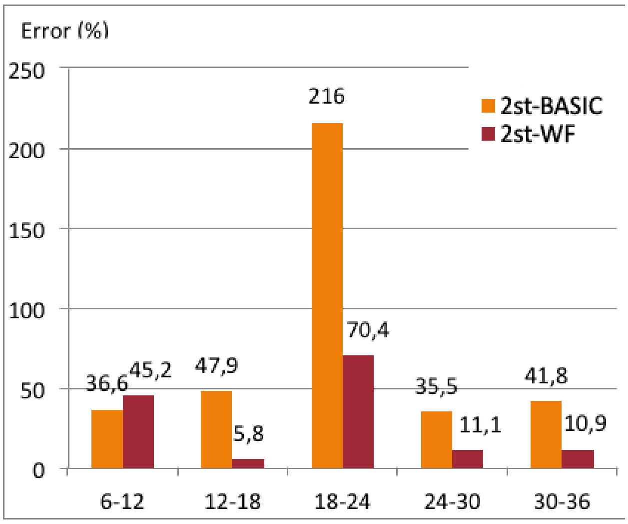


Figure 5.4: Prediction stage errors

From this experiment, we could draw some conclusions. As we expected, the population was too small, and, in future experiments, we must increase the number of individuals. The mutation rate was not enough to avoid population homogeneity. Therefore, our main goal was to uncouple the wind field modeler kernel from its graphical interface in order to be able to perform larger batch experiments.

### 5.1.2 Dynamic injection of meteorological data

In parallel to the work commented in the last paragraph, there were other lines on which we focused our efforts. First, we opted to change the fire simulator used until that moment (Firelib) for a more comprehensive, widely used and validated simulator such as Farsite. Using this new simulator, we carried out an experimental study to analyze the prediction quality when we added a weather forecast system to the 2ST-BASIC system.

In that case, we used a population size of 50 individuals, and the number of iterations was set to 10. In order to avoid random effects, each experiment has been repeated ten times, and the results shown below are the mean value of the corresponding ten results. As an experiment fire to test our proposal, we used the same synthetic terrain, fuel model and ignition point as in the early experimentation. The wind of the experiment fire varied from 3 to 15 mph in both experiment fires used as benchmarks. The ignition point was placed in the first region, and, due to wind conditions, fire propagated towards the second region.

The fire propagated over 8 hours, and wind direction and wind speed were changed every 30 minutes. The calibration stage considered the time interval from hour 0 to 4, and the prediction stage predicted the fire behavior from hour 4 to 8. In particular, we were interested in showing the benefits of predicting very dynamic parameters such as wind speed and wind direction when the working hypothesis for the two-stage prediction method was not accomplished. For this purpose, each of the two reference fires fell into one of these categories:

- Scenarios with homogeneous conditions: in this set of scenarios we introduced a slight variability in wind parameters from the calibration stage to the prediction stage in such a way that the working hypothesis was accomplished;

- Scenarios with heterogeneous conditions: in this set of scenarios, sudden changes were introduced both in wind speed and wind direction during the prediction stage, and, therefore, the working hypothesis was broken.

In both cases, we carried out three different kinds of experiments. The first kind of experiment was the 2ST-BASIC experiment (Experiment 1). In this scheme, the wind conditions, moisture values and fuel conditions were introduced as genes in the individuals of the GA population. Therefore, the wind speed and wind direction were calibrated by the GA as for the other parameters. During the time interval $t_i$ - $t_{i+1}$, the values of the parameters were considered constant. The calibrated values provided by the GA for all the parameters were then used as input parameters for the prediction stage during the time interval $t_i$ - $t_{i+1}$.

In the second kind of experiments (Real data assimilation Experiment 2), the wind conditions were not calibrated, but their measured values were assimilated dynamically at each subinterval (1 hour) in the simulations of the calibration stage. We considered that wind speed and direction were measured every 60 minutes instead of the 30 minute real wind evolution. In real wildfire cases, the wind data frequency depends on the meteorological data sources of the studied zone. These measured values were not the same as the ones that were used to generate the experiment fire, but these values had some measurement error. So the parameters that were calibrated by the GA were the moisture parameters and fuel conditions. In the prediction stage, the calibrated moisture and fuel parameters and the last measurement of the wind parameters were used. So, the prediction was based on a single measured value for the wind conditions.

The third kind of experiment was the model coupling experiment (Experiment 3). In this case, the calibration stage behaved like in Experiment 2. The wind conditions are assimilated dynamically, and the moisture and fuel parameters are calibrated. However, for the prediction stage the wind conditions were provided by an NWP model such as WRF. This experiment corresponds to the 2ST-MM prediction scheme. The synthetic fire of the experiment considered that the wind conditions changed every 30 minutes. For testing our approach, we assumed that the NWP model provided values that have a small deviation from the ones of the fire used as a benchmark. So, we were not injecting the real value of the wind conditions, but we were injecting certain perturbation on these values. The range of the perturbation was generated considering the statistical behavior of weather predictions.

Table 5.1 summarizes how each one of the three kinds of experiments managed the most sensitive input parameters such as wind speed and wind direction and the four moisture components: moisture content of dead fuel at 1 hour (M1), moisture content of dead fuel at 10 hours (M10), moisture content of dead fuel at 100 hours (M100) and moisture content of live fuel

| Prediction scheme | Inputs | Calibration Stage | Prediction Stage |
|---|---|---|---|
| **Experiment 1: 2ST-BASIC** | Wind | Random Values | Calibrated values |
| | Fuel Moisture | Random Values | Calibrated Values |
| **Experiment 2: Real Data Assimilation** | Wind | Real Data Sampling | Real Unique Value |
| | Fuel Moisture | Random Values | Calibrated Values |
| **Experiment 3: Models Coupling** | Wind | Real Data Sampling | Forecasted Values |
| | Fuel Moisture | Random Values | Calibrated Values |

Table 5.1: Settings of wind, moisture and fuel characteristics input parameters for each experiment.

(Mherb).

Figure 5.5 shows the time evolution of wind speed and wind direction considered in the experiment synthetic fire and in the three experiments for the homogeneous scenarios. Figure 5.6 shows the time evolution of wind speed and wind direction considered in the experiments for the heterogeneous scenarios. It is important to note which scenarios are homogeneous which ones are heterogeneous in this work. In homogeneous scenarios, the average of the values in the calibration stage is similar to the average in the prediction stage. Although the values of wind components of the experiment fire in Figure 5.5 are very irregular, the average of these values at every stage is almost the same. In heterogeneous scenarios (see Figure 5.6), the difference in the average of these values between stages is greater than in homogeneous conditions.

The quality of each approach was measured by an error function. The error function considered is the symmetric difference between the real burned area and the predicted burned area. Optimally, the real and the predicted burned area coincide and the symmetric difference is 0. The following subsections show the experimental results for the homogeneous and heterogeneous condition scenarios.

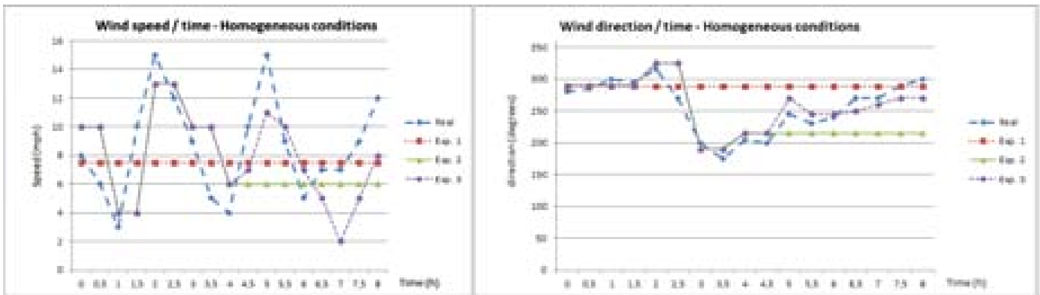


Figure 5.5: Wind speed and direction considered for the different experiments in the scenario with homogeneous conditions

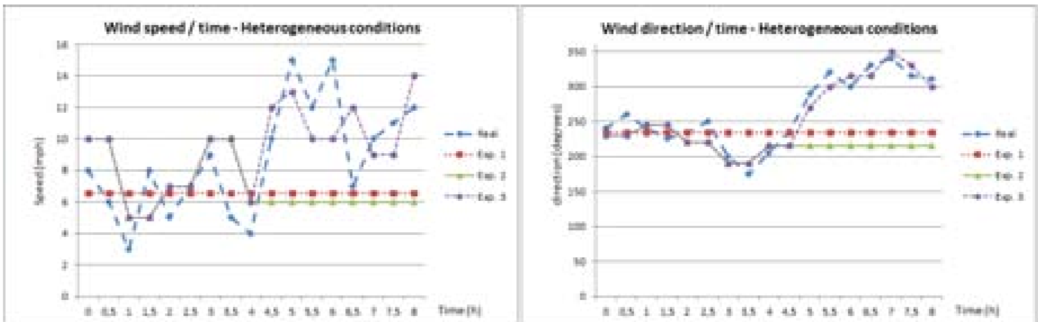


Figure 5.6: Wind speed and direction considered for the different experiments in the scenario with heterogeneous conditions

### Homogeneous conditions

The experiment settings correspond to a wind speed and wind direction samplings with low variability between the calibration and the prediction stage. The average of wind speed values in the calibration stage was 8 mph and 9.25 in the prediction stage. In the case of wind direction, the average was 258.3 degrees in the calibration stage and 255.6 in the prediction stage. As we can observe, the conditions were almost the same on average. Under these favorable conditions, the 2ST-BASIC prediction method worked the best. Figure

5.7 represents the mean errors obtained in the three described experiments for the calibration and prediction stages.

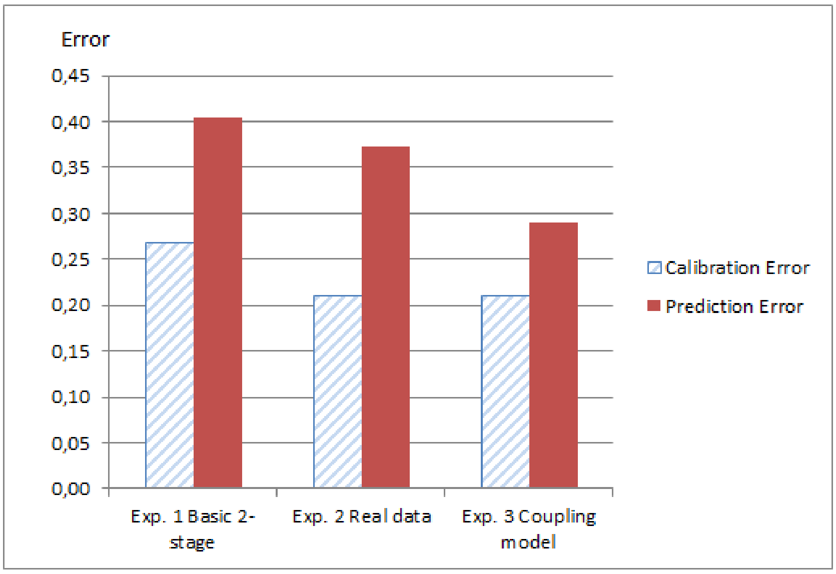


Figure 5.7: Calibration and prediction mean errors for the experiments under homogeneous conditions

It can be observed in Figure 5.7 that the calibration and prediction errors were quite similar in the three experiments. This was the expected behavior since the wind conditions were quite stable during the whole calibration and prediction stages, and, then, a single value could represent the wind behavior quite successfully. However, even in the favorable case, the results showed a tendency to reduce the error when the weather prediction model was introduced and the wind values for the prediction stage were considered.

### Heterogeneous conditions

In this case, wind conditions (speed and direction) were not constant, but they presented a higher degree of variability between stages. The average of wind speed values in the calibration stage was 6.1mph and 11.5 in the prediction stage. This average, in the case of wind direction, was 225.4 degrees in the calibration stage and 305 degrees in the prediction stage. It means that the wind conditions were significantly different in the calibration and prediction intervals. Figure 5.8 presents the mean error values for the three considered experiments in the calibration and prediction stages.

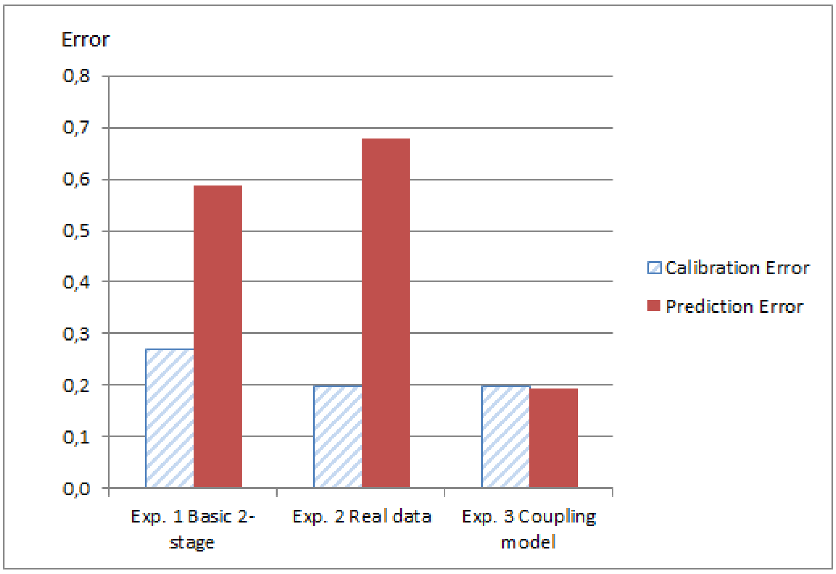


Figure 5.8: Calibration and prediction mean errors for the experiments under heterogeneous conditions

It can be observed that the calibration error was quite similar in the three experiments, because the wind conditions were more or less stable during that interval. However, since there was a sudden change in the wind conditions during the prediction interval, the prediction error was significantly different. Experiment 3, where the wind conditions during the prediction stage were injected from the predictions of a NWP model, reduced the error significantly. Experiment 2, that considered the wind conditions at time $t_{i+1}$ for the whole interval $t_i$ - $t_{i+1}$, produced the worst prediction results since the wind conditions considered were not representative for the time prediction interval.

## 5.2 Experiments using synthetic fires over real terrains

### 5.2.1 Coupling the wind field model in the 2ST schemes

The proposed enhanced alternatives for the basic two-stage prediction scheme are oriented to overcome two weak points: the uniform distribution of the parameters throughout the terrain and the incapacity of reacting to sudden changes in environmental conditions. Therefore, in order to validate the improvements introduced in the two-stage basic methodology to overcome those deficiencies, we have set up an experiment which reproduces the problems we wanted to solve.

The terrain where the experimental study is performed is called Cap de Creus , which is located in Catalonia (north-east of Spain). This zone has been selected for its irregularity in terms of slope variations and also because it is an area of interest due to the number of times it has been affected by forest fires over the last decade. Although we use the real values of elevation, slope and aspect of this terrain, there is a lack of information about fuels (vegetation types) and canopy cover. For this reason, we use a homogeneous fuel (number 7 in Rothermel & Albini classification) and a fixed percentage of canopy cover (25%).

In order to evaluate the quality of the enhanced methodology when dealing with sudden changes in certain meteorological variables, we create a reference fire with certain meteorological conditions. In particular, wind speed and wind direction present strong variations from the calibration stage to the prediction stage. The whole fire lasts 18 hours, and the components of the wind (wind speed and wind direction) vary every 30 minutes, as can be observed in Figure 5.9.

Taking into account that the typical time-step of coarse scale weather forecast models ranges from 3 to 6 hours, the time window selected for the calibration stage and the prediction stage was 6 hours each. Therefore, over the 18 hours, we are able to repeat the whole prediction scheme twice. That is, the first calibration stage goes from hour 0 to hour 6 and, the corresponding prediction stage goes from hour 6 to hour 12. Then, the second execution of the whole prediction method implies that the calibration stage goes from hour 6 to hour 12 and, the prediction window goes from hour 12 to hour 18.

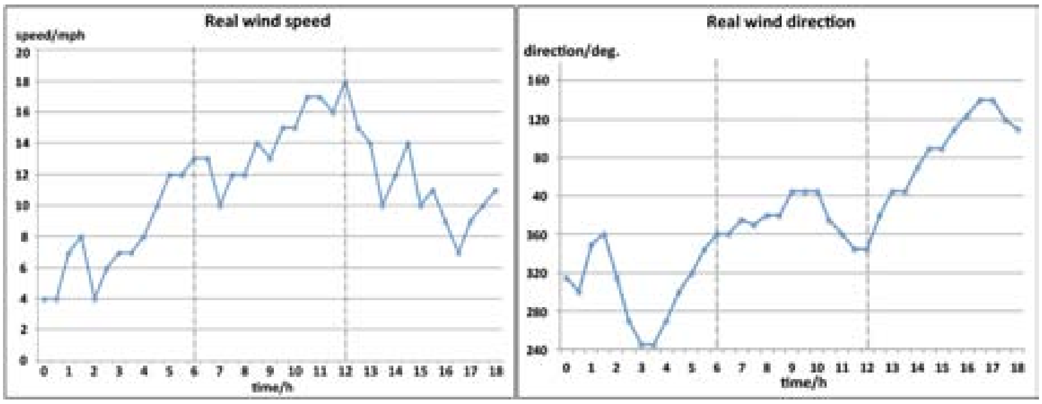


Figure 5.9: Wind speed and direction trend corresponding to reference fire

Since the calibration strategy applied in this case is GA, which consists of a stochastic optimization strategy, every kind of experiment was repeated ten times with ten different populations of 25 individuals. Thus, the results reported in this section are the mean values of those 10 experiments. The predicted data injected in the case of using a meteorological model is obtained by simulating the behavior of that model introducing a perturbation error at the reference fire data and injecting that perturbed data to the fire prediction system. In the experiments described below, the error in the case of wind speed is about 2.5 mph on average, and, for the wind direction, the error has been set to values greater than 20 degrees. It is noteworthy that it is not a constant error, and, in some phases, the error is greater; while, in other phases, observations and predictions are closer to reference fire conditions.

In the following sections, we analyze each iteration of the whole prediction scheme separately, in order to better understand the results obtained when applying each one of the four above-mentioned enhanced prediction schemes compared with the basic two-stage strategy.

### Calibration from 0 to 6 hours and prediction from 6 to 12 hours

The results obtained for the calibration and prediction stages are shown in Figure 5.10. As can be observed, the 2ST-BASIC and 2ST-WF approaches are the schemes that, although they provide a good calibration error, the prediction error is quite high. This effect is not observed when applying either the 2ST-MM or the 2ST-WF-MM schemes. The reason for that is that wind suffers sudden changes from the calibration stage to the prediction stage (see Figure 5.9). The 2ST-BASIC and the 2ST-WF rely on the wind values provided by the calibration stage, so they are not able to react to those changes. However, 2ST-MM and 2ST-WF-MM use forecasted wind data at the prediction stage; therefore, these strategies are able to cope with those wind changes.

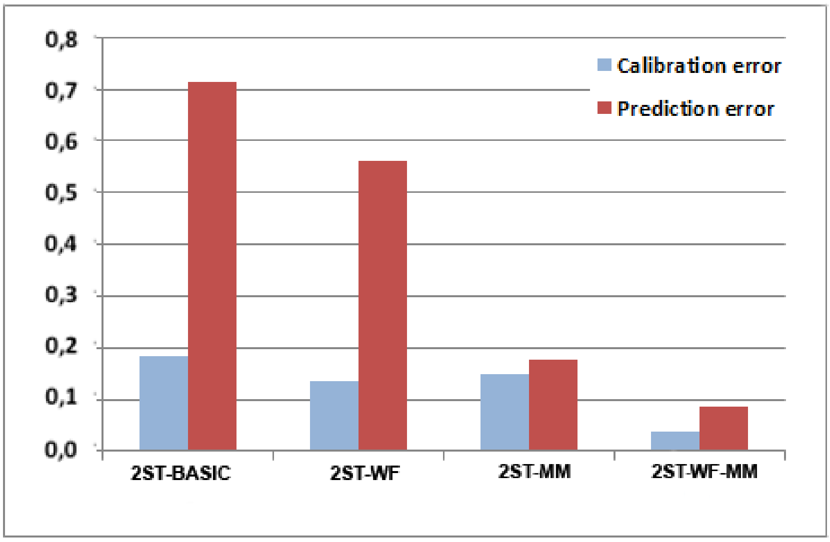


Figure 5.10: Calibration and prediction errors from 0 to 6 hours in calibration and 6 to 12 hours in prediction stage

To analyze the influence of coupling a wind field model with the prediction scheme, we might analyze the results provided by 2ST-BASIC and 2ST- -MM compared with 2ST-WF and 2ST-WF-MM. As can be observed, prediction errors including a wind field model are clearly better than not considering the influence of the terrain features in the wind components. The relevance of this effect can be better observed in Figure 5.11. This figure depicts the prediction results for all prediction schemes. In particular, each frame shows four fire perimeters. Three of them are the same for all frames: the perimeter at hour 0, the spread fire front at hour 6 and the real fire spread at time hour 12. The fourth perimeter is the predicted perimeter provided by each scheme at hour 12. Analyzing the images in detail, we can see that, when the wind field model is included, the fire front obtained better fits the underlying topography because wind speed and wind direction are not considered homogeneous values but vary according to the terrain heterogeneity.

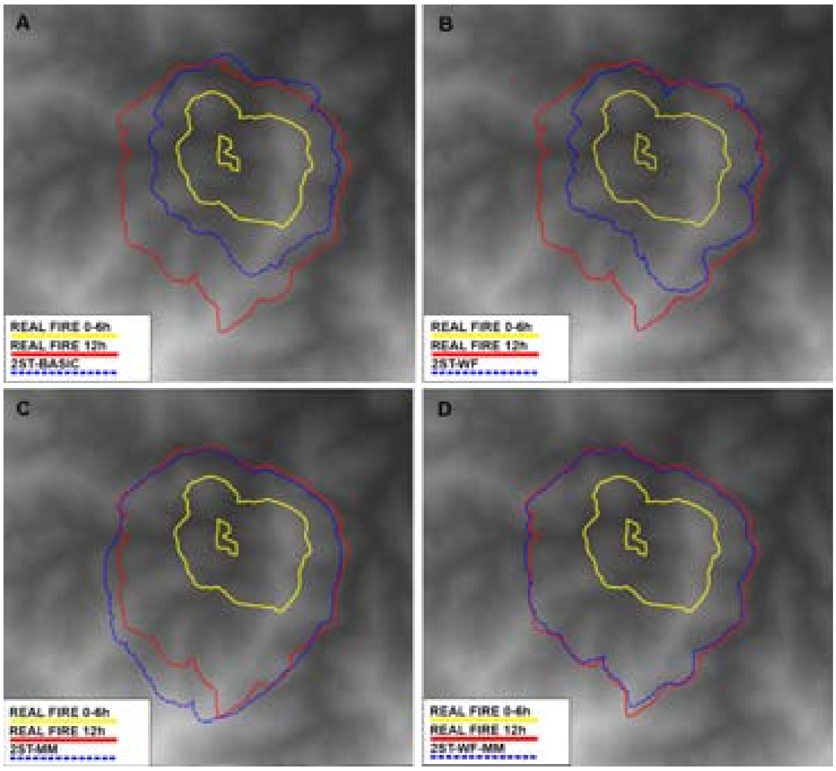


Figure 5.11: a.2ST-BASIC vs Real / b.2ST-BASIC-WF vs Real / c.2ST-BASIC-MM vs Real / d.2ST-BASIC-WF-MM vs Real

Finally, when considering the scheme where both prognostic and diagnostic models are coupled with the forest fire spread model, the prediction spread errors on average denote a great reduction.

### Calibration from 6 to 12 hours and prediction from 12 to 18 hours

In this section, we analyze the results obtained after finishing the second iteration of the whole prediction process for the four schemes studied. In general, the results in terms of quality improvements denote a similar trend to the ones reported in the previous section. In particular, in this calibration-prediction step, meteorological conditions also suffer notable changes from one stage to another; however, those changes are not as abrupt as in the previous experiment. In particular, wind speed has been set to have a downward trend. Under these conditions, the 2ST-WF-MM approach is the best both at the calibration stage and at the prediction stage in terms of error delivered error. Despite this, this softer change in conditions allows 2ST-BASIC and 2ST-WF strategies to reduce their prediction errors significantly.

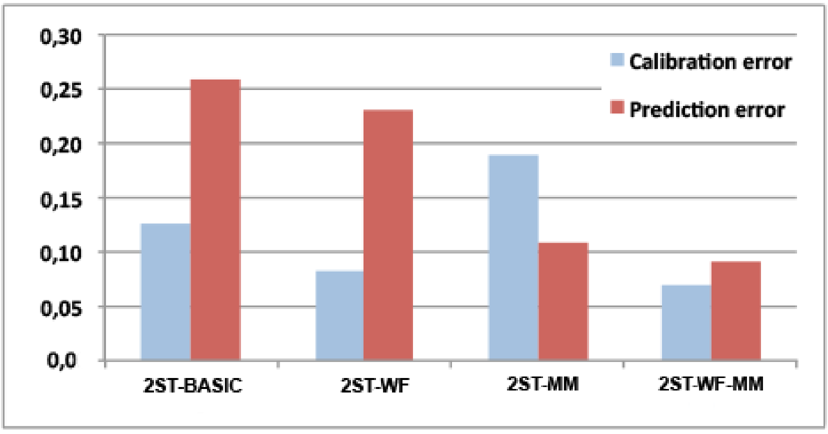


Figure 5.12: Calibration and prediction errors from 6 to 12 hours in the calibration stage and 12 to 18 hours in the prediction stage

## 5.2.2 Quality and time analysis of 2ST schemes

In this experimentation, we have chosen an area in Catalonia (north-east Spain) where a big fire occurred in 2012. This fire started on July 22 and burned about 13,000 ha., causing 2 deaths. This location (*La Jonquera*) is in the north of Catalonia, and we have used a map of 33Km x 48Km.

This map has been obtained from ASTER, a NASA satellite that takes high-resolution images of the earth (up to 30m). These images have been processed, and we have extracted raster files with the information needed to perform our simulations (elevation, aspect and slope). We have also used the vegetation information provided by the Corine Land Cover project.

Figure 5.13 shows the occurrences of fires bigger than 30 hectares in the period between 1980 and 2012 in Catalonia. As can be observed, one of the most recurrent areas corresponds to the north-east region, where La Jonquera is located. This area is zoomed on the right of the image, and the colors indicate how many times a cell (100m per 100m) has been burnt since 1980.

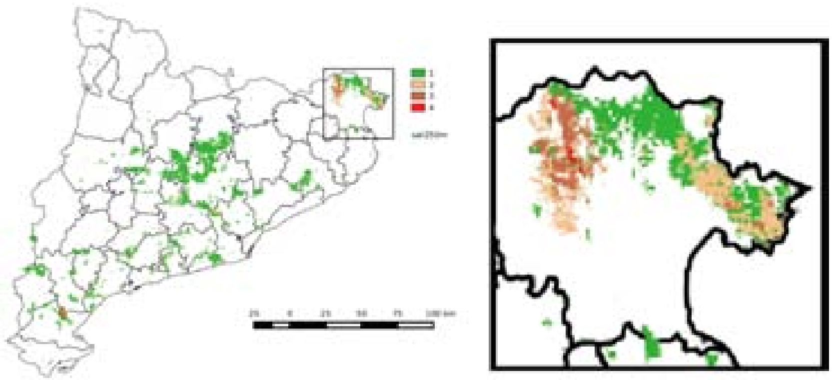

Figure 5.13: Fire occurrences in Catalonia (a) and Cap de Creus (b) during the period 1975-2010

In order to evaluate all proposed schemes, we created a reference fire that lasts 24 hours. The components of the wind (speed and direction) vary every 30 minutes and have been provided by the meteorological station of La Jonquera, which is located very close to where the fire took place. These values can be observed in Fig. 5.14 and Fig. 5.15. The resulting fire evolution is stored and used as a real fire evolution, and the input settings that were used to generate this propagation are dismissed. In this test case, the meteorological data, in particular the wind forecast, has been generated in a synthetically, introducing perturbations into the real wind behavior. Wind speed varies 4.49 mph on average, and, in some cases, the difference is up to 10 mph. Wind direction varies 21 degrees on average, and the highest difference reaches almost 90 degrees. This synthetic forecasted wind was the one used in the 2ST-MM and 2ST-WF-MM prediction schemes as the data to be dynamically injected during the prediction stage.

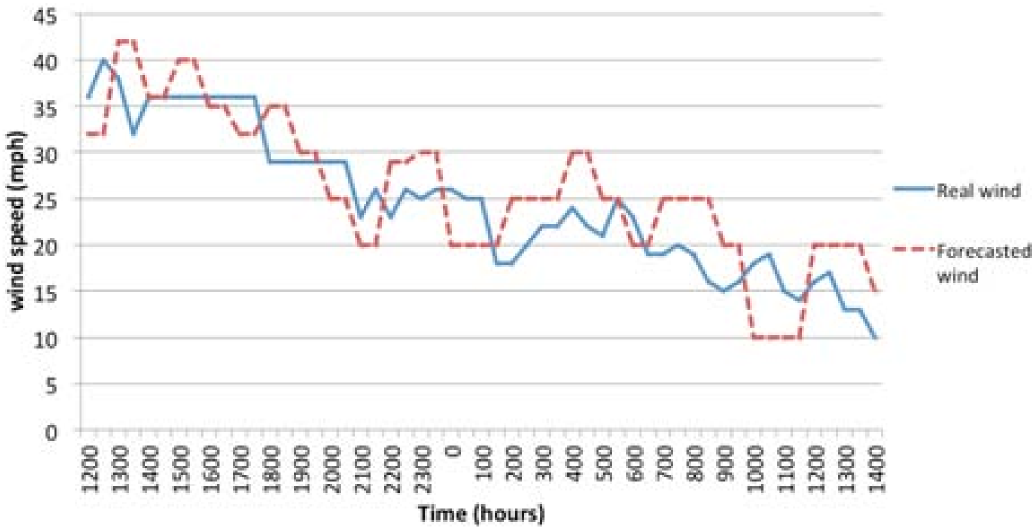


Figure 5.14: Real and forecasted wind speed evolution

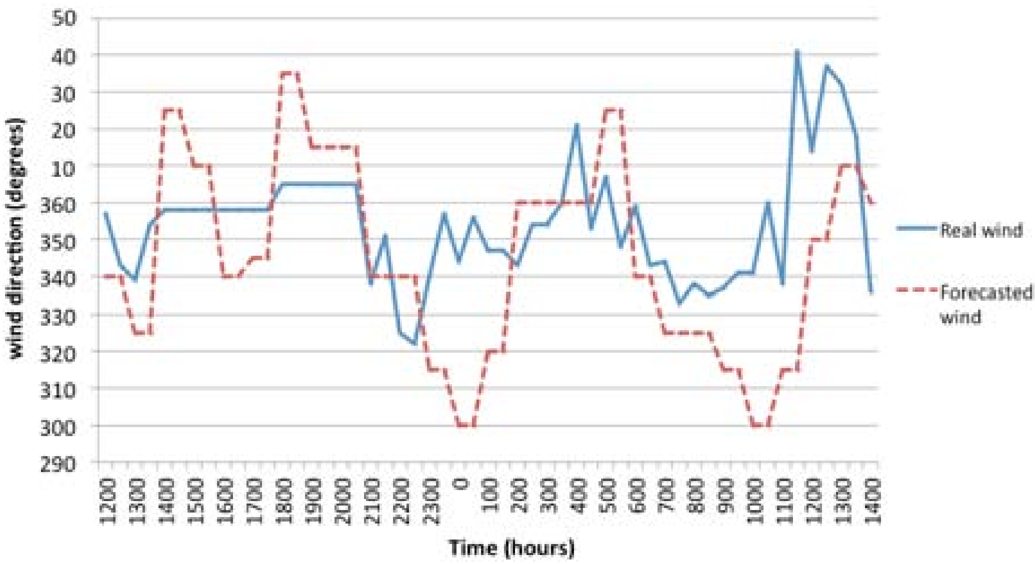


Figure 5.15: Real and forecasted wind direction evolution

As can be seen in Figure 5.16, the real fire propagation obtained for the fire analyst once the incident ends (yellow shape) and the reference fire that we have obtained using the synthetic forecast wind (the external red line) are very similar. It must be taken into account that the reference fire propagation has been obtained without considering mitigation actions that, obviously, were done and had a direct impact on the perimeter s shape. In other words, the reference fire is a free burning fire, which allows the prediction framework to forecast the potential danger of the fire.

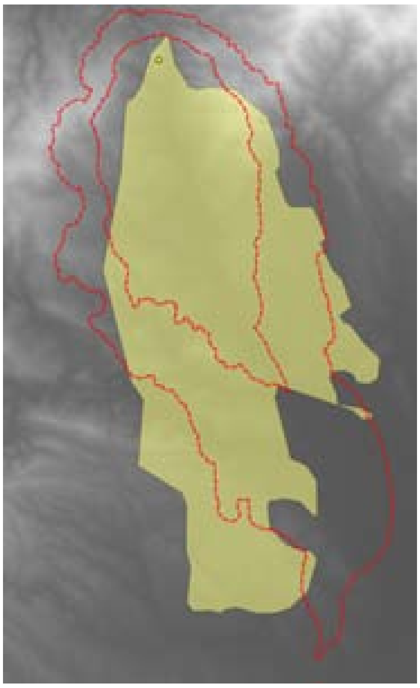

Figure 5.16: Real observed fire spread versus reference fire perimeter

The time window selected for the calibration and prediction stages was chosen taking into account the time period needed to gather useful information from weather forecast services and satellite sensor systems. In the former case, the typical time-step of coarse scale weather forecast models ranges from 3 to 6 hours, which determines the frequency of delivered data. In the latter case, we should consider the time interval required for receiving fire front images that could be properly used in that multi-model prediction framework. To obtain such perimeter information, we rely on sensor systems, which are on board both the NASA Terra and Aqua satellites. It is necessary for each satellite to complete 3 orbits (approximately 3 hours) to cover the whole European area, so it could be possible to obtain fire perimeters every 3-6 hours [65].

Since the calibration strategy implements a GA, we perform the experiment for different population sizes such as 25, 50 and 100 individuals in order to analyze the influence of this parameter on the results in terms of quality, execution time and efficiency. The GA has been iterated 5 times and, for each initial population size, we have performed 5 different experiments with different initial populations. Thus, the results reported in the following sections are the mean values of those 5 experiments.

### Parallel systems

Two different execution platforms have been used to perform the experiments reported in the following sections:

1. An IBM x3550 cluster with 32 compute nodes, where each node counts on 2x Dual-Core Intel Xeon CPU 5160, 3.00GHz, and 12GB Fully Buffered DIMM 667 MHz s. That means a total number of 128 cores and 384GB of main memory.

2. Two Dell nodes, where each node has 4 sockets with an AMD Opteron 6376, 2.30GHz processor with 16 cores each. So, every node has 64 cores and 128GB of main memory. That also means a total number of 128 cores.

These systems have been selected because they are small-medium size clusters that can be available to regional emergency services in real operation. In most cases, such services do not have Exascale systems with a large number of nodes available.

Throughout the rest of this work, we will refer to each one of the above-described platforms as IBM and DELL clusters.

### Quality analysis

The schemes incorporate a GA that iteratively improves the quality of the calibration. So, it is necessary to analyze the convergence of the GA and the influence of other parameters such as the population size in the quality delivered of every multi-model scheme. Fig. 5.17 summarizes the evolution of the error at each generation for all four schemes. It can be observed that the schemes that incorporate the meteorological model (2ST-MM and 2ST-WF-MM) do not require a significant number of generations to achieve an almost stable calibration error. The main reason for this behavior is that, in these cases, the wind is not a parameter to be calibrated, but is rather a measured or forecasted parameter. Therefore, the calibration process is easier since the wind speed and direction are very relevant parameters that must be calibrated carefully. In these cases, the population size does not appear to be a relevant factor since the errors are very similar.

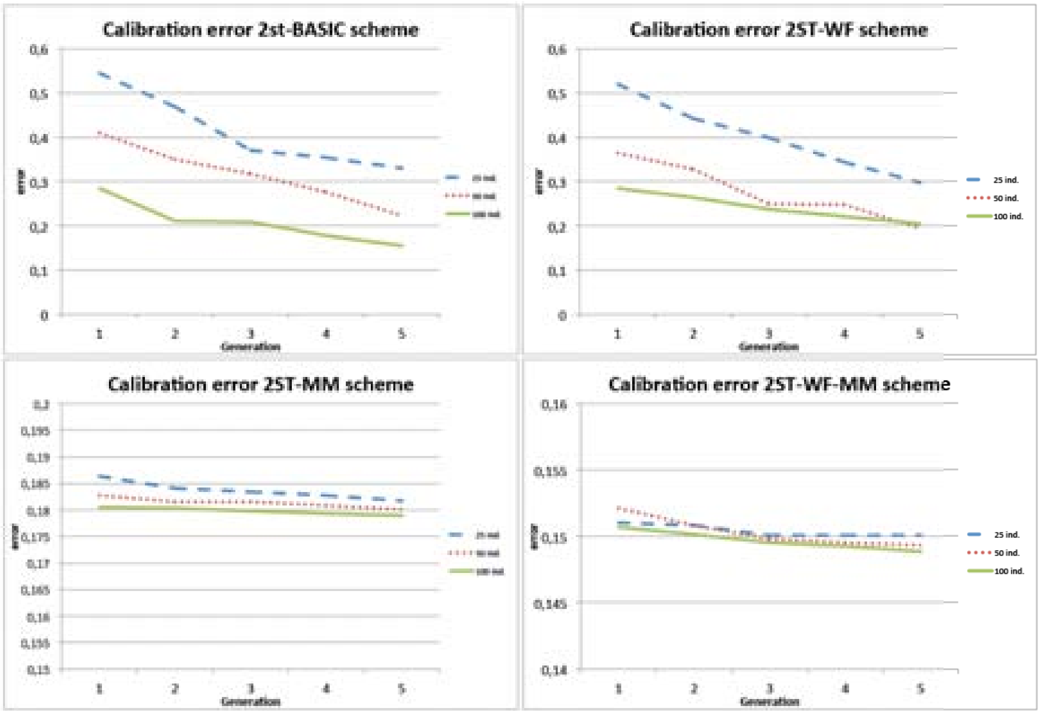


Figure 5.17: Calibration error evolution for 2ST-BASIC (a), 2ST-WF (b), 2ST-MM (c) and 2ST-WF-MM (d) using initial populations of 25, 50 and 100 individuals

The schemes that do not incorporate meteorological models (2ST-BASIC and 2ST-WF) must calibrate the meteorological wind value and, therefore, the calibration process requires more iterations. In these schemes, the population size is a relevant factor since larger populations provide better calibration results. It is also noteworthy to observe that the schemes that incorporate the wind field model (2ST-WF and 2ST-WF-MM)) provide better results than those that use a general value for the whole terrain. This means that the wind field model is an added value to the prediction process.

The calibration process is very significant, but, ultimately, the most relevant result is the prediction error. Fig. 5.18 shows the errors obtained at the end of the calibration process and the corresponding prediction errors. The same information is depicted in Figure 5.19 where a visualization of the delivered predicted fire front evolution for each multi-model scheme is plotted. As we can observe, the multi-model scheme that provides the best results in both the calibration stage as well as in the prediction stage is the 2ST-WF-MM, as expected. Let s analyze the results scheme by scheme in more detail.

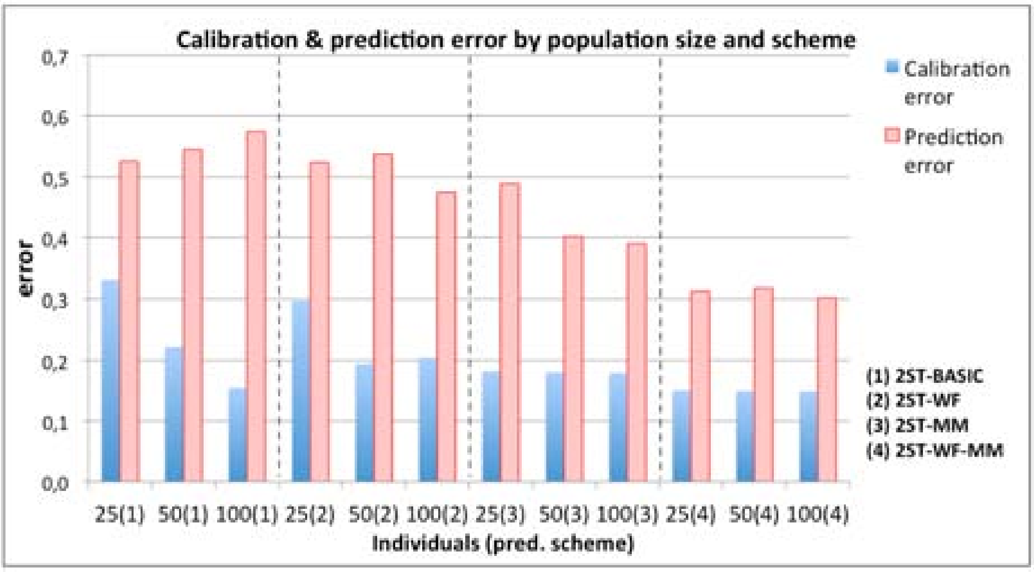


Figure 5.18: Calibration and Prediction errors for all multi-model predictions schemes using initial populations of 25, 50 and 100 individuals

2ST-BASIC is the scheme that delivers the worst results both in prediction and calibration errors. This scheme needs to perform all the iterations of the GA to reduce the prediction results, but, even iterating until the preset

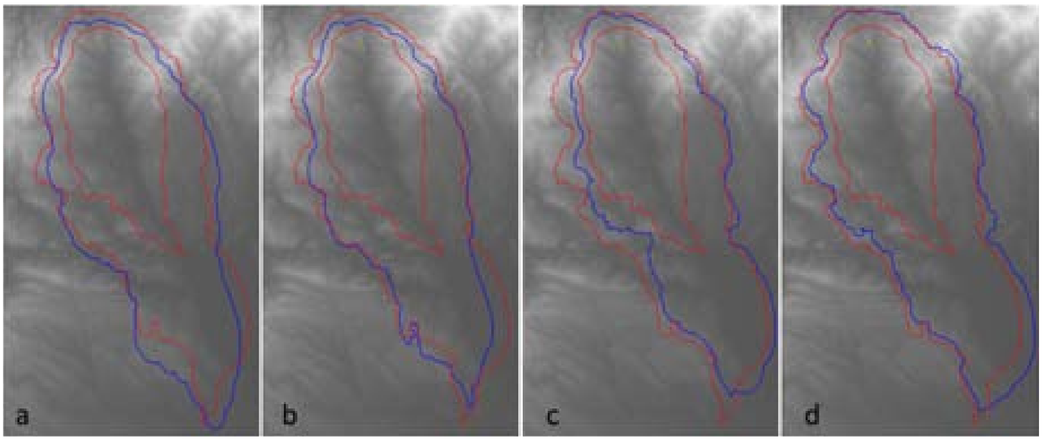


Figure 5.19: Best predicted fire front for each scheme: a) 2ST-BASIC b) 2ST-WF c) 2ST-MM d) 2ST-WF-MM

number of generations, it is not able to adapt to changes in the meteorological conditions such as wind parameters. The initial population size could slightly reduce this penalty, but, in general terms, this effect is independent of the number of the GA's individuals.

When observing the results provided by the 2ST-WF, we detected a quality improvement in the calibration stage. This enhancement is due not only to the inclusion of the wind field evaluation for each particular individual, but also to the ability to calibrate the general wind components. This ability enables the system to better adjust the wind parameters to reproduce more precisely the recent past behavior of the fire. However, this enhancement is not extrapolated to the prediction stage. As we can observe, the prediction error drastically increases despite having a good calibration error. The main reason for this is that wind values do not quite remain constant from the calibration stage to the prediction stage. Thus, the 2ST-WF scheme is not able to adapt to meteorological changes if those changes happens during the prediction interval.

To overcome this drawback, a meteorological model was coupled with the multi-model system. The advantages of including such a model are reflected in the prediction errors obtained for 2ST-MM and 2ST-WF-MM. Using forecasted data in the prediction stage helps the system to dynamically adapt to changes in environmental variables. However, 2ST-MM delivers worse results than 2ST-WF-MM. The reason for this difference is the ability of 2ST-WF-MM to provide high-resolution wind flow that better reproduces the wind variations at a field level. So, if the wind speed varies because of a mountain or a valley the 2ST-WF-MM multi-model scheme captures these effects. Those results remain quite similar for all initial population sizes, so, a last conclusion in this point could be that a population size of 25 individuals is enough to obtain reasonable prediction results. However, these conclusions must be contrasted with the execution time and efficiency results reported in the next section.

### Execution Time

Prediction quality is a very important issue, but execution time is just as critical as accuracy. So, it is necessary to study the execution time of each scheme and the efficiency reached. As an initial experimental platform, we use the IBM cluster running the MPI prediction scheme. The execution time of each scheme for different population sizes (25, 50 and 100 individuals) has been evaluated, and the results are shown in Table 5.2. As can be observed, the execution time of 2ST-WF is by far the most time-consuming multi-model scheme. The need to execute a wind field model for each individual at each iteration results in an increment of time. Fig. 5.20 summarizes the execution time of all schemes.

The execution time depends on the number of workers, the number of nodes on the system, the architectural features of each node and, more specifically, on the multi-model scheme selected and the particular scenario represented. We use 128 cores to perform our test. Since our populations are 100 individuals at most, there are enough cores to execute one worker per core, but the internal architecture of each node has a crucial effect. Each node has 2 dual-core processors and 12GB of memory.

Comparing the execution time obtained by the different schemes, it can be observed that 2ST-MM and 2ST-WF-MM are the fastest schemes. In the other two schemes, the calibration and prediction processes take longer.

| Population size | 25 | 50 | 100 |
|---|---|---|---|
| **2ST-BASIC** | 1169s | 2952s | 2240s |
| **2ST-WF** | 2022s | 3734s | 6892s |
| **2ST-MM** | 663s | 796s | 936s |
| **2ST-MM-WF** | 1027s | 1101s | 1103s |

Table 5.2: Execution time of every scheme and population size.

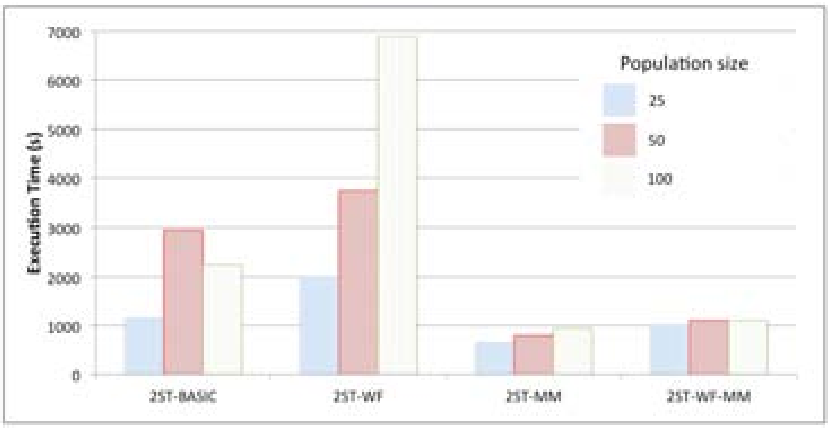


Figure 5.20: Execution time of 2ST-BASIC, 2ST-WF, 2ST-MM, 2ST-WF-MM for all population sizes

This is due to the fact that the individuals can contain wind values that are very different from the real values and take longer to be simulated. Such longer individuals do not appear in the schemes using dynamic wind injection (2ST-MM and 2ST-WF-MM).

The 2ST-MM scheme assumes that the measurements of the meteorological stations and the forecasted data provided by meteorological models are available before the prediction stage is launched. So this scheme does not introduce any additional computational cost. The complete multi-model 2ST-MM-WF scheme does not require the calibration of the wind parameters since they are measured or forecasted, and the calibration then requires less iterations. So the computational cost of introducing the wind field modeler does not suppose a significant time increase. The only overhead introduced is the one incurred when the fire simulator reads the set of precalculated wind fields which are necessary to carry out the simulation.

| Population size | Cores/individual | Calibration Time |
|---|---|---|
| **IBM Cluster** | 1 | 471s |
| **IBM Cluster** | 4 | 348s |
| **Dell Cluster** | 4 | 363s |

Table 5.3: Calibration time of the 2ST-WF-MM scheme and 25 individuals on different platforms.

Larger populations (100 and 50 individuals) need fewer generations to converge to a reasonable error in 2ST-BASIC and 2ST-WF schemes, but the execution time of each generation is larger. We can conclude that 25 individuals are enough to achieve good prediction errors in a reasonable time. 2ST-MM and 2ST-WF-MM are schemes that converge very fast, so a few generations are enough to reach a good prediction error. In these schemes, using larger populations does not significantly increase execution time, but the benefits of using more resources are poor.

Taking this into account, we have focused on 2ST-WF-MM scheme and the 25 individual population. Since our test systems have 128 cores each, it is possible to use four cores per individual in the calibration stage by applying the hybrid MPI-OpenMP implementation of the calibration stage. The calibration time (average of five executions) using four cores per individual in both systems can be seen in Table 5.3.

When using four cores to simulate each individual, WindNinja and Farsite can take advantage of such resources and reduce their execution time. So the overall calibration time is reduced 26% using four cores in the IBM cluster. The execution time obtained in the Dell Cluster is similar to the IBM one. Although the Dell Cluster has more modern and powerful processors, they have to share memory bandwidth among many individuals, and, in this kind of application, memory is a significant resource. However, the IBM cluster has 32 isolated nodes with 12GB of memory that are only shared among four cores (1 individual).

When the calibration stage has been carried out, the prediction taking the best individual from the calibration stage must be run. In this case, this prediction only implies the execution of one simulation. In 2ST-WF-

| **Prediction time** | **Serial time** | **2 cores** | **4 cores** | **8 cores** | **16 cores** |
|---|---|---|---|---|---|
| **IBM Cluster** | 607.53s | 455.19s | 451.23s | - | - |
| **Dell Cluster** | 641.05s | 492.37s | 400.71s | 366.49s | 340.21s |

Table 5.4: Prediction time using OpenMP in IBM and Dell clusters.

| **Total execution time** | **IBM Cluster** | **Dell Cluster** |
|---|---|---|
| **Best Calibration case** | 4 cores/individual | 4 cores/individual |
| **Best Prediction case** | 4 cores/individual | 16 cores/individual |
| **Calibration time** | 348.39s | 363.15s |
| **Prediction time** | 451.23s | 340.21s |
| **Total time** | **799,62s** | **703,52s** |

Table 5.5: Total time using OpenMP in IBM and Dell clusters.

MM, we have seen that the prediction stage lasts longer than the calibration stage. The execution of the best scenario is about 600s in all cases. In the prediction stage, the initial perimeter is the real perimeter at 12 hours, so the complexity of the simulation is greater than in the calibration stage (ignition is a single point). In this stage, we have executed these simulations in the IBM cluster with 1, 2 and 4 cores and in the Dell Cluster using 1, 2, 4, 8 and 16 cores. Prediction times are shown in Table 5.4.

It can be observed that the prediction stage time is reduced when using more cores in the DELL system. So a promising approach is to consider the best configuration for each stage. That is, on the one hand, in the calibration stage to exploit the hybrid two-level parallelization scheme using an MPI Master/Worker GA with a 25 individual population and four cores per individual, applying OpenMP. And, on the other hand, in the prediction stage, to exploit to the maximum the OpenMP parallelism using four cores in the case of the IBM cluster or 16 cores in the DELL cluster. The execution times obtained when applying such multi-level parallelism schemes are summarized in Table 5.5. As can be observed, the time incurred in the whole prediction scheme benefits from this combined approach, which enables the regional emergency services to have the possibility to be able to obtain reliable predictions, keeping the time bound.

## 5.3 Experiments using real fires

The last experimentation has been oriented to test the forest fire prediction system (FFSS) and the different prediction schemes in real fire scenarios. One of the most important issues in these cases is the great uncertainty on the input parameters. Another factor to be taken into account is human intervention, which will modify the perimeter s shape. The human fight against the fire is not considered by the fire spread simulators, so our prediction errors should be greater than in prescribed and synthetic fires.

Since the 2ST-WF requires an execution time too large to be useful in real scenarios, we have compared the other three schemes in terms of quality. We are applying techniques to reduce the execution time of the wind field modeler and to reduce the time of the calibration stage in the 2ST-WF scheme.

For the three remaining schemes (2ST-BASIC, 2ST-MM, and 2ST-WF-MM), we have generated five random populations that have been evaluated for each scheme and fire scenario. The calibration technique, as in the rest of the experimentation, is a Genetic Algorithm. We have increased the number of generations to 20, because these tests were the first using real fires. The number of individuals of every population was 50, as in the previous experiments. The GA maintains by elitism the two best individuals to the next generation, the crossover rate is 80%, and the mutation rate has been set at 10%.

In these real cases, the error equation is the modified symmetric difference between maps, which has been shown in Section 2.4. It is depicted again in Eq. 5.2, adding the initial fire factor that has been neglected in that section to simplify it.

$$Error = \frac{\frac{(\cup - Ini) - (\cap - Ini)}{Real - Ini} + \frac{(\cup - Ini) - (\cap - Ini)}{Sim - Ini}}{2} \tag{5.2}$$

### 5.3.1 Top fires - Case 7: Arkadia, Greece

This fire occurrence was located in Arkadia, one of the seven prefectures of the Peloponnese peninsula in Greece. The forest fire occurred in 2011 and began on the 26th of August. The burnt area was 1,761 ha. In Fig. 5.21, the first three perimeters that have been used are depicted as ignition, calibration, and prediction perimeters, respectively.

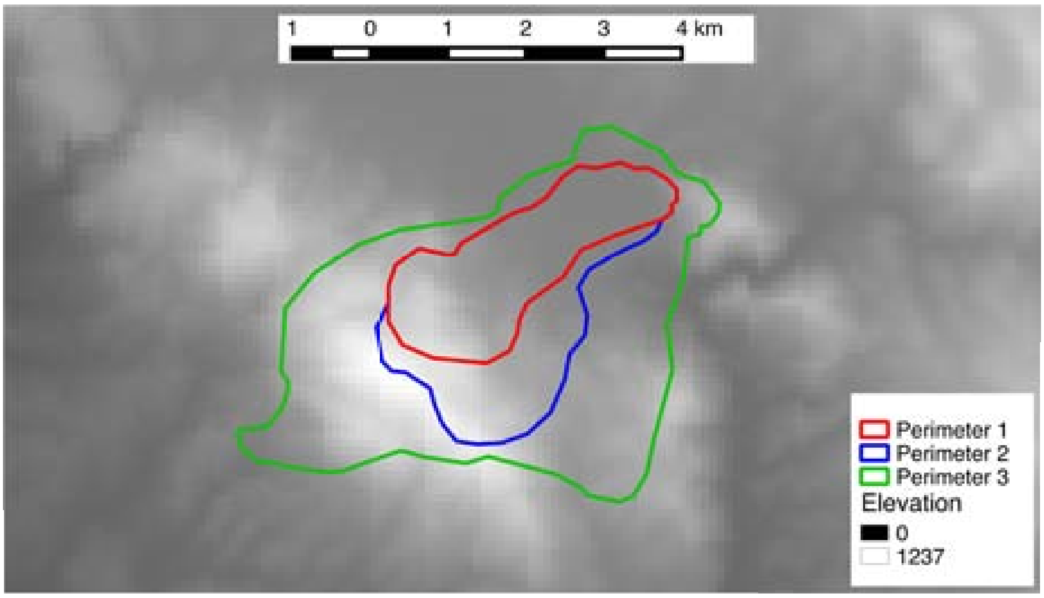


Figure 5.21: Fire perimeters corresponding to Arkadia fire

We have compared the 2ST-BASIC with the 2ST-MM and the 2ST-WF-MM prediction schemes, and the average errors of the population evaluated are shown in Fig. 5.22.

The calibration errors show that the 2ST-BASIC scheme achieves the best error. Neither 2ST-MM nor 2ST-WF-MM can reach the error of the basic scheme, and they performed similarly. It is noteworthy that the interval between the first and second perimeters is around two hours, and there is only a single weather sample in this interval. This lack of knowledge has a direct impact on the quality of the calibration.

Fig. 5.23 shows an example of the best calibrated perimeter for each scheme. All three methods under-predict the fire behavior, and there are

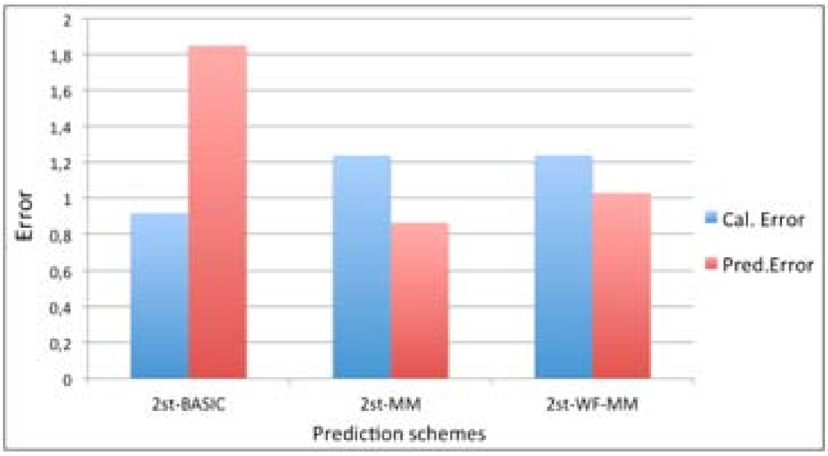


Figure 5.22: Calibration and prediction errors for every prediction scheme

some possible causes for this fact. The measured wind could be less than the reality, and the schemes could not tune the other parameters to minimize this effect. With the last explanation as a possibility, we think that the problem comes from the fuel models used. It is possible that the fuel model conversion from the European cover uses to the standard fire models resulted in low-propagation types. The main reason to support this idea was the behavior of the 2ST-BASIC scheme. Although not sensitive to sudden changes, this method usually finds calibrated winds that make the fire spreading quite close to the real fire, although the final shape can differ due to its uniform conditions.

This situation changes when we analyze the prediction stage that lasts around 23 hours. In this case, the best prediction errors are the ones given by the 2ST-MM and 2ST-WF-MM schemes. The dynamic injection of data seems to be positive to the system and to provide good prediction shapes, as we can see in Fig. 5.24. Although in numerical values the 2ST-MM is the best scheme, the 2ST-WF-MM gives back better perimeters and better covers the real burnt area.

The 2ST-BASIC scheme uses the tuned weather values obtained in the calibration stage, which present a high wind speed value. This causes it to excessively over-predict the real fire behavior.

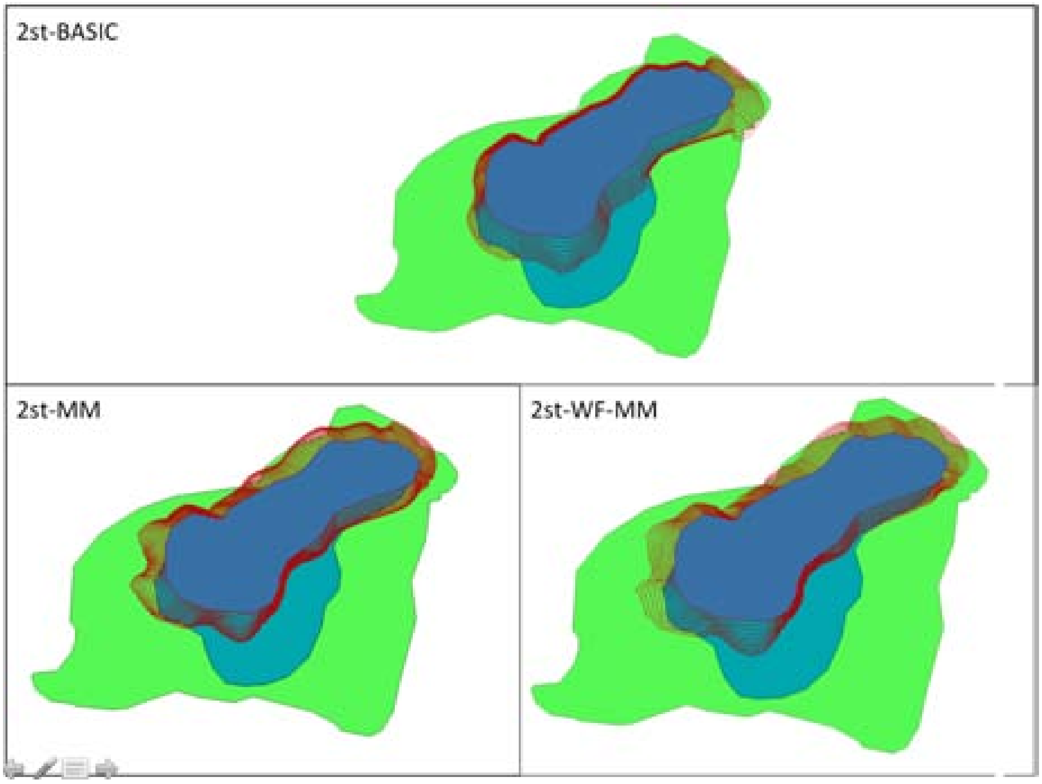


Figure 5.23: Calibration stage perimeters for each prediction scheme

### 5.3.2 Top fires - Case 3: Douro, Portugal

The fire that has been studied in this section was located in Douro, a region that belongs to Portugal. This fire burnt 3,678 ha. and took place on August 28th, 2013. Fig. 5.25 shows the first three perimeters that have been used as ignition, calibration, and prediction perimeters.

In this scenario, the calibration and the prediction stage each last around a complete day. This results in more weather samples for every stage that will benefit the results of the schemes with dynamic weather data injection. Fig. 5.26 shows the calibration and prediction errors for each scheme in this scenario.

The three schemes reach quite good calibration errors, with as the 2ST-BASIC the best by far. The perimeter s shape reflects this situation (see Fig. 5.27), and we can see that the basic scheme almost exactly reproduces

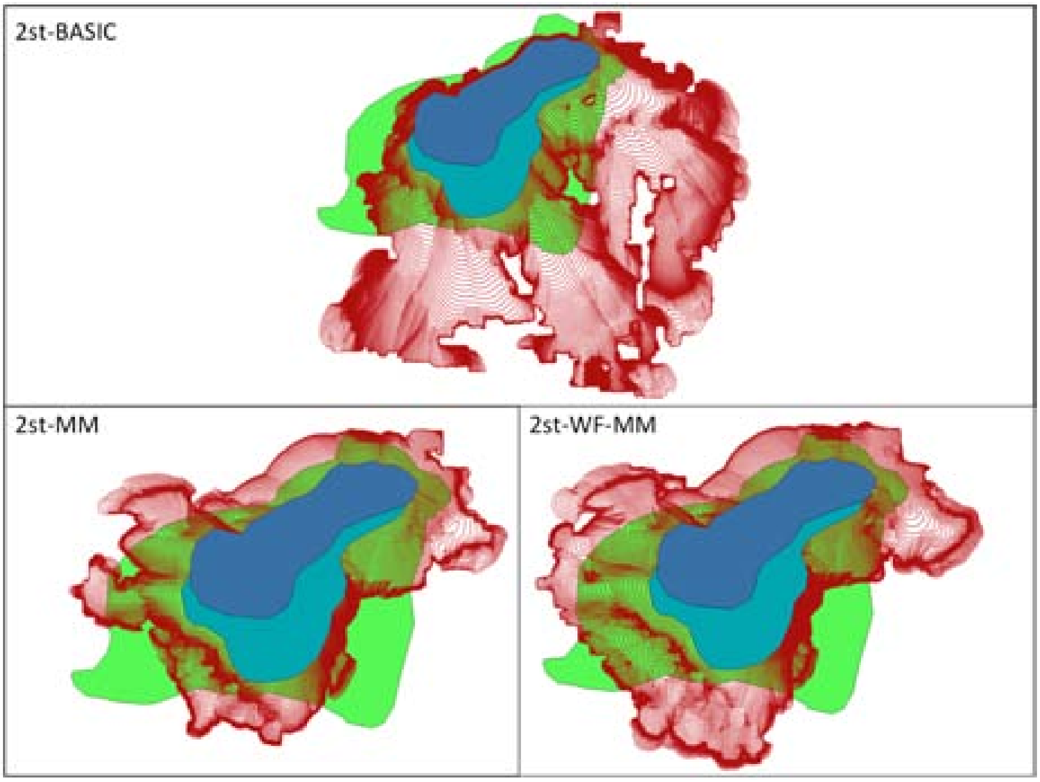


Figure 5.24: Prediction stage perimeters for each prediction scheme

the first perimeter. The other schemes give back shapes with a slight over-prediction, but we must take into account that the system does not consider human action against the fire.

In the prediction stage, the 2ST-MM and the 2ST-WF-MM perform similarly and achieve better errors than the 2ST-BASIC. Despite this, both schemes under-predict the real fire behavior (Fig. 5.28). Again, it seems that the vegetation types modeled for this terrain do not correspond with the real ones, and the calibrated fuel moistures are unable to reverse this mistake.

Despite this fact, the predicted perimeters in these schemes show that the growth trend is quite close to the real fire behavior, and a more accurate vegetation modeling may return better predictions.

Something common in the experiments presented in this section is the fact that the inclusion of wind fields does not provide a high reduction in errors

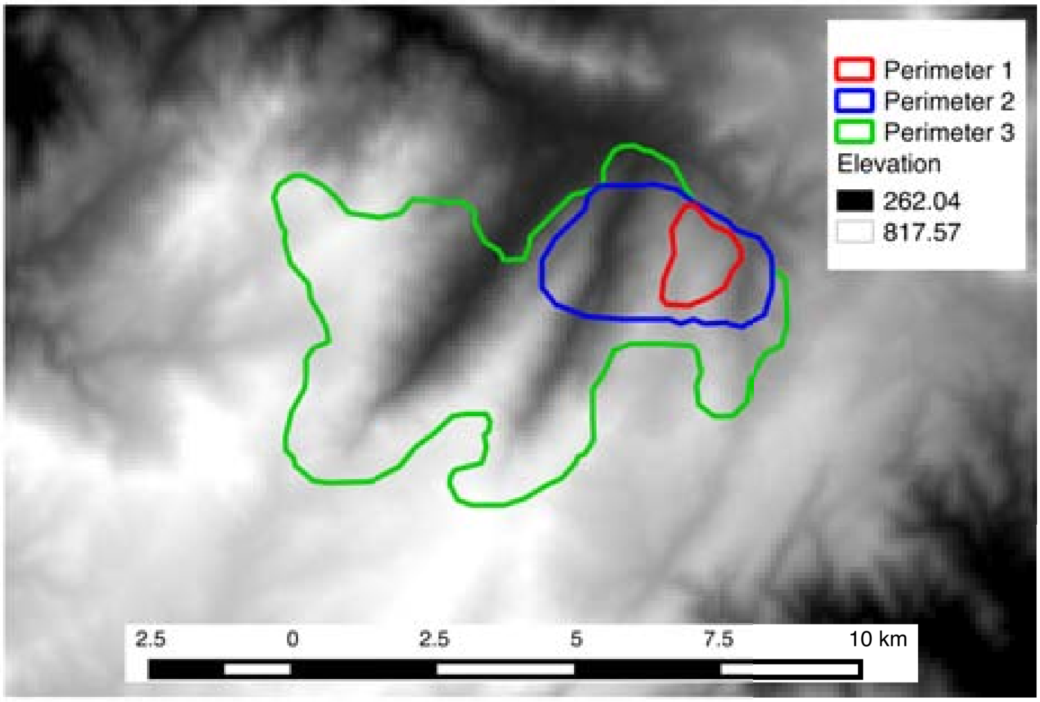


Figure 5.25: Fire perimeters corresponding to Douro fire

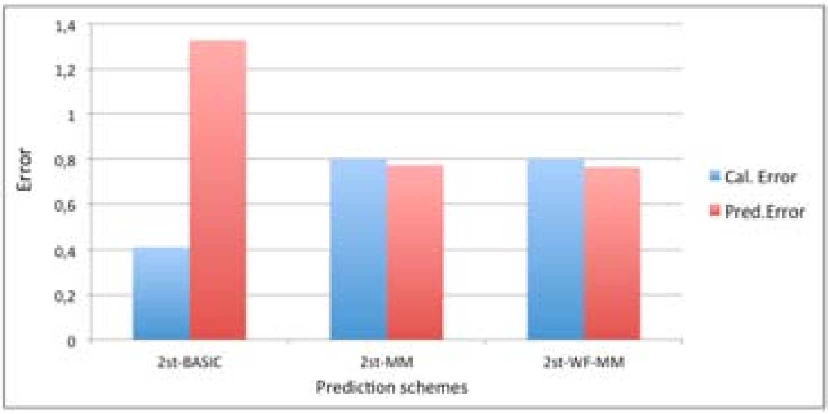


Figure 5.26: Calibration and prediction errors for each prediction scheme

or a great difference in perimeters. We must take into account that the data obtained from this fire case has a common resolution of 100m, instead of 30m, which is the resolution that we usually work with. In addition, the mean wind speed values are low, around 5 to 8 mph, and there are no high wind gusts.

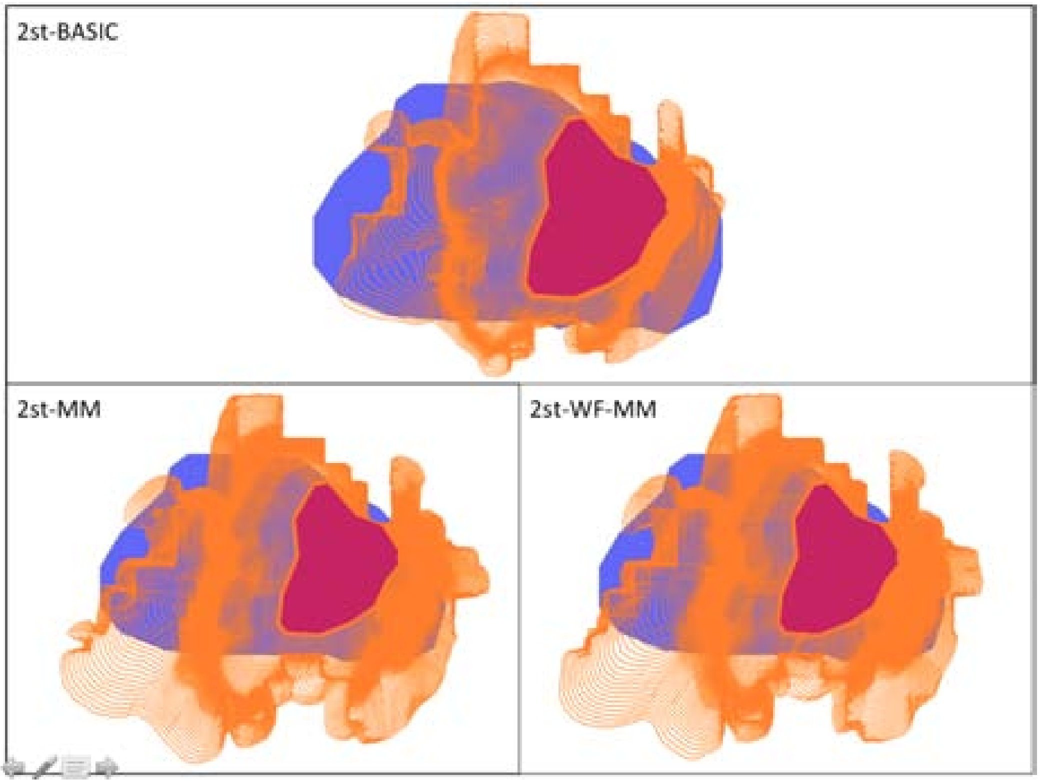


Figure 5.27: Calibration stage perimeters for each prediction scheme

### 5.3.3 Top fires - Case 1: Dao-Lafoes, Portugal

The last fire analyzed also occurred in Portugal, three days after the Douro fire. In this case, it took place in the region of Dao-Lafoes and burnt 2,994 ha. In Fig. 5.29, the first three perimeters that have been used are depicted as ignition, calibration and prediction perimeters, respectively.

Fig. 5.30 shows the calibration and prediction errors in the Dao-Lafoes fire, using the three schemes as in the rest of the real cases. As in the other fire cases, the 2ST-BASIC is the best scheme in the calibration stage. It achieves an extraordinary calibration error, while the schemes with dynamic data injection provide high calibration errors.

Besides, glancing at the perimeter shapes (Fig. 5.31), we extract that the injected weather predictions are wrong. At least, the wind direction seems to have a considerable deviation from the one that guided the real fire. Because

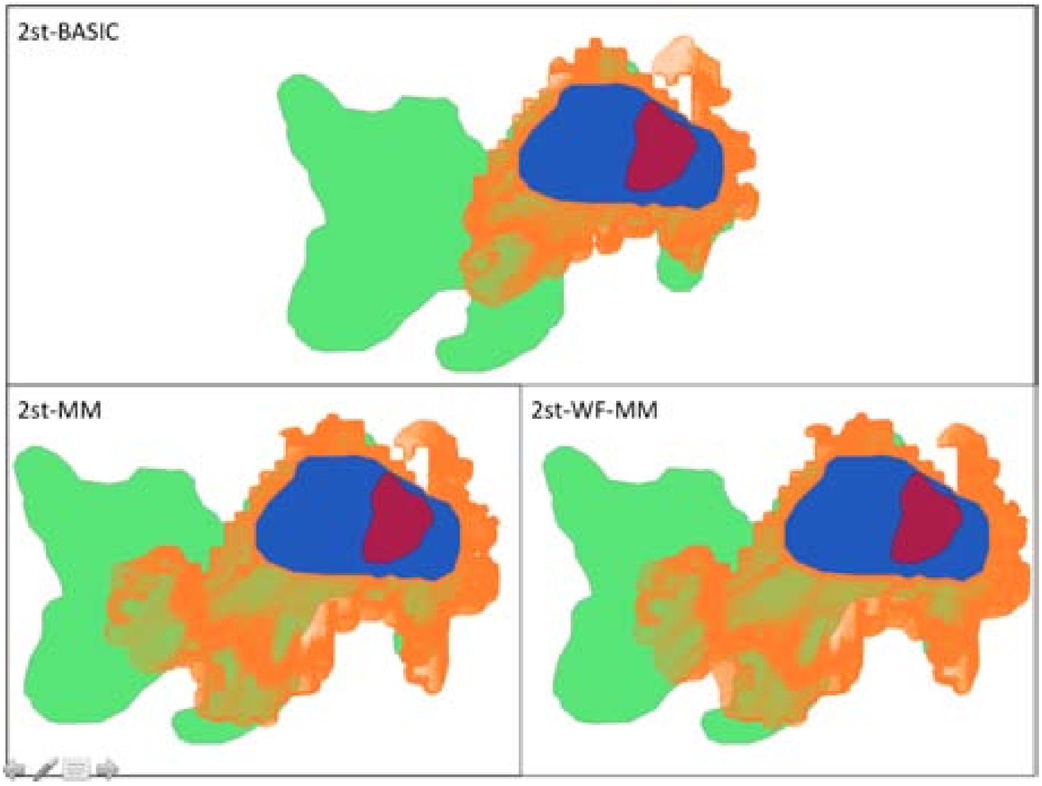


Figure 5.28: Prediction stage perimeters for each prediction scheme

of this, the 2ST-BASIC achieves a very good perimeter while the 2ST-MM and 2ST-WF-MM return bad perimeters.

This situation does not get better in the prediction stage. The prediction errors remain high in those schemes, but where this issue can be better seen is in comparing the real and predicted fire perimeters (see Fig. 5.32). 2ST-MM AND 2ST-WF-MM give back perimeters that do not fit at all, with the real perimeter. However, the 2ST-BASIC provides quite a good prediction.

In this case, the meteorological conditions do not suffer from great changes in conditions, but only have slight changes. This favors the 2ST-BASIC that is able to perform well although it is not sensitive to changes between stages.

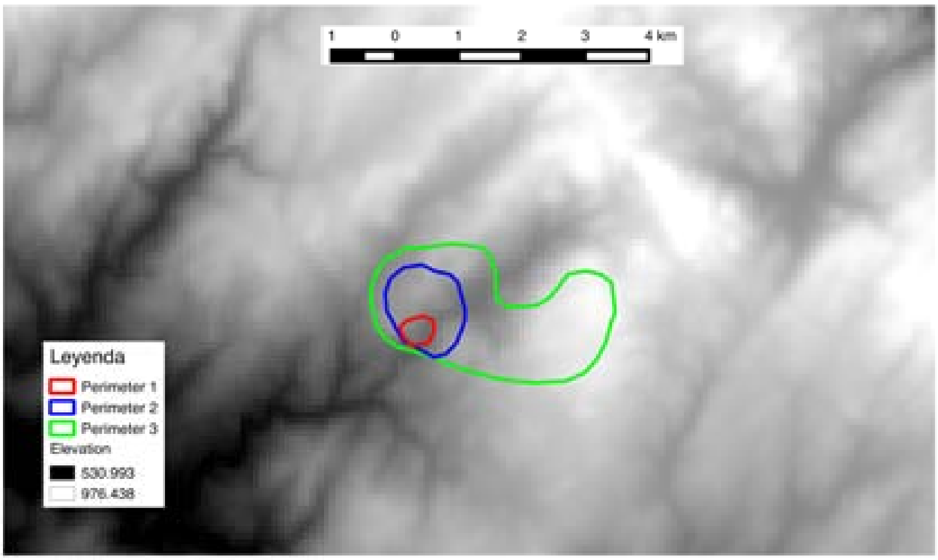


Figure 5.29: Fire perimeters corresponding to Dao-Lafoes fire

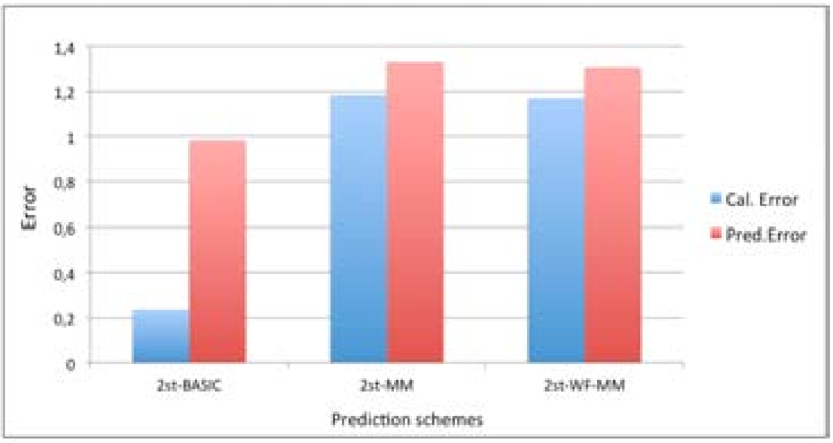


Figure 5.30: Calibration and prediction errors for each prediction scheme

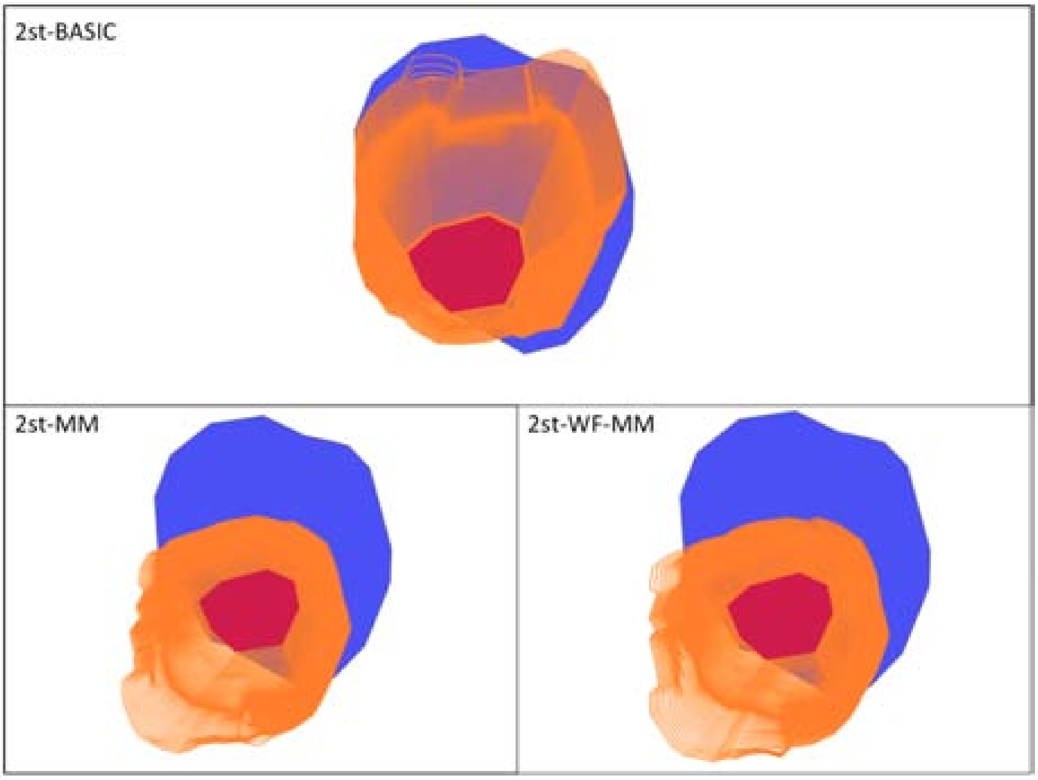


Figure 5.31: Calibration stage perimeters for each prediction scheme

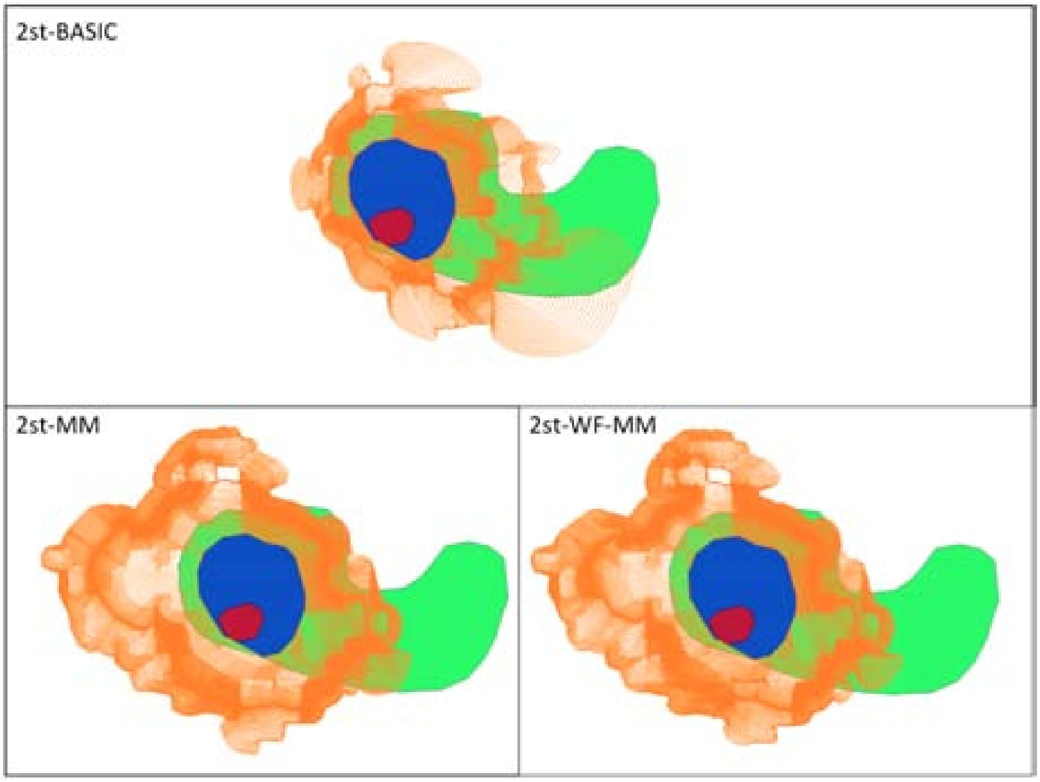


Figure 5.32: Prediction stage perimeters for each prediction scheme

# Chapter 6

# Conclusions and Open Lines

This research work has been focused on improving the prediction of real forest fires using computer simulation and profiting from our knowledge of HPC resources and programming paradigms. Dealing with this kind of fire introduces several new issues that must be tackled to maintain a good response. We started validating our methodology and new proposals using synthetic fires and conditions and introducing step-by-step single features of real fires to study the impact on the spreading individually. Gradually, we have been simulating with more realistic scenarios up to preforming experiments with completely real fires.

To maintain the quality in real environments, we coupled complementary models to produce more realistic simulations. Two models have been analyzed in this work: a wind field modeler to study the wind effect over complex terrain and its consequences in the fire propagation, and a meteorological model to dynamically inject real weather data while the simulation is being carried out. The use of these models produces significant benefits in the prediction quality of our simulations.

But the inclusion of these models is not free, and the computational impact of these new features has been studied, and we have proposed solutions. It is also important to see what the data requirements of this natural hazard are and to propose a methodology to reduce the time and complexity of acquiring and processing the data required to simulate fires.

In the following section, we depict the main contributions of this research.

## 6.1 Contributions

The two-stage prediction method provides us with an improvement in several fire scenarios, but it suffers from two main handicaps. It does not consider the parameter heterogeneity throughout the terrain or their evolution over time. We have included a wind field modeler to avoid the spatial problem, and we have tested the dynamic injection of data coming from a meteorological model. This information gives us a more complete knowledge of the environment and allows us to provide more accurate predictions. These improvements have been included in a hybrid MPI-OpenMP framework that takes advantage of the HPC resources, in order to be able to evaluate a great amount of simulations in an acceptable time. In addition, we have defined a methodology to reproduce and predict real forest fire scenarios, characterizing large areas with high fire risk.

The basic principles that guided this research, the preliminary results of the inclusion of wind field data and the first results evaluating the injection of dynamic weather data in the two-stage prediction system can be seen in the following works:

- Carlos Brun, Tomàs Margalef: Modelos de viento aplicados a la mejora en la predicción de incendios forestales, Master Thesis, Computer Architecture and Operating Systems Department, Universitat Autònoma de Barcelona

- Carlos Brun, Tomàs Margalef, Ana Cortés: Acoplando modelos complementarios para la mejora de un sistema de predicción de incendios forestales, Jornadas Sarteco, JP2012.

Later, we performed a deeper analysis of the effect of the injection of wind dynamics into the system. We tested the influence of including wind data using a synthetic fire evolved in an ideal terrain and comparing our predictions under homogeneous and heterogeneous conditions. The objective was to study how the system behaved when the weather conditions remain quite stable and to compare this with its behavior when there are sudden changes in some meteorological variables. The results of this work were published in:

- Carlos Brun, Tomás Artés, Tomàs Margalef, Ana Cortés: Coupling wind dynamics into a DDDAS forest fire propagation prediction system. Procedia Computer Science 9(0) (2012) 1110 1118 Proceedings of the International Conference on Computational Science, ICCS 2012 (CORE A)

Based on previous work, we coupled the wind field simulator kernel with the prediction system, and we were able to compare the four prediction schemes. To evaluate the performance of all the schemes, we introduce a synthetic fire evolved over a real complex map. The features of this terrain favor the evaluation of the wind field modeler and allow us to study the fire spreading under more realistic conditions. The resulting publication of this work was:

- Carlos Brun, Tomàs Margalef, Ana Cortés: Coupling Diagnostic and Prognostic Models to a Dynamic Data Driven Forest Fire Spread Prediction System. Procedia Computer Science 18 (2013): 1851-1860. ICCS 2013 (CORE A)

In addition, we carried out parallel studies analyzing the computational impact of coupling complementary models. We also used the same synthetic fire evolved over a real terrain, and we studied the execution times of the different schemes using different calibration settings. This last work was extended using another synthetic reference fire, and we used the static data (terrain, vegetation, and conditions) of a big forest fire that occurred in Catalonia (Spain). The comparison of the schemes including combinations of both models can be seen in the following works:

- Carlos Brun, Tomàs Margalef, Ana Cortés: Exploiting multi-core platforms for multi-model forest fire spread prediction. CMMSE(2013) p.308-319

- Carlos Brun, Tomàs Margalef, Ana Cortés, Anna Sikora: Enhancing multi-model forest fire spread prediction by exploiting multi-core parallelism. The Journal of Supercomputing, 1-12 (2014). (Impact factor: 0.917)

In the last phase of this research, we have formalized our methodology to predict real forest fires, we have defined the data sources, the data conversion and adaptation processes and the effective coupling of all the submodules that integrate the prediction framework. In addition, we carried out a complete redefinition of the framework to take advantage of the multi-core parallelism of the new computer architectures using the OpenMP programming paradigm, in conjunction with the Master-Worker MPI scheme. A brief summary of this work has been submitted, and we are waiting for response:

- Carlos Brun, Tomás Artés, Andrés Cencerrado, Ana Cortés, Tomàs Margalef: A GIS-centered methodology for the characterization of large topographic areas for forest fire spread prediction. International Journal of Geographical Information Science. Submitted. (Impact factor: 1.613)

First attempts to accelerate the wind field calculation have resulted in an additional research line, where it has been deeply studied a strategy to speed up these simulations. Some publications have been presented about this issue:

- Gemma Sanjuan, Carlos Brun, Tomás Margalef, Ana Cortés: Paralelización del cálculo del campo de vientos para predicción de la propagación de incendios forestales. XXIII Jornadas de Paralelismo 2013

- Gemma Sanjuan, Carlos Brun, Tomás Margalef, Ana Cortés: Wind field uncertainty in forest fire propagation prediction. International Conference on Computational Science 2014. (ACCEPTED)

- Gemma Sanjuan, Carlos Brun, Tomás Margalef, Ana Cortés: Effect of wind field parallelization on forest fire spread prediction. 14th International Conference on Computational Science and Applications (ICCSA-2014). (ACCEPTED)

- Gemma Sanjuan, Carlos Brun, Tomás Margalef, Ana Cortés: Map Partitioning to accelerate wind Field calculation for Forest Fire Propagation Prediction. International Conference on Forest Fire Research 2014. (ACCEPTED)

- Gemma Sanjuan, Carlos Brun, Tomás Margalef, Ana Cortés: Determining map partitioning to accelerate wind field calculation. International Conference on High Performance Computing & Simulation. (ACCEPTED)

## 6.2 Open Lines and Improvements

At this time, we are performing several experiments to check how our framework behaves using real forest fire scenarios. The first results have been shown in this work, but we want to continue with this experimentation, analyze more cases and decide the next steps in our research.

We are experiencing common problems in real fires, and one of the factors that we think must be revised is the information about the vegetation or fuel models. Up to now, the process was to acquire the land cover uses of the studied terrain and to convert them to the 13 standard fuel models. These models were designed by classifying and characterizing the predominant vegetation types of the United States. What we do is establish a relationship between a land cover use and the closest standard model. Obviously, this generalization produces a loss of information. One of the possible solutions that we are evaluating is to increase the number of genes of the GA and to include an extra parameter per fuel model present in the terrain. These new parameters will be used as adjustment factors to try to modify the standard fuel models and get a more realistic fire spreading.

Another option is to do a deeper analysis of the vegetation features in our area of interest, try to characterize them and, finally, create new custom models adapted to our needs. The drawback of this option is that we need expertise knowledge of this field, and it may require a huge effort to achieve this goal.

As we can see, dealing with real fires adds new issues that must be tackled. The complexity of these scenarios increases the error of our predictions, and new improvements must be studied. Real fire perimeters are influenced by human interaction, and fire spread models do not take this variable into account. Inserting a module to generate fire barriers where is the fire is being attacked into our system will be very interesting to deliver better predicted perimeters.

In addition, our schemes are based on the complementary models need for accurate weather data to give back good predictions. Therefore, when the knowledge of the environment is poor, the injection of real weather data produces wrong predictions. In this case, we propose a hybrid prediction scheme using individuals fully generated by the GA and individuals with predicted weather data. The idea is to increase and guide the search space of the schemes that inject the meteorological data. Now, these schemes acquire the real data and replace these parameters in the individuals of the whole population. The objective is to replace only a portion of the population, and the rest will evolve without any restrictions. In addition, it will be interesting to implement an algorithm that does not inject the meteorological data with the exact value that we receive, but rather uses this value as the center of a confidence interval. Therefore, the meteorological data will not be injected exactly as it comes, but we will introduce disturbances within a range, to take into account the possible measurement errors of this data.

# Bibliography


[1] European Commission and others. Forest fires in Europe, Middle East and North Africa 2012 . In: *JRC Scientific and Technical Reports* 13 (2012).

[2] B. J. Stocks, T. J. Lynham, B. D Lawson, M. E. Alexander, C. E. Van Wagner, R. S. McAlpine, and D. E. Dubé. Canadian Forest Fire Danger Rating System: An Overview . In: *The Forestry Chronicle* 65.4 (1989), pp. 258 265. DOI: 10.5558/tfc65258-4.

[3] European Commission and others. Forest fires in Europe 2003 . In: *JRC Scientific and Technical Reports* 4 (2003).

[4] European Commission and others. Forest fires in Europe 2007 . In: *JRC Scientific and Technical Reports* 8 (2007).

[5] Jon E Keeley, Hugh Safford, CJ Fotheringham, Janet Franklin, and Max Moritz. The 2007 southern California wildfires: lessons in complexity . In: *Journal of Forestry* 107.6 (2009), pp. 287 296.

[6] Luís Guadalupe Zárate López. Estudio de las características físicas y geométricas de la llama en los incendios forestales . PhD thesis. Universitat Politècnica de Catalunya, 2004. URL: http://hdl.handle.net/10803/6436.

[7] Frank A Albini. A Model for Fire Spread in Wildland Fuels by-Radiation . In: *Combustion Science and Technology* 42.5-6 (1985), pp. 229 258.

[8] RO Weber. Analytical models for fire spread due to radiation . In: *Combustion and flame* 78.3 (1989), pp. 398 408.

[9] Jean-Luc Dupuy and Michel Larini. Fire spread through a porous forest fuel bed: A radiative and convective model including fire-induced flow effects . In: *International Journal of Wildland Fire* 9.3 (2000), pp. 155 172.

[10] A.G. McArthur. *Weather and Grassland Fire Behaviour*. Leaflet. Forest Research Institute, Forestry and Timber Bureau, 1966. URL: `http://books.google.es/books?id=H1onHQAACAAJ`.

[11] DH Anderson, EA Catchpole, NJ De Mestre, and T Parkes. Modelling the spread of grass fires . In: *The Journal of the Australian Mathematical Society. Series B. Applied Mathematics* 23.04 (1982), pp. 451 466.

[12] David G Green. Shapes of simulated fires in discrete fuels . In: *Ecological Modelling* 20.1 (1983), pp. 21 32.

[13] E Pastor, L Zarate, E Planas, and J Arnaldos. Mathematical models and calculation systems for the study of wildland fire behaviour . In: *Progress in Energy and Combustion Science* 29.2 (2003), pp. 139 153.

[14] Wallace L Fons. Analysis of fire spread in light forest fuels . In: *Journal of Agricultural Research* 72.3 (1946), pp. 93 121.

[15] Hal E Anderson and Intermountain Forest. *Heat transfer and fire spread*. Intermountain Forest and Range Experiment Station, Forest Service, US Department of Agriculture, 1969.

[16] C. E. Van Wagner. A Simple Fire-Growth Model . In: *The Forestry Chronicle* 45.2 (1969), pp. 103 104. DOI: `10.5558/tfc45103-2`.

[17] Richard C Rothermel and Intermountain Forest. A mathematical model for predicting fire spread in wildland fuels . In: (1972).

[18] Hal E Anderson. Aids to determining fuel models for estimating fire behavior . In: *The Bark Beetles, Fuels, and Fire Bibliography* (1982), p. 143.

[19] Frank A Albini. Wildland fire spread by radiation-a model including fuel cooling by natural convection . In: *Combustion Science and Technology* 45.1-2 (1986), pp. 101 113.

[20] RO Weber. Modelling fire spread through fuel beds . In: *Progress in Energy and Combustion Science* 17.1 (1991), pp. 67 82.

[21] CE Van Wagner. Conditions for the start and spread of crown fire . In: *Canadian Journal of Forest Research* 7.1 (1977), pp. 23 34.

[22] Richard C Rothermel. Predicting behavior and size of crown fires in the Northern Rocky Mountains . In: (1991).

[23] Frank A Albini and Brian J Stocks. Predicted and observed rates of spread of crown fires in immature jack pine . In: *Combustion Science and Technology* 48.1-2 (1986), pp. 65 76.

[24] Mark A Finney. Modeling the spread and behavior of prescribed natural fires . In: *Proceedings of the 12th Conference on Fire and Forest Meteorology*. 1994, pp. 138 143.

[25] Miguel Gomes da Cruz. Modeling the initiation and spread of crown fires . MA thesis. Theses, Dissertations, Professional Papers. Paper 2996., 1999. URL: `http://scholarworks.umt.edu/cgi/viewcontent.cgi?article=4015&context=etd`.

[26] Frank A Albini and Intermountain Forest. *Spot fire distance from burning trees: A predictive model*. Intermountain Forest and Range Experiment Station, Forest Service, US Department of Agriculture, 1979.

[27] Frank A Albini. *Spot fire distance from isolated sources: extensions of a predictive model*. Vol. 309. US Dept. of Agriculture, Forest Service, Intermountain Forest and Range Experiment Station, 1981.

[28] Frank A Albini and Intermountain Forest. *Potential spotting distance from wind-driven surface fires*. US Department of Agriculture, Forest Service, Intermountain Forest and Range Experiment Station, 1983.

[29] Richard C Rothermel. BEHAVE and YOU can predict fire behavior . In: *Fire Management Notes* 44.4 (1983), pp. 11 15.

[30] Patricia L Andrews, Collin D Bevins, Robert C Seli, et al. *BehavePlus Fire Modeling System: Version 2.0: User's Guide*. US Department of Agriculture (Forest Service), Rocky Mountain Research Station Fort Collins, 2003.

[31] Collin D Bevins. FireLib user manual and technical reference . In: (1996). URL: `http://www.fire.org/downloads/fireLib/1.0.4/doc.html`.

[32] D Albright and BN Meisner. Classification of fire simulation systems . In: *Fire management notes* (1999).

[33] Mark A Finney et al. An overview of FlamMap fire modeling capabilities . In: *Proceedings of the Fuels Management-How to Measure Success* (2006), pp. 213 220.

[34] Mark A Finney, Isaac C Grenfell, Charles W McHugh, Robert C Seli, Diane Trethewey, Richard D Stratton, and Stuart Brittain. A method for ensemble wildland fire simulation . In: *Environmental Modeling & Assessment* 16.2 (2011), pp. 153 167.

[35] Mark A Finney, Charles W McHugh, Isaac C Grenfell, Karin L Riley, and Karen C Short. A simulation of probabilistic wildfire risk components for the continental United States . In: *Stochastic Environmental Research and Risk Assessment* 25.7 (2011), pp. 973 1000.

[36] Don W. Carlton. User s guide to using fuels management analyst suite version 3.0.1 . In: (2005). URL: `http://www.fire.org/downloads/fireLib/1.0.4/doc.html`.

[37] Joe H Scott. NEXUS: a system for assessing crown fire hazard . In: *Fire Management Notes* (1999).

[38] AMG Lopes, MG Cruz, and DX Viegas. FireStation an integrated software system for the numerical simulation of fire spread on complex topography . In: *Environmental Modelling & Software* 17.3 (2002), pp. 269 285.

[39] Mark Arnold Finney et al. FARSITE, Fire Area Simulator model development and evaluation . In: (2004).

[40] Baker Abdalhaq, Ana Cortés, Tomàs Margalef, Germán Bianchini, and Emilio Luque. Between classical and ideal: enhancing wildland fire prediction using cluster computing . In: *Cluster Computing* 9.3 (2006), pp. 329 343.

[41] Andrew K. Skidmore. A comparison of techniques for calculating gradient and aspect from a gridded digital elevation model . In: *International journal of geographical information systems* 3.4 (1989), pp. 323 334.

[42] Peter A Burrough, Rachael McDonnell, Peter A Burrough, and Rachael McDonnell. *Principles of geographical information systems*. Vol. 333. Oxford university press Oxford, 1998.

[43] John H Holland. Genetic algorithms . In: *Scientific american* 267.1 (1992), pp. 66 72.

[44] B. Abdalhaq, A. Cortés, T. Margalef, and E. Luque. Enhancing Wildland Fire Prediction on Cluster Systems Applying Evolutionary Optimization Techniques . In: *FGCS The International Journal of Grid Computing* 21.1 (2005). Methods & Applications, pp. 61 67.

[45] M. Denham, K. Wendt, G. Bianchini, A. Cortés, and T. Margalef. Dynamic Data-Driven Genetic Algorithm for forest fire spread prediction . In: *Journal of Computational Science* 3.5 (2012). Advanced Computing Solutions for Health Care and Medicine, pp. 398 404. ISSN: 1877-7503.

[46] Germán Bianchini, Mónica Denham, Ana Cortés, Tomàs Margalef, and Emilio Luque. Wildland fire growth prediction method based on multiple overlapping solution . In: *Journal of Computational Science* 1.4 (2010), pp. 229 237.

[47] Roque Rodriguez, Ana Cortés, Tomás Margalef, and Emilio Luque. An adaptive system for forest fire behavior prediction . In: *Computational Science and Engineering, 2008. CSE'08. 11th IEEE International Conference on*. IEEE. 2008, pp. 275 282.

[48] Neil D Bennett, Barry FW Croke, Giorgio Guariso, Joseph HA Guillaume, Serena H Hamilton, Anthony J Jakeman, Stefano Marsili-Libelli, Lachlan TH Newham, John P Norton, Charles Perrin, et al. Characterising performance of environmental models . In: *Environmental Modelling & Software* 40 (2013), pp. 1 20.

[49] J.M. Forthofer. Modeling wind in complex terrain for use in fire spread prediction . PhD thesis. Colorado State University, 2007.

[50] Frederica Darema. Dynamic Data Driven Applications Systems: A New Paradigm for Application Simulations and Measurements . In: *Computational Science - ICCS 2004*. Ed. by Marian Bubak, Geert van Albada, Peter Sloot, and Jack Dongarra. Vol. 3038. Lecture Notes in Computer Science. Springer Berlin / Heidelberg, 2004, pp. 662 669. ISBN: 978-3-540-22116-6.

[51] TL Clark, J Coen, and D Latham. Description of a coupled atmosphere fire model . In: *Int. J. Wildland Fire* 13.1 (2004), pp. 49 63.

[52] J. L. Coen. Simulation of the Big Elk Fire using coupled atmosphere fire modeling . In: *Int. J. Wildland Fire* 14.1 (2005), pp. 49 59.

[53] J. Mandel, J. D. Beezley, and A. K. Kochanski. Coupled atmosphere-wildland fire modeling with WRF 3.3 and SFIRE 2011 . In: *Geoscientific Model Development* 4.3 (2011), pp. 591 610.

[54] J Michalakes, J Dudhia, D Gill, T Henderson, J Klemp, W Skamarock, and W Wang. The weather research and forecast model: software architecture and performance . In: *Proceedings of the 11th ECMWF Workshop on the Use of High Performance Computing In Meteorology.* Vol. 25. World Scientific. 2004, p. 29.

[55] C. Brun, T. Artés, T. Margalef, and A. Cortés. Coupling Wind Dynamics into a DDDAS Forest Fire Propagation Prediction System . In: *Procedia Computer Science* 9 (2012). Proceedings of the International Conference on Computational Science, ICCS 2012, pp. 1110 1118. ISSN: 1877-0509.

[56] A. Cencerrado, R. Rodríguez, A. Cortés, and T. Margalef. Urgency versus Accuracy: Dynamic Driven Application System Natural Hazard Management . In: *International Journal of Numerical analysis and Modeling* 9 (2012). Ed. by Usda Forest ServiceEditors, pp. 432 448.

[57] T. Artés, A. Cencerrado, A. Cortés, and T. Margalef. Relieving the Effects of Uncertainty in Forest Fire Spread Prediction by Hybrid MPI-OpenMP Parallel Strategies . In: *Procedia Computer Science* 18 (2013). 2013 International Conference on Computational Science, pp. 2278 2287. ISSN: 1877-0509.

[58] NASA (National Aeronautics and Space Administration. *ASTER: Advanced Spaceborne Thermal Emission and Reflection Radiometer*. Jan. 2014. URL: http://asterweb.jpl.nasa.gov.

[59] M Bossard, J Feranec, and J Otahel. *CORINE land cover technical guide: Addendum 2000*. 2000.

[60] E. Chuvieco, D. Riano, I. Aguado, and D. Cocero. Estimation of fuel moisture content from multitemporal analysis of Landsat Thematic Mapper reflectance data: Applications in fire danger assessment . In: *International Journal of Remote Sensing* 23.11 (2002), pp. 2145 2162.

[61] F.M. Danson and P. Bowyer. Estimating live fuel moisture content from remotely sensed reflectance . In: *Remote Sensing of Environment* 92.3 (2004), pp. 309 321. ISSN: 0034-4257.

[62] EFFIS. *European Forest Fire Information System, Joint Reasearch Centre, European Comission*. Jan. 2014. URL: http://forest.jrc.ec.europa.eu/effis/.

[63] METEO.CAT. *Servei Meteorologic de Catalunya*. Dec. 2013. URL: http://www.meteo.cat.

[64] Joseph S Scire, Francoise R Robe, Mark E Fernau, and Robert J Yamartino. A user s guide for the CALMET Meteorological Model . In: *Earth Tech, USA* (2000).

[65] Dario Rodriguez-Aseretto, Daniele de Rigo, Margherita Di Leo, Ana Cortés, and Jesús San-Miguel-Ayanz. A data-driven model for large wildfire behaviour prediction in Europe . In: *Procedia Computer Science* 18 (2013), pp. 1861 1870.