Universitat Autònoma de Barcelona

Modeling Wildland Fire Behaviour using a Multi-physics System on HPC Platforms

Àngel Farguell Caus

Modeling Wildland Fire Behaviour using a Multi-physics System on HPC Platforms

---

Thesis submitted by
Àngel Farguell Caus
in fulfillment of the requirements for the doctoral degree from
Universitat Autònoma de Barcelona
advised by
Dra. Ana Cortés Fité,
Dr. Josep Ramón Miró Cubells and
Dr. Jan Mandel

Barcelona, Spain, September, 2018

Universitat Autònoma de Barcelona
Escola d'Enginyeria
Departament d'Arquitectura de Computadors i Sistemes Operatius

Modeling Wildland Fire Behaviour using a Multi-physics System on HPC Platforms

Thesis submitted by Àngel Farguell Caus in fulfillment of the requirements for the doctoral degree from Universitat Autònoma de Barcelona. This work has been developed under RD 99/2011 in the Computer Science doctoral program and presented to the Computer Architecture & Operating Systems Department at the Escola d'Enginyeria of Universitat Autònoma de Barcelona. This thesis was advised by Dr. Ana Cortés Fité, Dr. Josep Ramón Miró Cubells and Dr. Jan Mandel.

Advisors signature

Author signature

Tutor signature

Acknowledgements

This work was carried out during the years 2015-2018 at the Department of Computer Architecture and Operating Systems, Universitat Autònoma de Barcelona (Spain) with an internship at the Department of Mathematical and Statistical Sciences, University of Colorado Denver (USA).

Firstly, I would like to express my sincere gratitude to my advisors Dr. Ana Cortés Fité, Dr. Josep Ramón Miró Cubells, and Dr. Jan Mandel for the continuous support of my Ph.D study and related research, for their patience, motivation, and immense knowledge. Their guidance helped me in all the time of research and writing of this thesis.

My sincere thanks also goes to my fellow James Haley, without his English corrections and suggestions this thesis would hardly have been written.

I also owe a great debt of gratitude to the European Network on High Performance and Embedded Architecture and Compilation (HiPEAC) for the Collaboration Grant which provides me the opportunity to work as a Research Assistant of Jan Mandel at the Department of Mathematical and Statistical Sciences, University of Colorado Denver.

This thesis was partially supported by grants NSF ICER-1664175, NASA NNX13AH-59G, MINECO-Spain under contracts TIN2014-53234-C2-1-R and TIN2017-84553-C2-1-R.

High-performance computing support at the HPCA4SE (High Performance Computing Applications for Science and Engineering) group at the Universitat Autònoma de Barcelona and CCM (Center for Computational Mathematics) at the University of Colorado Denver, are sincerely thankful. Furthermore, high-performance computing support at CHPC (Center for High Performance Computing) at the University of Utah and Cheyenne (doi:10.5065/D6RX99HX) at NCAR CISL, sponsored by the NSF, are gratefully acknowledged.

I thank my fellow labmates in Universitat Autònoma de Barcelona and University of Colorado Denver for the stimulating discussions, for the sleepless nights we were working together before deadlines, and for all the fun we have had in the last three years.

My sincere thanks also goes to my family and friends for supporting me throughout the years, financially, practically and with moral support, especially my parents. This journey would not have been possible without your support.

Last but not the least, I would like to express my sincere gratitude to my fiancée Natalia Villegas Franco for supporting me spiritually throughout writing this thesis and my days in general.

Abstract

Damages resulting from wildfires have arisen as a major threat worldwide. Properly accounting for the interaction between the fire and the atmosphere surrounding the hazard could aid fire fighters and civil protection staff in making more informed, better decisions during an ongoing event. In that sense, WRF-SFIRE is a wildfire simulator which couples the meteorological model WRF-ARW and the fire spread model SFIRE solving Rothermel's equation through the level set method. This model solves the complex interaction between the atmosphere and the fire through a Computational Fluid Dynamics (CFD) approach. However, it has some limitations which provide the motivation for this investigation.

The aforementioned coupled system needs to run fast enough to assure real-time execution. A deep analysis of the parallelism programmed in WRF-SFIRE is an important matter to get operational results. The best way to run WRF-SFIRE fast is using a distributed memory parallelism with MPI, but it has some limitations because of the dimension of the division of the domain.

Another important element of WRF-SFIRE, which evolves the fire being modeled and keeps it updated, is the level set method. The level set method with strong and heterogeneous rates of spread suffers from instabilities, resulting in spurious fires. This is solved by enforcing the constraint that the level set function at a point may not decrease below the minimum value at neighbors.

Finally, a new method of fitting the fire arrival time to observed perimeter data is proposed. This new method can be used to generate an artificial fire history, which can be used to spin up the atmospheric model for the purpose of starting a simulation from the observed fire perimeter. The main idea is to minimize a non-linear objective function, which is zero when the fire arrival time satisfies the eikonal equation. This new method, unlike position or additive time corrections, respects the dependence of the fire rate of spread on topography, diurnal changes of fuel moisture, winds, as well as spatial fuel heterogeneity. This interpolation method could be used to assimilate fire perimeters and satellite fire detections into the coupled atmosphere-fire model.

Keywords

Coupled Atmosphere-fire Model, WRF-SFIRE, HPC Platforms, Level Set Method, Nonlinear Optimization, Eikonal Equation, Data Assimilation.

Resumen

Los daños causados por los incendios forestales han surgido como una gran amenaza en todo el mundo. Una explicación apropiada de la interacción entre el fuego y la atmósfera alrededor del incendio pueden ser clave para que los bomberos y los cuerpos de protección civil puedan tomar decisiones durante el transcurso del evento. En este sentido, WRF-SFIRE es un simulador de incendios forestales que combina el modelo meteorológico WRF-ARW y el modelo de propagación de incendios SFIRE resolviendo la ecuación de Rothermel a través del método del conjunto de nivel. Este modelo soluciona la compleja interacción entre la atmósfera y el fuego a través de la dinámica de fluidos computacional (CFD). Sin embargo, tiene algunas limitaciones que proporcionan el objetivo de esta investigación.

El sistema acoplado antes mencionado necesita ser ejecutado lo suficientemente rápido como para asegurar ejecuciones en tiempo real. Un análisis profundo del paralelismo programado en él es una cuestión importante para obtener resultados operacionales. La mejor manera de ejecutar rápidamente WRF-SFIRE es utilizando un paralelismo de memoria distribuida con MPI, pero tiene algunas limitaciones debido al tamaño de las partes del dominio.

Otro elemento importante de WRF-SFIRE, que permite evolucionar el incendio y lo mantiene actualizado, es el método del conjunto de nivel. El método del conjunto de nivel con tasas de propagación fuertes y heterogéneas sufre de inestabilidades, lo que provoca incendios espurios. Esto se resuelve cumpliendo que la restricción de que la función del conjunto de nivel en un punto no pueda disminuir por debajo del valor mínimo en los vecinos.

Finalmente, se propone un nuevo método para ajustar el tiempo de llegada del fuego a los datos perimetrales observados, que se puede usar para generar un historial artificial del incendio, que puede utilizarse para *spin up* el modelo atmosférico con el fin de comenzar una simulación desde un perímetro de fuego observado. La idea principal es minimizar una función objetivo no lineal, que satisface la ecuación eikonal cuando va a 0. Este nuevo método, a diferencia de la posición o las correcciones de tiempo aditivas, respeta la dependencia de la velocidad de propagación en la topografía, cambios diurnos en la humedad del combustible, vientos, así como la heterogeneidad espacial del combustible. Este método de interpolación se puede usar para asimilar los perímetros de fuego y las detecciones de fuego satelitales al modelo acoplado de fuego atmosférico.

Palabras clave

Modelo Acoplado de Atmósfera-fuego, WRF-SFIRE, Plataformas HPC, Método del Conjunto de Nivel, Optimización No-lineal, Ecuación Eikonal, Asimilación de Datos.

Resum

Els danys causats pels incendis forestals han sorgit com una gran amenaça a tot el món. Una explicació adequada de la interacció entre el foc i l'atmosfera al voltant del foc pot ser clau perquè els bombers i els cossos de protecció civil puguin prendre decisions durant el transcurs de l'esdeveniment. En aquest sentit, WRF-SFIRE és un simulador d'incendis forestals que combina el model meteorològic WRF-ARW i el model de propagació d'incendis SFIRE que resol l'equació de Rothermel a través del mètode del conjunt de nivell. Aquest model soluciona la interacció entre l'atmosfera i el foc a través de la dinàmica de fluids computacional (CFD). No obstant això, té algunes limitacions que permeten l'objectiu d'aquesta investigació.

El sistema acoblat abans esmentat necessita ser executat prou ràpid com per assegurar execucions en temps real. Una anàlisi profunda del paral·lelisme programat en ell és una qüestió important per obtenir resultats operacionals. La millor manera d'executar ràpidament WRF-SFIRE és utilitzant un paral·lelisme de memòria distribuïda amb MPI, però té algunes limitacions causa de la grandària de les parts del domini.

Un altre element important de WRF-SFIRE, que permet evolucionar l'incendi i el manté actualitzat, és el mètode del conjunt de nivell. El mètode del conjunt de nivell amb taxes de propagació fortes i heterogènies pateix d'inestabilitats, el que provoca incendis espuris. Això es resol complint la restricció de que la funció del conjunt de nivell en un punt no pugui disminuir per sota del valor mínim en els veïns.

Finalment, es proposa un nou mètode per ajustar el temps d'arribada del foc a les dades perimetrals observades, que es pot utilitzar per generar un historial artificial de l'incendi, que pot utilitzar-se per *spin up* el model atmosfèric per tal de començar una simulació des d'un perímetre de foc observat. La idea principal és minimitzar una funció objectiu no lineal, que satisfà l'equació eikonal quan va a 0. Aquest nou mètode, a diferència de la posició o les correccions de temps additives, respecta la dependència de la velocitat de propagació en la topografia, canvis diürns en la humitat del combustible, vents, així com la heterogeneïtat espacial del combustible. Aquest mètode d'interpolació es pot usar per assimilar els perímetres de foc i les deteccions de foc satelitals al model acoblat de atmosfera-foc.

Paraules clau

Model Acoblat d'Atmosfera-foc, WRF-SFIRE, Plataformes HPC, Mètode del Conjunt de Nivell, Optimització No-lineal, Equació Eikonal, Assimilació de Dades.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Every year, millions of hectares of forest are devastated by wildfires. This fact causes dramatic damage to innumerable factors such as health [24], economy [9], ecosystem [48], biodiversity [49], and society [57]. It has been recognized that the recent increase in fire severity is associated with strict fire suppression policies, that over last several decades led to significant accumulation of fuel, which when ignited makes fires difficult to control [1]. In order to reverse this effect, prescribed burns are routinely used as a method of fuel reduction and habitat maintenance [56, 71]. The previous strategy of putting out all wildland fires is being replaced by a new approach where the fire is considered as a tool in the land management practice, and some fires are allowed to burn under appropriate conditions in order to reduce the fuel load and meet the forest management goals [21].

Fire management decisions regarding both prescribed burns, as well as wildland fires, are very complex. They require a careful consideration of potential fire effects under changing weather conditions, values at risk, firefighter safety, and air quality impacts of wildfire smoke [74]. In order to help in the fire management practice, a wide range of models and tools have been developed. The first attempt to reproduce the fire spread in a forest was done by Curry and Fons (1938) [20] and Fons (1946) [27] which was used to build the first set of semi-empirical equations describing the fire spread [64, 2]. This first formulation gave the first impulse to start modeling wildland fires using numerical fire spread models to propagate the evolution of the fire. These models consist of Rate of Spread (ROS) formulation and a moving interface [35]. There are several examples of ROS formulation in literature: Rothermel [64], Finney [26], Mallet [40] that can be combined with any moving interface method. Among different methods for moving interfaces, the Level Set Method (LSM) [55] is a well-known technique used to evolve dynamical systems such as the progression of a fire perimeter which has been applied for different authors [38, 45, 63]. The LSM method is based on an Eulerian moving interface method and gives a useful representation of what part of a domain is burned (taking negative values) and what part is not (taking positive values). Another moving interface methodology is based in Lagrangian Discrete Event System Specification (DEVS) [34, 52, 23], this formulation does not use a mesh for calculating the evolution of fire and permits faster simulations at higher resolution. Finally there is the formulation based on Huygens' principle [61, 62, 3] which uses an elliptical spread

of each point of the fire front.

The first operational models were constricted by the computing capabilities and in general were uncoupled from atmospheric models. This models are based on a semi-empirical formulation in which the ROS equation is formulated using the energy balance equation tuned with experimental data. In these models, elevation data (slope) and fuel characteristics are used together with ambient weather conditions or a general weather forecast as inputs to the rate of spread model, which computes the fire propagation, neglecting the impact of the fire itself on local weather conditions (see BehavePlus [5], FARSITE [25] or PROMETHEUS [72]). In these tools, the atmospheric model provides an external forcing to the fire spread model, but the prognostic and diagnostic variables of the fire spread model (fire front position, heat fluxes,...) are not used to update the atmospheric model. Therefore, they do not take into account one of the essential parts of simulating the fire spread model, the fire's effect on the atmosphere. The predictability potential of a multi-physics wildfire spread system is supposed to be better than considering the atmospheric and wildfire models as two isolated systems [12, 10, 11]. Therefore, as computational capabilities increase, a new generation of coupled fire-atmosphere models become available for fire managers as operational tools. In a coupled fire-atmosphere model, weather conditions are computed simultaneously with the fire propagation. This means that the state of the atmosphere is modified by the fire so that the fire spread model is driven by the local micro-meteorology modified by the fire-released heat and moisture fluxes. The main examples of this types of models are WRF-SFIRE [43] which uses the LSM moving interface and FOREFIRE/Meso-NH [22] with the DEVS moving interface. Other examples are CAWFE [15] and BRAMS-FIRE [28].

The last versions of the mesoscale models incorporates a CFD scheme which allow have high resolution weather simulation up to resolutions of tens of meters, that are coupled with semi-empirical fire spread models. In this cases, the fire model have a reference NWP model which gives the meteorological variables involved in the fire spread. This approach is fundamentally similar to so-called physics-based models like FIRETEC [39] and WFDS [47], which also use CFD approach to compute the flow near the fire, but focus on flame-scale processes in order to directly resolve combustion and heat transfer within the fuel and between the fire and the atmosphere. As the computational cost of running these models is too high to facilitate their use as forecasting tools using high performance computing, this thesis focuses on the aforementioned hybrid approach, where the fire and the atmosphere evolve simultaneously affecting each other, but the fire spread is parameterized as a function of the wind speed and fuel properties, rather than resolved based on the detailed energy balance. Therefore, this thesis proposes to improve some aspects of WRF-SFIRE [37], a fire-atmosphere model described in Chapter 2. WRF-SFIRE combines a mesoscale numerical weather prediction system WRF [68], which represents the state-of-the-art of the Numerical Weather Prediction (NWP) developments, with a surface fire behavior model implemented by a level set method [54], a fuel moisture model [73], and chemical transport of emissions [41]. The coupling between the models is graphically represented in the diagram in Figure 1.1. The fire heat flux modifies the atmospheric state (including local winds), which in turn affects fire progression and the fire heat release. WRF-SFIRE has evolved

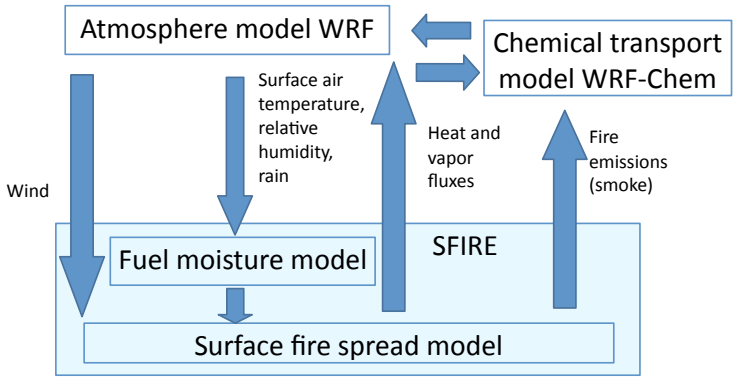


Figure 1.1: Diagram of the coupling between the atmospheric and fire spread models in WRF-SFIRE. Figure from [41].

from CAWFE [11, 13], and an earlier version [43], distributed with the WRF release as WRF-Fire [14], was recently improved by a high-order accurate level-set method [50]. The model WRF [69] is a

WRF-SFIRE solves the multi-physical problem in a cyclic fashion. That is, meteorological data is obtained by running WRF and the obtained WRF data is used to expand the fire front numerically. At that point, the heat fluxes that fire emits to the atmosphere are captured by the system and used to compute WRF for the next time step. This constant feedback between the two models makes it possible to describe all the processes in a more realistic way. However, it has a significant computational cost which requires to exploit High Performance Computing (HPC) to the maximum. In general the number of cores and number of CPU used are determinant for HPC, but isn't straightforward look for configuration that optimizes the processes [33]. The computer architecture and the configurations of CPU and memory use are very significants in the performance of the simulations running time [30]. WRF-SFIRE is implemented using two different parallel paradigms, MPI distributed memory parallelization and OpenMP shared memory parallelization. The MPI parallelization consists in dividing the domain into different square regions called patches which are evolved in parallel. In the case of OpenMP, each patch is divided into horizontal tiles which share the same memory space and compute in parallel as well. Therefore, a first objective is to analyze these two parallel paradigms in order to observe their characteristics and limitations (Chapter 3).

In this thesis as the WRF-SFIRE model will be used, the ROS formulation will be the one of Rothermel and the moving interface the LSM. In this case, the fire front structure is able to evolve using the level set equation which gives the location where the perimeter is going to move depending on the rate of spread, which depends on the wind and the slope of the terrain. If the rate of spread is homogeneous, the level set function produces at a smooth solution. The problem appears when the rate of spread is strong and heterogeneous. This fact causes spurious ignitions which do not make sense in terms of the real solution. Therefore, in Chapter 4 the problem is analyzed and a solution to avoid this unrealistic situation is proposed. The final solution is tested

and added to the master branch of WRF-SFIRE Git repository [16].

In order to adjust the coupled fire-atmosphere models to the observation, several methods of initialization and real-time adjusts can be used. These techniques have been evolved together with the computation power and very related to the atmospheric modeling. The most straightforward techniques were based in successive corrections and most representative methods are Cressman [19] and Barnes [7]. Later, the statistical methods are developed as Optimum Interpolation for obtaining an state which minimizes the error variance [66], in the 70s Anthes [6] and Hoke [31] developed the nudging method in which the model is dynamically corrected using the error between observations and the forecast, actually the variational data assimilation methods are widely applied for data assimilation using the 3D-Var [18] and 4D-Var techniques [58], more recently the Kalman Filters [32] has become a basic tool for data assimilation

Focusing in wildland forest fires, some authors try to correct the simulations using collected meteorological data from the surroundings of the wildfire [44], the main problem of this technique is that meteorological conditions near and inside the fire are too changing and the combustion region quite thin. An alternative to this was adjusts the perimeters simulated by the models to the ones observed using data from satellite or observed fire perimeters [70], [63], both using Kalman filters for adjusting these perimeters. In our case, as we are working with a coupled fire-atmosphere numerical model, the ignition procedure itself affects the atmospheric state (especially local updrafts near the fire line and the near fire winds). Therefore, particular attention is needed during to the data assimilation process in order to assure that realistic fire-induced atmospheric circulation is established at the time of data assimilation.

One possible solution to this problem, assuring consistency between the fire and the atmospheric models, is defining an artificial fire progression history, and using it to replay the fire progression prior to the assimilation time. In this case, the heat release computed from the synthetic fire history is used to spin up the atmospheric model and assure consistency between the assimilated fire and the local micro-meteorology generated by the fire itself. Therefore, a new method of fitting the fire arrival time to data is proposed, which can be used to generate an artificial fire history (Chapter 5).

The organization of the thesis is according to the following summary. A detailed explanation of WRF-SFIRE is given in Chapter 2. After this, Chapter 3 analyzes the behavior of the multi-physics system on multi-core platforms. Then, a solution to the problem of spurious fire ignitions for the level set method with strong and heterogeneous rate of spread is presented in Chapter 4. A new method of fitting the fire arrival time to data is formulated, solved, and tested for some experiments in Chapter 5. Finally, the main conclusions of this work and the open lines are reported in Chapter 6.

Chapter 2

WRF-SFIRE

This chapter describes the different components of the wildfire simulator WRF-SFIRE, which couples the meteorological model WRF-ARW [60] and the fire spread model SFIRE in an open source Fortran code available from the Github repository [16].

2.1 General structure

In order to couple the two different scales (atmosphere and fire combustion), WRF-SFIRE uses a 3D mesh representing the atmosphere and the fire is represented on a finer 2D mesh located at the surface of the 3D atmospheric mesh (Figure 2.1).

The three dimensions from the 3D atmospheric mesh are two horizontal x and y directions and one vertical z direction. The two horizontal directions are defined as longitude and latitude geolocation coordinates, and the vertical levels or η vertical coordinates are defined as

$$\eta = \frac{p_h - p_{ht}}{\mu},$$

where $\mu = p_{hs} - p_{ht}$, p_h is the hydrostatic component of the pressure, p_{hs} is the hydrostatic pressure value along the surface boundary (it changes depending on the land or orography) and p_{ht} is the hydrostatic pressure value along the top boundary (it is a constant). Therefore, η varies from 1 at the surface to 0 at the upper boundary of the model domain.

In Figure 2.2, one can observe that the atmospheric variables are located at different places of the cubic cell according to the following:

- θ point locations, not staggered: They are located at the center of the cubic cell. For instance, the thermodynamical variables such as temperature and pressure.
- U point locations, staggered in x direction: They are located at the center of the left face. The x component of the wind is an example of this category.
- V point locations, staggered in y direction: They are located at the center of the front face. The y component of the wind is located in the V points.
- W point locations, staggered in η direction: They are located at the center of the bottom face. For instance, the η components of the wind and the geopotential.

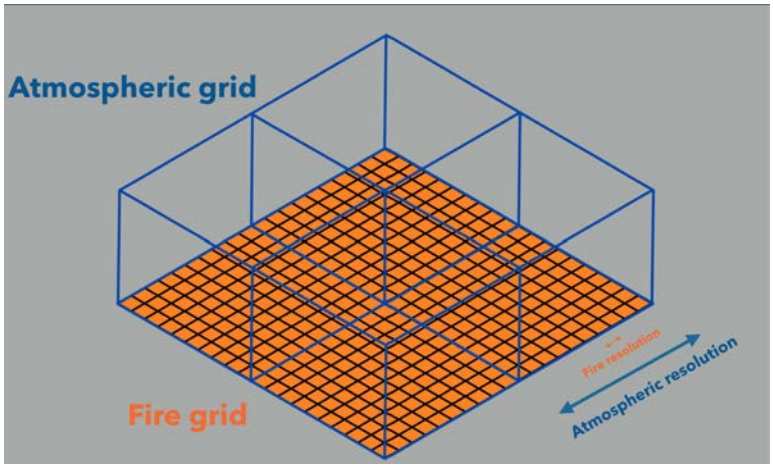


Figure 2.1: 3D atmospheric mesh and 2D fire mesh with their respective resolutions.

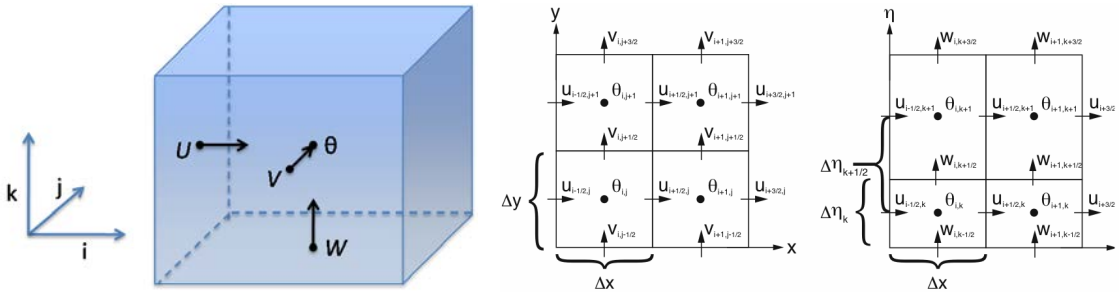


Figure 2.2: Left figure: Definition of the θ, U and V locations of a cubic atmospheric cell. Right figure: Detailed location of θ, U and V locations from the x-y and x-η planes. Both figures are from [17].

All variables on the fire mesh are located at the centers of the 2D fire mesh cells except the x and y components of the fire mesh winds, *UAH* and *VAH*, which are represented on U and V locations of the fire mesh cells. Furthermore, 4 subcells are generated for each fire cell in order to solve the combustion process. In order to construct the fire mesh, one needs to define mesh ratio between atmosphere and fire meshes. This ratio gives the number of fire cells inside each atmospheric cell. In Figure 2.3 the ratio 1:4 between atmosphere and fire meshes is illustrated. These two main meshes are recommended to satisfy at least a ratio of 1:10 in x and y direction (sr_x and sr_y) [36]. So, for each atmospheric node, 10 fire points are required in order to model in a suitable way the interaction between the atmosphere and the fire. Furthermore, to ensure convergence, it is required that the ratio between the temporal resolution in seconds (dt) and the spatial resolution of the atmospheric mesh in kilometers (dx) satisfies:

$$\frac{dt}{dx} \leq 6 \text{ [43].}$$

WRF-SFIRE is implemented in each iteration of WRF, where the fire is evolved in

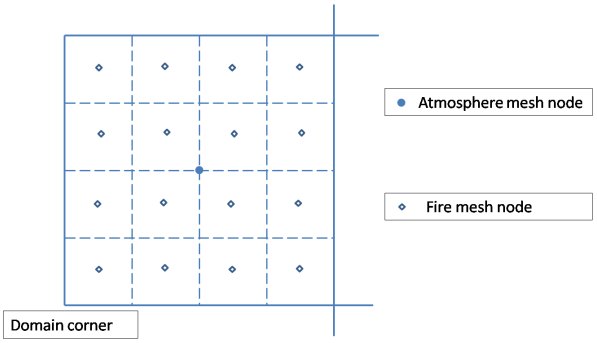


Figure 2.3: Ratio 1:4 between atmosphere and fire meshes using an example of a one atmospheric cell and four fire cells. The figure is from [17].

a cyclic fashion. Therefore, in each iteration of WRF:

1. The WRF model is solved on the atmospheric grid.
2. The winds are interpolated from the atmospheric to the fire mesh.
3. The fire ignition is generated if it was not previously generated.
4. The fire front is evolved using the discretized level set method.
5. The fire arrival time and the fuel fraction are computed in each fire node.
6. The heat fluxes from the fire to the atmosphere are computed from the previous fuel fraction and the previous fire arrival time.
7. These fluxes are added to the WRF model, which is computed again returning to step 1.

Using this work-flow, the two dependences are achieved. The atmosphere depends on the fire from the heat fluxes and the fire depends on the atmosphere from the wind, temperature, humidity, and other variables.

2.2 Fire spread model

The variable of interest when modeling fire spread is the speed at which the fire line propagates or rate of spread R. This rate is computed using the semi-empirical Rothermel formula [65] which computes the rate of spread at the normal direction of the fire front depending on the effects of fuel, slope and wind. Therefore, the Rothermel formula gives the rate of spread in the normal direction as

$$R = R_0(1 + \phi_w + \phi_s), \tag{2.1}$$

where R_0 is the rate of fire spread dismissing the effects of wind and slope, ϕ_w is the wind factor and ϕ_s is the slope factor. However, a front-tracking solver is required in

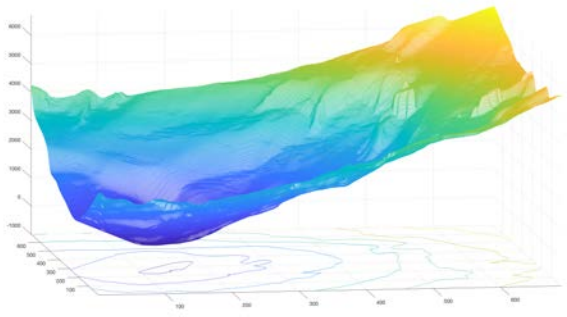


Figure 2.4: The level set function used for modeling the evolution of a wildland fire.

order to propagate the fire front at the previous rate of spread in equation (2.1). WRF-SFIRE uses the level set method [54] which uses a three dimensional construction, the level set function, to define the evolution of the fire front by level sets as time progresses. The level set function is used in order to define which parts of the domain are either part of the fire front, burned or unburnt. So, it is defined as

$$\begin{cases} \psi(\vec{x},t) < 0 & \text{Burned area} \\ \psi(\vec{x},t) = 0 & \text{Fire front} \\ \psi(\vec{x},t) > 0 & \text{Unburnt area} \end{cases} \tag{2.2}$$

where $\vec{x} \in \mathbb{R}^2$ is a point in the domain and t is the temporal variable. Therefore, at each time τ, the level set $\psi(\vec{x},\tau) = 0$ defines the fire front. Figure 2.4 shows these fire fronts for different times using different colors (z axis of the figure) over the domain (x-y plane of the figure). One can also observe that the level set function is a three dimensional volume because it has two spatial dimensions and one temporal dimension.

This level set function is evolved temporally using the partial differential equation

$$\frac{\partial \psi}{\partial t} + R||\nabla \psi|| = 0, \tag{2.3}$$

which uses the previous rate of spread R as a vector in the normal direction defined as

$$\vec{n} = \frac{\nabla \psi}{||\nabla \psi||},$$

which is computed by finite central differences.

The previous equation (2.3) is solved numerically using the Runge-Kutta method of order 2. Therefore, having solved the iteration k, the iteration $k+1$ is defined as

$$\begin{aligned} \psi^{k+\frac{1}{2}} &= \psi^k + \Delta t \; F(\psi^k) \\ \psi^{k+1} &= \psi^k + \Delta t \Big(\frac{1}{2} F(\psi^k) + \frac{1}{2} F\big(\psi^{k+\frac{1}{2}}\big) \Big) \end{aligned}$$

where F is a discretization of $-R||\nabla \psi||$ using an upwinding method and an artificial viscosity term for added stability considerations. So

$$F(\psi) = -R||\hat{\nabla} \psi|| + \epsilon \tilde{\Delta} \psi,$$

where $\hat{\nabla}\psi$ is the upwinding finite differences approximation of $\nabla\psi$ selected in the configuration file (*namelist.input* file). The essentially nonoscillatory (ENO) method [53] of order one explained in the Section 5.4.3 is typically used to ensure evolving the information from the past to the future. The viscosity term is scaled using the scaled-free artificial viscosity $\epsilon = 0.4$ and

$$\tilde{\Delta}\psi \approx \Delta x \frac{\partial^2\psi}{\partial^2 x} + \Delta y \frac{\partial^2\psi}{\partial^2 y} \approx$$
$$\approx \frac{\psi(x+\Delta x, y) - 2\psi(x,y) + \psi(x-\Delta x, y)}{\Delta x} + \frac{\psi(x, y+\Delta y) - 2\psi(x,y) + \psi(x, y-\Delta y)}{\Delta y}.$$

2.3 Coupling

In this section, the interaction between WRF atmosphere model and SFIRE spread fire model is explained. Therefore, steps 2 and 6 in the iteration work-flow of the Section 2.1 are described in more detail.

First, in step 2, the output winds from WRF-ARW solved in the 3D atmospheric mesh are horizontally interpolated from the atmospheric mesh and vertically interpolated to a vertical level ζ_f in the 2D fire mesh by assuming the ideal logarithmic wind profile. Therefore, wind u depends on the height about the terrain ζ as defined in [43]

$$u(\zeta) \approx \begin{cases} c \ln \frac{\zeta}{\zeta_0}, & \zeta_0 \leq \zeta \\ 0, & 0 \leq \zeta \leq \zeta_0 \end{cases}$$

where ζ_0 is the roughness height and c is a proportionality constant. The common roughness height is $\zeta_0 = 0.13H$, where H is the vegetation height and the vertical level to interpolate from ζ_f depends also on the vegetation category. This wind interpolated into the fire mesh is then used to compute the rate of spread formula in the equation (2.1) which in turn is used to drive the fire spread simulation.

In step 6, after running the fire spread model, the heat fluxes are inserted into the WRF-ARW atmospheric model. In order to compute the heat fluxes, one needs to calculate at each location the fuel fraction. Each location starts with fuel fraction $F_f = 1$. Then, once the location is ignited at time t_i, in all times such that $t > t_i$, the fuel fraction is defined as

$$F_f(t) = \exp\Big(-\frac{t - t_i}{T_f}\Big) \text{ [43]},$$

where T_f is the fuel burn time, which is proportional to the fuel weight ω as

$$T_f \approx \frac{\omega}{0.8514}.$$

In WRF-SFIRE the values ω from each of 13 Rothermel's fuel categories are from the NCAR's CAWFE code [12, 10, 11, 13]. Then, this fuel fraction is integrated in the previous subcells and the heat fluxes are computed as in the CAWFE code, so the

average sensible heat flux density released in the temporal interval $[t, t + \Delta t]$ in (W m^{-2}) is computed as

$$\phi_h = \frac{F_f(t) - F_f(t + \Delta t)}{\Delta t} \frac{1}{1 + M_f} \omega_l h,$$

and the average latent heat flux density at the same temporal interval is computed as

$$\phi_q = \frac{F_f(t) - F_f(t + \Delta t)}{\Delta t} \frac{M_f + m_w}{1 + M_f} L \omega_l,$$

where M_f is the fuel particle moisture content (dimensionless), ω_l is the total fuel load in (kg m^{-2}), h is the fuel heat contents of dry fuel in (J kg^{-1}), $m_f = 0.56$ is the estimated mass ratio of the water output from the combustion of dry fuel, and $L = 2.5 \times 10^6$ J Kg^{-1} is the specific latent heat of condensation of water at 0^oC. All of these variables depend on the fuel type category assigned at each particular point.

Finally, these heat fluxes ϕ_h and ϕ_q are added to the WRF model as forcing terms on a layer above the surface in the differential equations of the atmospheric model, which assumes an exponential decay with increasing altitude.

Chapter 3

Scalability Study

In order to predict the evolution of a fire operationally, one needs to be able to run the simulation in real-time. A coupled atmosphere-fire simulator is computationally expensive because of the necessity for evolving the atmosphere and the fire simulation while interchanging information. In this chapter, the parallel paradigms programmed in the coupled atmosphere-fire simulator WRF-SFIRE are analyzed. Therefore, a scalability study of WRF-SFIRE parallelization is achieved. Furthermore, the operational time capabilities of WRF-SFIRE are analyzed.

3.1 Experimental scenario

3.1.1 Cardona fire

In order to properly analyze the parallel paradigms, it is important to choose an appropriate test case. The real case analyzed is a wildfire which occurred in Catalonia (North-East of Spain) on July 8, 2005 [29]. The 2005 Cardona fire burned a total of 1438 hectares and lasted for 6 hours. The fire started at 14:30 and it burns until approximately 20:30. This particular case was labeled by the firefighters as a wildfire driven by the winds generated by the fire itself. Thus, it is a perfect example to show the benefits of using a multi-physics system that takes into account the feedback between the atmosphere and fire processes.

To run WRF-SIFRE, it is required to determine the computational domains and mesh resolutions, taking into account that the atmospheric model is represented using a 3D grid and that the fire is propagated at the surface, on a 2D horizontal grid of the 3D atmospheric grid (Chapter 2). The size of the computational domain is fixed in this case and corresponds to 49 km^2 (7 km $\times$ 7 km). This computational domain is large enough to contain the evolution over the whole 6-hour time period. Once the domain has been determined, the mesh resolutions are defined from the input data resolutions. The digital elevation map used has a resolution of 25 meters, which is the maximum input data resolution used in the system. For that reason, the initial coarser fire grid resolution is chosen to be 24 meters, which is the number closest to 25 that allows for having fire and meteorological grid resolutions that fit the underlying domain size and at the same time exploit the initial data fully. Regarding the atmospheric mesh

Table 3.1: Grid resolutions used in the four experiments performed.

Atmospheric res. (meters)	Fire res. (meters)	Atmospheric grid points per vertical level
236	24	29x29
118	12	59x59
100	10	70x70
59	6	116x116

initialization, the initial weather data used as initial conditions for the simulations is a weather data set provided by the SMC (Servei Meteorologic de Catalunya) [67], which is at 3 km horizontal resolution and interpolated at 19 vertical levels. Then, the coupled atmosphere-fire model is simulated using different atmosphere and fire grid resolutions. Table 3.1 summarizes the four atmospheric and fire grid resolutions used in the experimental study as well as the number of points in the atmospheric grid in one vertical level for each configuration.

Figure 3.1 shows the fire front position predicted by WRF-SFIRE when using the different grid resolutions. The green filled shape corresponds to the observed final burnt area after the 6-hour event and the dotted lines are the simulated fire front positions after the event for each configuration in Table 3.1. As one can observe, the atmospheric grid resolution plays a relevant role in the final result in terms of quality. Moving to higher spatial resolutions improves the match with the final observed perimeter, especially at the head of the fire. There is a significant change in behavior on the flanks of the fire only when reaching the higher atmospheric resolutions (59m). The resolution does not have a significant impact at the rear part of the fire as expected since the propagation is negligible in this part of the fire. The main reason for these results is the capacity of WRF to better detect local wind convections due not only to the atmospheric effects but also because of the wildfire. Figures 3.2 and 3.3 show the wind speeds and wind directions at two different instants of the fire propagation. Figure 3.2 depicts wind conditions before the fire ignition and Figure 3.3 shows wind parameters five hours later. In both cases, the four grid configurations mentioned above were used to determine not only the influence of the coupled system on the atmospheric behavior, but also the precision in detecting such interaction. In Figure 3.2, independent of the grid resolution, the only wind observed is the one generated by the WRF model. However, when considering the effect of the atmospheric grid resolution once the fire has started (see Figure 3.3), one can clearly observe that the results are quite different. Increasing the atmospheric resolution allows the model to capture the winds generated by the fire itself. Therefore, a significant quantity of atmospheric grid points is necessary to ensure accuracy in the results. However, high resolution simulations imply longer execution times (Figure 3.4). Therefore, it is necessary to use high performance computing resources in order to achieve as much resolution as possible. Since the simulations that provide better quality results are the ones with atmospheric grid resolutions equal to 100 and 59 meters, the next analysis is focused on these two cases.

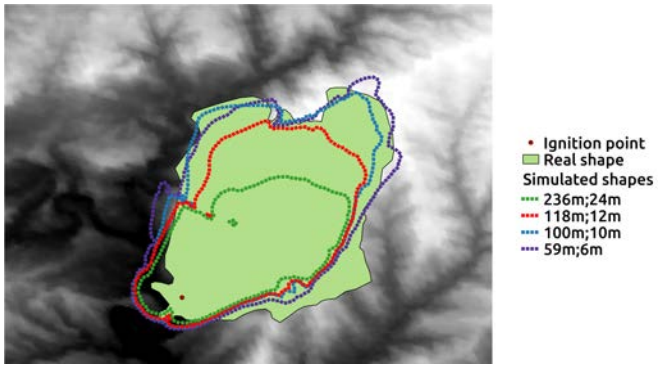


Figure 3.1: Cardona wildfire final burnt area compared to the simulated fire fronts when using WRF-SFIRE with atmospheric grid resolutions equal to 236, 118, 100 and 59 meters.

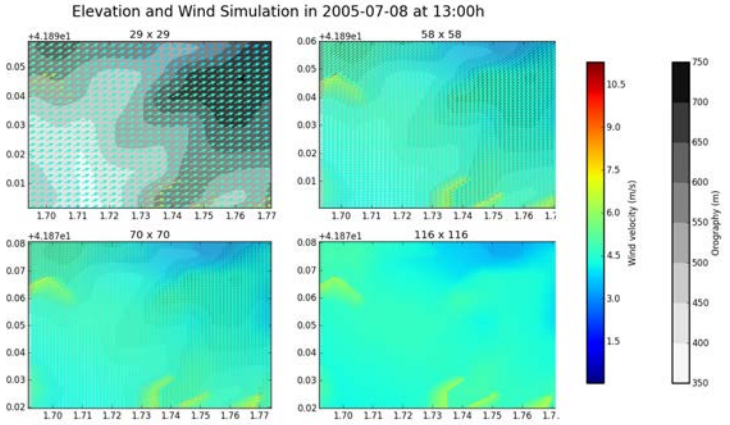


Figure 3.2: Graphical representation of the wind speed and wind direction at fire initial time using WRF-SFIRE for the four experiments described in Table 3.1.

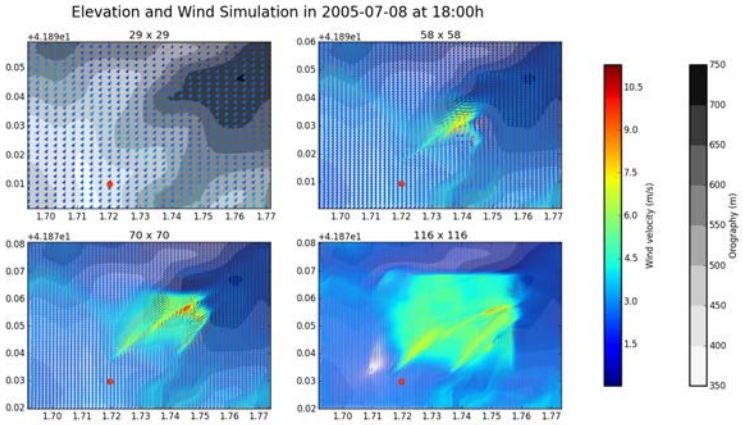


Figure 3.3: Graphical representation of the wind speed and wind direction five hours after the fire starts using WRF-SFIRE for the four experiments described in Table 3.1. The red dots are the ignition points.

Figure 3.4: Total, WRF, and SFIRE serial execution times simulating the Cardona fire using WRF-SFIRE with atmospheric grid horizontal resolutions equal to 236, 118, 100 and 59 meters.

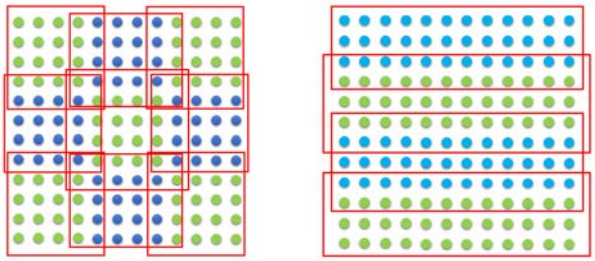


Figure 3.5: Patches division and tiles division of a domain depicted in the horizontal plane with 16x16 points in one vertical level.

3.1.2 Computer characteristics and baseline serial configuration

The execution platform used for the experiments with shared memory is a multi-core system composed of 2 sockets integrating Intel Xeon processors with 8 cores and multi-threading. On the other hand, the execution platform when running the model in its MPI version, consists of a cluster of 24 compute nodes each equipped with two Intel Xeon E5-2670 Sandy Bridge CPUs with 16 cores each. In order to analyze the parallelization improvements and the scalability of the model, the simulation is first executed using WRF-SFIRE in its baseline serial configuration. Figure 3.4 shows the execution time spent when simulating the Cardona fire using a single thread approach for all tested grid resolutions. Moreover, the execution time spent for each individual model (WRF and SFIRE) is also depicted. As one can observe, higher resolution imply longer execution time varying from, approximately, 17 hours to 7 days depending on the resolution used. All the execution times obtained are prohibitively slow for operational purposes.

3.2 Shared memory study

Shared memory parallelization in WRF-SFIRE is programmed using OpenMP. OpenMP is an Application Program Interface (API) for parallel programming in shared memory systems. In an OpenMP parallelization, a process is divided in different threads, which have the same globally shared memory. Therefore, data transfer is transparent to the programmer and synchronization takes place but it is mostly implicit. Using this structure, one can work simultaneously in different parts of the domain and the job is parallelized.

In the particular case of the OpenMP parallelization in WRF-SFIRE, some piece of the domain is divided into tiles which are horizontal divisions of the domain piece which are sharing memory (Figure 3.5). Therefore, the computations in this domain piece can be done in parallel in each of the tiles. The number of tiles are set to the amount of threads that will be run the model.

In this study, the Cardona fire experiment is run using 2, 4, 8 and 16 threads locating each thread in a different core and with atmospheric grid resolutions of 100 and 59 meters. The WRF and SFIRE execution times are collected for each configuration. Then, the ideal execution time T_i is computed using the total serial execution time T_s (from Figure 3.4) and the number of parallel threads N as

$$T_i = \frac{T_s}{N}.$$

Figure 3.6 shows the ideal execution time (green line), the real total execution time (blue line) and the execution time of the two models WRF and SFIRE (bar chart) when using 2, 4, 8 and 16 threads for an atmospheric grid resolution of 100 meters. Figure 3.7 depicts the same information as in Figure 3.6 but for an atmospheric grid resolution of 59 meters. As one can observe, the scalability of the system is reasonable because the total execution time has almost the same tendency as the ideal case. However, in terms of absolute execution time incurred in both cases, it is clear that the one that could reach the execution time requirements for operational purposes is the experiment with the atmospheric grid resolution equal to 100 meters. Nevertheless, both experiments are too slow to be used in an operational setting.

3.3 Distributed memory study

Distributed memory parallelization in WRF-SFIRE is programmed using MPI. MPI or Message Passing Interface is a communication protocol to run parallel distributed systems. In a MPI framework, different processes run different parts of a job. The processes are independent and have their own piece of memory. All the process run in parallel. After each computation, all the processes send each other the important information to keep working properly. The synchronization among all the processes usually is the bottleneck using this parallel paradigm. Therefore, it is important to reduce as much as possible the number of synchronizations and communications between processes in order to have an efficient code.

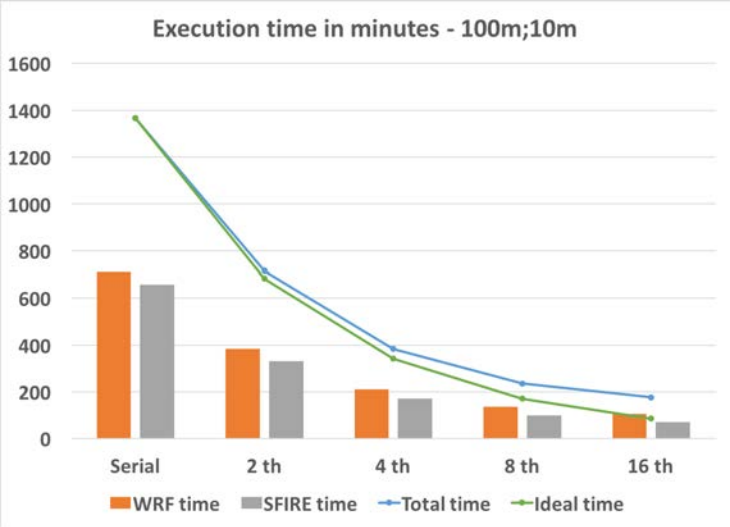


Figure 3.6: Ideal, WRF, SFIRE and total execution time in minutes simulating the Cardona fire using WRF-SFIRE with an atmospheric grid resolution of 100 meters and OpenMP parallelization with 2, 4, 8 and 16 threads.

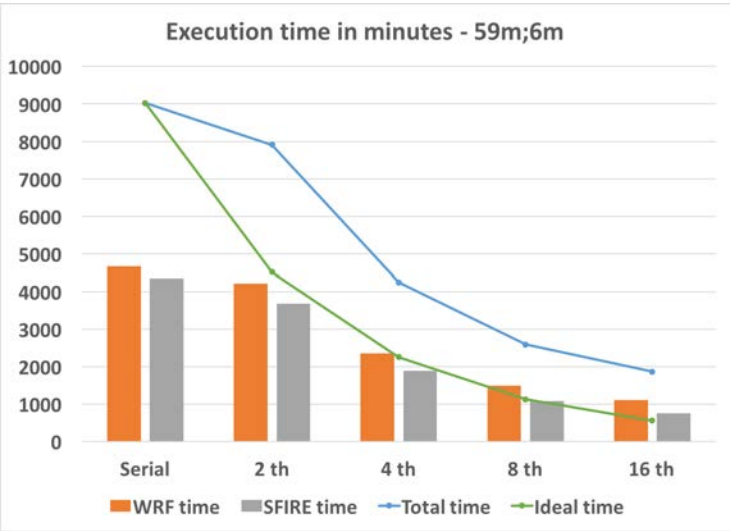


Figure 3.7: Ideal, WRF, SFIRE and total execution time in minutes simulating the Cardona fire using WRF-SFIRE with an atmospheric grid resolution of 59 meters and OpenMP parallelization with 2, 4, 8 and 16 threads.

The MPI parallelization in WRF-SFIRE is done through patches, that is, the domain is divided into a fixed number of parts according to the amount of cores available for a given simulation. These patches usually are square divisions of the domain using the horizontal plane as a reference (Figure 3.5). Since each patch needs information from its neighbors' patches to run the model, each patch includes extra points (called halo) to incorporate those points from the four boundaries that are required to execute during one iteration of the model. After finishing each iteration, the patches must exchange the results from the points in the halo, therefore, a synchronization barrier is required. This scheme implies that all MPI processes proceed in a synchronized fashion which implies a non-depreciable communication time if the patch size is not well evaluated.

In this study, the Cardona fire experiment is run using 48, 96 and 144 cores and with atmospheric grid resolutions of 100 and 59 meters. The WRF and SFIRE execution times are collected again for each configuration. Then, the ideal execution time T_i is computed using the total serial execution time T_s (from Figure 3.4) and the number of cores N from the equation (3.2).

Figures 3.8 and 3.9 show the ideal execution time, the real total execution time, and the execution time of the two models when using 48, 96 and 144 cores in a MPI distributed memory parallelization. It is possible to observe that execution time is significantly reduced, but it is very different from the ideal execution time. Furthermore, the CPU hours spent are large because this execution time is distributed among all the cores. Therefore, it is important also to take into account which cases provide an accurate result, while minimizing the computational cost in terms of CPU hours. Another limitation of MPI is that the domain is partitioned in patches. Therefore, if the number of MPI processors increases, the number of patches increases, and so the size of the patches decreases. There is a minimum size of the patches which in turns limits the number of MPI processors which can be used. Therefore, the maximum number of MPI processors is limited by the size of the domain. However, it seems that using an adequate set up and parallelization strategy, it could be possible to reach operational times.

3.4 Summary

Finally, both parallel paradigms OpenMP and MPI are compared. Therefore, the ideal time is compared to the parallel execution time T_p in terms of the percentage efficiency E as

$$E = \frac{T_i}{T_p} \cdot 100, \tag{3.1}$$

for all the previous cases.

One can observe in Table 3.2, that operational times can only be found using MPI distributed memory parallelization. However, the percentage efficiency is always below 40% showing that the computational cost in terms of CPU hours is too large using this parallel paradigm. Using OpenMP shared memory parallelization, the total execution time is always far from real-time and the efficiency, in the case of 16 OpenMP threads, is always below 50%. So, using OpenMP threads also results in a big computational

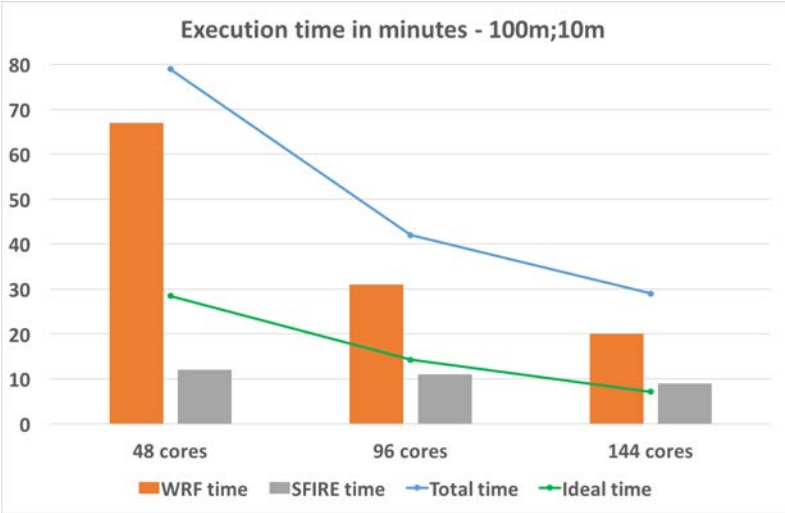


Figure 3.8: Ideal, WRF, SFIRE and total execution time in minutes simulating the Cardona fire using WRF-SFIRE with an atmospheric grid resolution of 100 meters and MPI parallelization with 48, 96 and 144 cores.

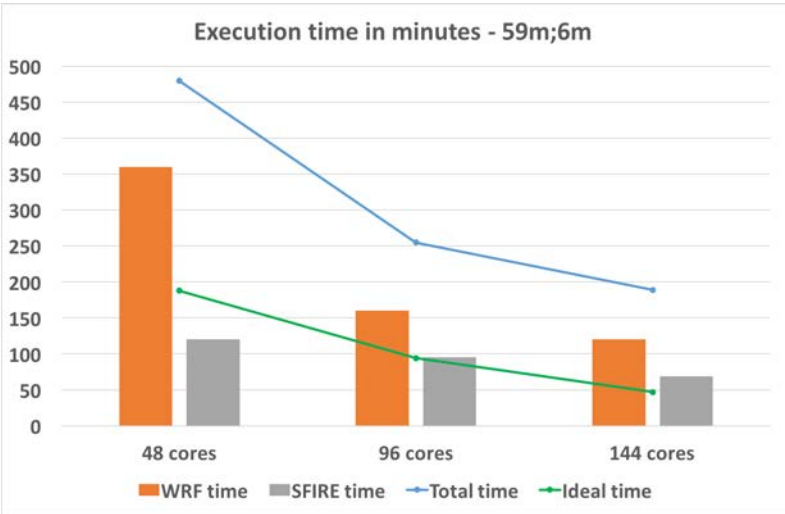


Figure 3.9: Ideal, WRF, SFIRE and total execution time in minutes simulating the Cardona fire using WRF-SFIRE with an atmospheric grid resolution of 59 meters and MPI parallelization with 48, 96 and 144 cores.

Resolutions	Configuration	Total	WRF	SFIRE	Ideal	Efficiency
100m;10m	Serial	1366	709	657		
100m;10m	2 OMP threads	716	384	332	683	95%
100m;10m	4 OMP threads	383	212	171	342	89%
100m;10m	8 OMP threads	236	137	99	171	72%
100m;10m	16 OMP threads	176	106	70	85	48%
100m;10m	48 MPI cores	79	67	12	28	35%
100m;10m	96 MPI cores	42	31	11	14	33%
100m;10m	144 MPI cores	29	20	9	7	24%
59m;6m	Serial	9026	4673	4353		
59m;6m	2 OMP threads	7907	4220	3687	4513	57%
59m;6m	4 OMP threads	4250	2359	1891	2257	53%
59m;6m	8 OMP threads	2593	1501	1092	1128	44%
59m;6m	16 OMP threads	1870	1114	756	564	30%
59m;6m	48 MPI cores	480	360	120	188	39%
59m;6m	96 MPI cores	255	160	95	94	37%
59m;6m	144 MPI cores	213	141	72	47	22%

Table 3.2: Summary of the execution times, in minutes, using the different parallel configurations for the Cardona fire simulation. The last column is the percentage efficiency computed from equation (3.1) using the ideal execution time and the total execution time.

cost. In summary, the best configuration in order to get predictions in operational times is found using distributed memory parallelization through MPI. However, the computational cost in terms of CPU hours is huge and the maximum number of MPI processors is limited by the size of the domain (patches size limitation).

Chapter 4

Level Set Method for heterogeneous Rate of Spread

In this chapter, the problem when applying the level set method explained in Section 2.2 with strong and heterogeneous rate of spread is shown. Then, a solution avoiding local minima in the level set method is proposed. The resulting approach was merged with the main branch of the WRF-SFIRE Github repository.

4.1 Spurious ignitions

In [42] was observed that if, in the level set method, a local minimum appears on the boundary, its value keeps decreasing out of control. It is found out that this can in fact happen anywhere in the presence of spatially highly variable rate of spread.

The partial differential equation to evolve the level set function (2.3) can be rewritten as

$$\frac{\partial \psi}{\partial t} = -R||\nabla \psi||, \tag{4.1}$$

where one can observe that the right hand side of the equation is a negative magnitude. Therefore, if there is a large rate of spread at a specific point surrounded by small rates of spread, the level set function is going to evolve downward at that point. This effect causes local minima of the level set function which conceptually means burning took place in the past. Therefore, this situation ends in a physically unrealistic fire arrival time causing spurious ignitions to be created. Usually, the rate of spread is highly heterogeneous because of local effects of the atmosphere on the fire or an heterogeneous fuel (Figure 4.1). Therefore, a solution for this situation is required in order to avoid these local minima.

In order to properly analyze the problem, it is important to choose a test case for which the simulation produces spurious ignitions. The real case analyzed is a wildfire which occurred in Mariposa County, California on July 16, 2017 [51]. The 2017 Detwiler fire burned a total of 33113 hectares and lasted for 14 days. The fire started at 3:56 am and it kept until 14 days later. This particular case has a highly heterogeneous rate of spread, which causes spurious fires when using the level set method.

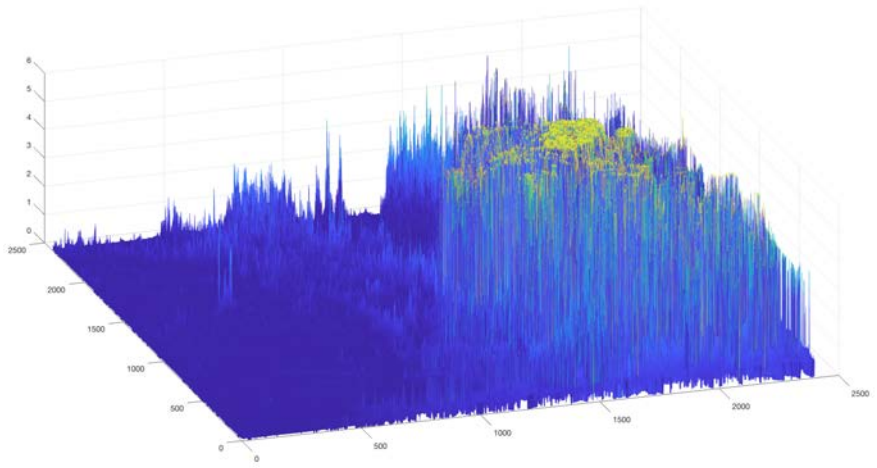


Figure 4.1: Heterogeneous rate of spread computed from semi-empirical Rothermel's formula (2.1) for the Detwiler fire occurred in California.

In this case, one can observe an heterogeneous rate of spread with large magnitude in the right side of Figure 4.1. In Figure 4.2, one can observe the level set function after some iterations of WRF-SFIRE for the Detwiler fire. This heterogeneous rate of spread causes some local minima to develop on the same side of the level set function which causes some spurious fire ignitions. Therefore, the level set method ends up with an unrealistic fire arrival time causing new ignitions before the actual ignition time of the fire which can be seen at the tip of the cone.

4.2 Avoiding local minima approach

In order to solve the problem stated in the previous section, it is proposed to avoid creation of local minima. The solution attempts to ascertain if the level set function at a point $\vec{x}$ and time τ tends to go lower than its neighbors. In this case, the level set function at this point and time is going to be defined as

$$\psi(\vec{x},\tau) = \min_{\vec{y} \text{ neighbor of } \vec{x}} \psi(\vec{y},\tau), \tag{4.2}$$

where $\vec{y}$ neighbor of $\vec{x}$ means that the point $\vec{y}$ is at one mesh step from $\vec{x}$ in either of the two dimensional directions.

This replacement gives a solution to the previous problem because it avoids evolving the level set function at a particular point to a point below its neighbors, i.e. it avoids creating local minima. This prevents getting an unrealistic situation in the fire arrival time. This restriction is defined by the fact that the fire arrival time cannot evolve to the past. Applying this new approximation, one can observe in Figure 4.3 that the local minima approach gives a realistic result without spurious ignitions for the same fire as in the previous example.

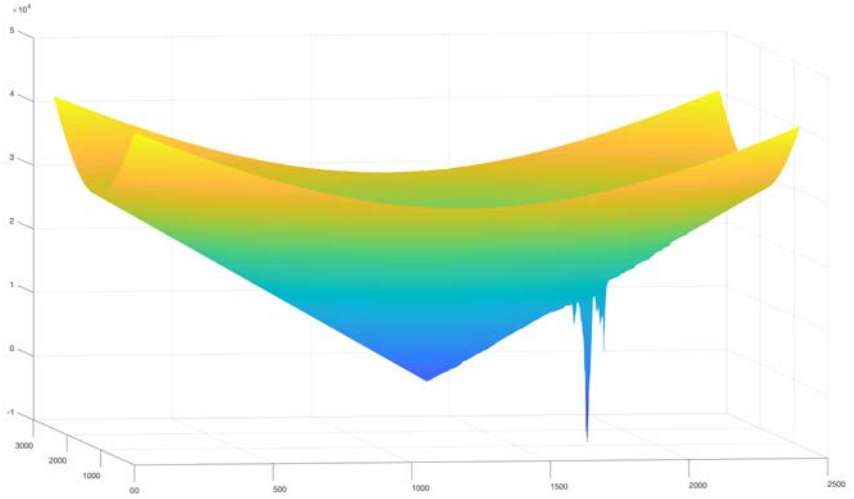


Figure 4.2: The level set function after few iterations of WRF-SFIRE for the Detwiler fire which has the rate of spread in Figure 4.1.

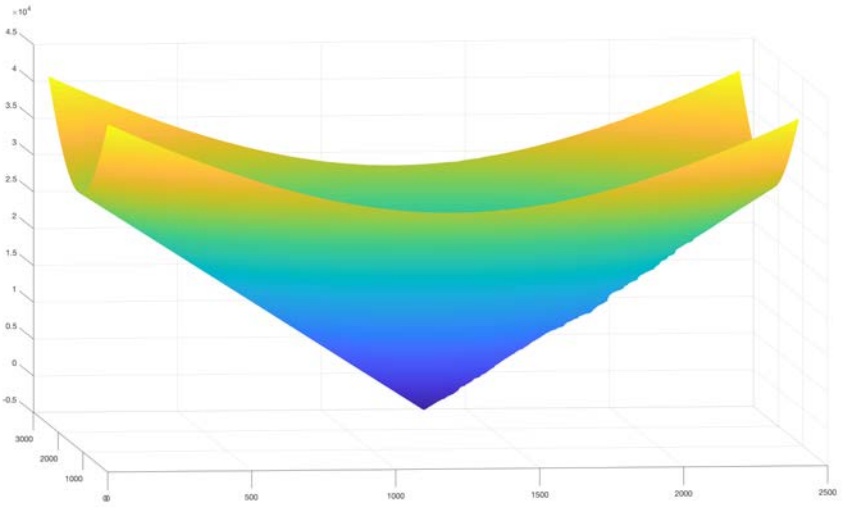


Figure 4.3: The improved level set function after few iterations of WRF-SFIRE for the Detwiler fire which has the rate of spread in Figure 4.1. Note that creation of local minima is avoided by this new approach.

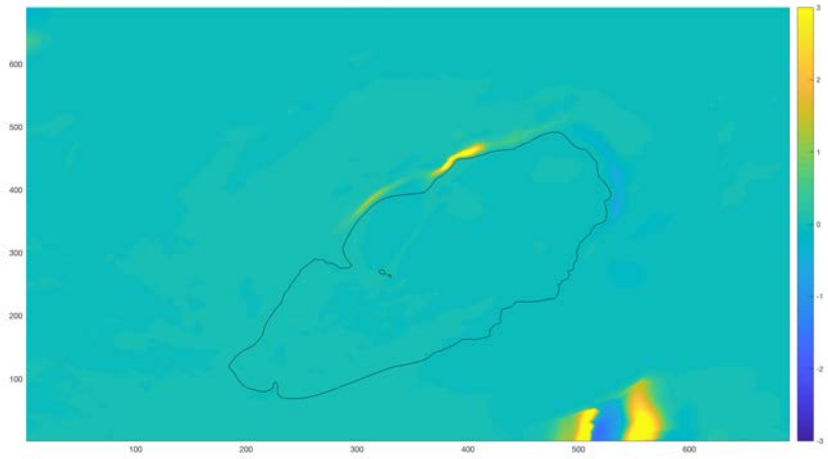


Figure 4.4: Difference between the level set function using and not using the proposed solution to avoiding local minima in the fire arrival time applied to the Cardona fire in Section 3.1.1.

4.3 Consistency of the solution

In this section the consistency of the proposed solution is tested. The Cardona fire from Section 3.1.1, which did not have spurious ignitions, is tested using the new local minima approach and comparing it with the previous results. Figure 4.4 shows the differences between not using the proposed solution or using it after some iterations of WRF-SFIRE. The results show little differences in one of the boundaries and later instants outside the fire perimeter at this particular time. However, in all the time steps, the fire front is identical in both cases. Therefore, one can conclude that the new proposed solution avoiding local minima does not change the simulated result significantly. In conclusion, the new proposed solution is a consistent approach.

Chapter 5

Interpolation of the Fire Arrival Time

In the previous sections, one can observe that WRF-SFIRE is a simulator which couples an atmospheric model, WRF, and a fire propagation model, SFIRE. In such a coupled model, atmosphere and fire evolve together, influencing each other. Since the atmosphere and fire evolve simultaneously in the model, incorporating real world observations about the fire can lead to inconsistencies when simulated fire is forced to match the observations. In these situation, the simulated atmosphere and fire are incompatible. Therefore, to modify the simulated fire in a consistent way, it is necessary to evolve the atmosphere from moment of the ignition to the instant of the observed data. Thus, it is necessary to construct an interpolated evolution of the fire from the ignition to the observed data in order to define the atmosphere using the heat fluxes from the defined fire. Once there is a consistency between atmosphere and fire, the coupled simulation can continue from the observed data.

This chapter proposes a new method to interpolate the fire from the ignition point $\vec{x}_i$ at time T_i to some observed perimeters $\Gamma_1, \ldots, \Gamma_N$ at times $T_1, \ldots, T_N$. The main goal is to simulate the coupled atmosphere-fire model from the last observed perimeter Γ_N in a consistent manner. Therefore, it is necessary to interpolate the fire arrival time from $\vec{x}_i$ to Γ_1, from Γ_1 to Γ_2 and so on.

5.1 Proposed work-flow of the interpolation method

The work-flow of the new fire arrival time interpolation technique is defined as:

1. Run a first simulation of WRF-SFIRE from moment of the ignition, T_i, to some time after the fire arrival time at the first observed perimeter Γ_1, T_1.

2. Use the variables from the model to construct a reasonable rate of spread R using the fire spread model of the equation (2.1).

3. Interpolate the fire arrival time by using the rate of spread R computed in the previous step, knowing that the fire arrival time at the ignition point is T_i and the fire arrival time at the first observed perimeter Γ_1 is T_1.

4. Spin up the atmosphere in WRF-SFIRE using the previous interpolated fire arrival time until the fire arrival time at the first observed perimeter Γ_1, T_1.

5. Go to step 1 and use T_1 instead of T_i and T_2 instead of T_1. Repeat the cycle until T_N.

6. Run WRF-SFIRE coupled from the last observed perimeter Γ_N at T_N.

Therefore, the main goal of this chapter is finding the interpolation in step 3.

5.2 Fire spread model

The state of the fire spread model is the fire arrival time $T(\vec{x})$ at locations $\vec{x}$ in a rectangular simulation domain $\Omega \subset \mathbb{R}^2$. The isoline $T(\vec{x}) = \tau$ is then the fire perimeter at time τ. The normal vector to the isoline is $\nabla T / \|\nabla T\|$. The rate of spread in the normal direction and the fire arrival time at a location on the isoline then satisfy the eikonal equation

$$\|\nabla T\| = \frac{1}{R}. \tag{5.1}$$

It is assumed that R depends on location (because of differing fuels and terrain) and time (because of wind and fuel moisture changing with time). Rothermel's model [65] for 1D fire spread formulated in equation (2.1) gives an expression for the rate of spread R. The 1D model is adapted to spread over the 2D landscape by postulating that the wind factor and the slope factor are functions of the components of the wind vector and the terrain gradient in the normal direction. Thus,

$$R = R\left(\vec{x}, T(\vec{x}), \nabla T(\vec{x})\right).$$

The fire spread model is coupled to an atmospheric model. The fire emits sensible and latent heat fluxes, which change the state of the atmosphere, and the changing atmospheric conditions in turn impact the fire (Figure 1.1). Wind affects the fire directly and temperature, relative humidity, and rain affect the fire indirectly through changing fuel moisture.

The fire model is implemented on a rectangular mesh by finite differences. For numerical reasons, the gradient in the eikonal equation (5.1) needs to be computed by an upwinding-type method [53], which avoids instabilities caused by breaking causality in fire propagation: for the computation of ∇T at a location $\vec{x}$, only the values from the directions that the fire is coming from should be used, so the methods switch between one-sided differences depending on how the solution evolves. Sophisticated methods of an upwinding type, such as ENO or flux-limiters [59], aim to use more accurate central differences and switch to more stable one-sided upwind differences only as needed. Unfortunately, the switching causes the numerical gradient of T at a mesh node to become a nondifferentiable function of the values of T at that point and its neighbors. In addition, a penalty term is added to prevent the creation of local minima. It was shown in Chapter 4 that level set function can decrease out of control in the presence of spatially highly variable rate of spread.

Figure 5.1: Graphical representation of the problem proposed in Chapter 5. An interpolation of the fire arrival time between two perimeters Γ_1 and Γ_2 at times $T_1 < T_2$ is represented by blue arrows.

5.3 Minimal residual formulation

5.3.1 Eikonal equation

The main goal is to define the fire arrival time between two consecutive perimeters Γ_k and Γ_{k+1} at times T_k and T_{k+1} respectively. A special case is to interpolate from the ignition point $\vec{x}_i$ at time T_i to the first perimeter Γ_1 at time T_1. However, one can solve this problem in the same way as interpolating between two perimeters. Therefore, without loss of generality, consider the situation when the two observed fire perimeters Γ_1 and Γ_2 at times $T_1 < T_2$ are known, and the fire progression between the two perimeters is of interest (Figure 5.1).

In this situation, a function $T : \mathbb{R}^2 \longrightarrow \mathbb{R}$, defining for each domain point $\vec{x} \in \Omega \subset \mathbb{R}^2$ a fire arrival time $T(\vec{x}) \in \mathbb{R}$, is required. This problem has the constraints that for all points $\vec{x}_1 \in \Gamma_1$, then $T(\vec{x}_1) = T_1$; and $\vec{x}_2 \in \Gamma_2$, then $T(\vec{x}_2) = T_2$. These fire arrival times define the level set function which is going to drive the coupled simulation. This is going to be done using a spin-up of the atmosphere from the heat fluxes produced by the interpolated fire arrival times.

The fire arrival time is defined using the rate of spread R. It is well known that the rate of spread is a positive velocity defined as the distance traveled by the fire in the normal direction, divided by the time spent. Thus,

$$R = \left| \frac{\text{Distance traveled}}{\text{Time spent}} \right| = \left| \frac{\Delta \vec{x}}{\Delta T} \right| = \frac{1}{||\nabla T||}.$$

This is the eikonal equation (5.1) which is solved in this chapter.

5.3.2 Nonlinear optimization problem

In this section, a nonlinear constrained optimization problem is formulated in order to solve the eikonal equation (5.1) approximately

$$||\nabla T|| \approx \frac{1}{R}, \tag{5.2}$$

subject to

$$T \approx T_1 \text{ in } \Gamma_1$$

and

$$T \approx T_2 \text{ in } \Gamma_2.$$

The reason for using an approximated solution is to give a necessary flexibility to the fire arrival time T. Furthermore, it is assumed that the rate of spread R once computed from the prognostic variables of the system is constant over the whole domain.

Therefore, the problem (5.2) is solved using the nonlinear constrained optimization problem

$$\begin{aligned} &\min_{T} \ J(T) \\ &T = T_1 \text{ at } \Gamma_1 \\ &T = T_2 \text{ at } \Gamma_2, \end{aligned} \tag{5.3}$$

where $J(T)$ is the objective function to be minimized

$$J(T) = \left(\int_{\Omega} \left| f(||\nabla T||_2^2, R^2) \right|^p \right)^{1/p}.$$

Therefore, $J(T)$ is constrained using the p norm of the residuals of the eikonal equation. The reason for using the p norm is that the p norm is differentiable and when p goes to infinity, the p norm is equivalent to the maximum norm (the maximum norm is not differentiable). A value of $p = 4$ is used in all the experiments in Section 5.5. The norm of the gradient of T is squared in order to avoid the square root and so, R is also squared. The residuals are defined using a function $f : \mathbb{R}^2 \longrightarrow \mathbb{R}$ such that

$$f(||\nabla T||_2^2, R^2) = 0 \iff ||\nabla T|| = \frac{1}{R}. \tag{5.4}$$

Therefore, the function f is 0 when the eikonal equation is solved. Thus, minimizing $J(T)$ is the same as solving the eikonal equation approximately.

For stability reasons, in all the experiments f is defined as

$$f(x, y) = xy - 1.$$

However, other functions such as

$$\begin{aligned} f(x, y) &= x - \frac{1}{y}, \\ f(x, y) &= \frac{1}{x} - y, \\ f(x, y) &= \log(xy), \\ f(x, y) &= e^{xy-1} - 1, \end{aligned}$$

have advantages in some situations. Thus, it is possible to define f as desired, complying with the restriction in equation (5.4). Finally, there are no boundary conditions imposed on the boundary of Ω.

5.3.3 Nonlinear optimization problem with dynamic rate of spread

In the eikonal equation (5.1), one can observe that the fire arrival time and the rate of spread have a mutual dependency. So, when the fire arrival time is modified by the rate of spread during the minimization, this new defined fire arrival time should affect the rate of spread imposed in the minimization. Therefore, this idea gives a new nonlinear optimization problem defined in this section.

The main idea is to solve the same nonlinear optimization problem in Section 5.3.2 but using a dynamic rate of spread. Therefore, the nonlinear constrained optimization problem in equation (5.3) is solved considering the rate of spread depends on the fire arrival time in the minimization. This gives rise to a slightly modified objective function

$$J(T) = \left(\int_\Omega \left| f(||\nabla T||_2^2, R(T)^2) \right|^p \right)^{1/p}.$$

Therefore, a change in the fire arrival time T is going to affect the rate of spread $R(T)$. This new consideration gives more reality to the optimization problem because the rate of spread depends on the fire arrival time as shown in the previous sections. So, the eikonal equation (5.1) is solved approximately and simultaneously for T and R.

5.4 Numerical minimization of the residuals

5.4.1 Discretization and the constraint matrix

The coupled atmosphere-fire simulation is done on a uniformly spaced $m \times n$ rectangular grid (aligned approximately with longitude and latitude). Therefore, the nonlinear constrained optimization problem (5.3) formulated in the previous section has to be solved on that grid. Thus, a discretization of the objective function $J(T)$ and the constraints

$$T = T_1 \text{ at } \Gamma_1, \quad T = T_2 \text{ at } \Gamma_2 \tag{5.5}$$

are used.

The discretization of the objective function $J(T)$ on the $m \times n$ mesh is

$$J(T) = \left(\int_\Omega \left| f(||\nabla T||_2^2, R^2) \right|^p \right)^{1/p} \approx \left(\Delta x \cdot \Delta y \cdot \sum_{i=1}^{m} \sum_{j=1}^{n} \left| f(||\nabla T_{ij}||_2^2, R_{ij}^2) \right|^p \right)^{1/p}, \tag{5.6}$$

where Δx and Δy are the x and y mesh spacings, $T_{ij} = T(x_{ij})$, and $R_{ij} = R(x_{ij})$.

A term μ is also added to the objective function, which penalizes local minima in the fire arrival time. This term is added to the objective function in order to avoid the fire

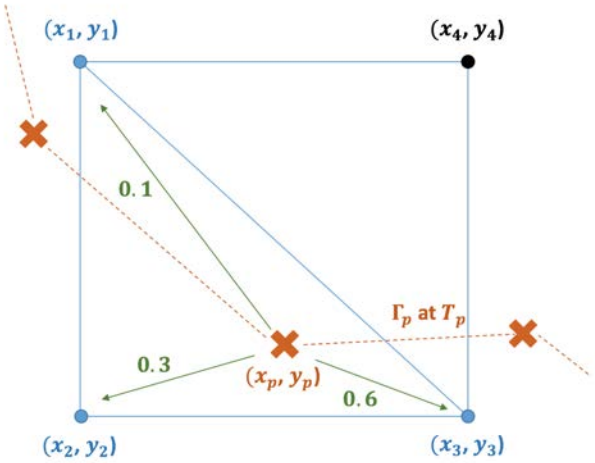


Figure 5.2: Graphical explanation of the barycentric interpolation where (x_p, y_p) is a perimeter point from the shapefile Γ_p at time T_p and (x_1, y_1), (x_2, y_2), (x_3, y_3) are the three grid points with nonzero coefficients for the interpolation of (x_p, y_p). The grid point (x_4, y_4) is an example of a point in the mesh with coefficient equal to 0 for the interpolation of (x_p, y_p).

arrival time going down because of spatially highly variable rate of spread (Chapter 4). This penalization term is formulated as

$$\mu = K \cdot \sum_{i=1}^{m} \sum_{j=1}^{n} \max \left\{ \min \left\{ T_{i-1,j} - T_{ij}, T_{i+1,j} - T_{ij}, T_{i,j-1} - T_{ij}, T_{i,j+1} - T_{ij} \right\}, 0 \right\},$$

where K is a scale constant. In all of the experiments, $K = 10$.

The constraints of the problem are the ignition point at time T_i and the observed perimeters Γ_p at time T_p. They are shapefiles, i.e., collections of points on the perimeter $\vec{x}_p = (x_p, y_p)$ which do not necessary lie on the mesh (Figure 5.2). Therefore, it is necessary to redefine the nonlinear constrained optimization problem (5.3) as

$$\min_{T} \; J(T) \tag{5.7}$$

$$\text{subject to } HT = g,$$

where $HT = g$ is a linear system of constraints. H is a sparse matrix and since the points in the shapefiles do not need to lie on the grid, the rows of H are the coefficients of an interpolation from the grid to the points in the shapefiles which define the perimeters. Each column of H corresponds to a different node on the $m \times n$ mesh. The right hand side of the linear system, g, is a vector with the fire arrival times of the perimeters for each row interpolated. This vector has the same length as the number of rows in the matrix H (number of points in the perimeters).

The coefficients of the interpolation matrix H are defined from barycentric interpolation. The rectangles of the grid are split into two triangles each, and, for each triangle, we compute the barycentric coordinates of the points in the shapefile, i.e., the

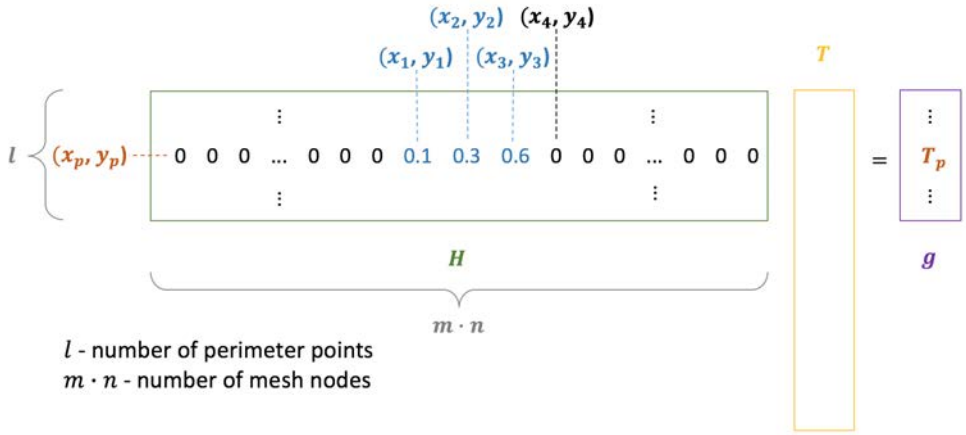


Figure 5.3: Graphical explanation of the linear system of constraints $HT = g$ from the example in Figure 5.2.

coefficients of the unique linear combination of the vertices of the triangle that equals to the point in the shapefile. Therefore, each point in the perimeter (x_p, y_p) is contained in a triangle with vertices (x_1, y_1), (x_2, y_2), (x_3, y_3) on the mesh and is interpolated to these mesh nodes by finding the coefficients α_1, α_2, α_3 which satisfy the system of equations

$$\begin{aligned} \alpha_1 x_1 + \alpha_2 x_2 + \alpha_3 x_3 &= x_p \\ \alpha_1 y_1 + \alpha_2 y_2 + \alpha_3 y_3 &= y_p \\ \alpha_1 + \alpha_2 + \alpha_3 &= 1. \end{aligned}$$

In the example of Figure 5.2, the point in the perimeter (x_p, y_p) is defined as a linear combination of the three points in the triangle (x_1, y_1), (x_2, y_2) and (x_3, y_3); and the coefficients $\alpha_1 = 0.1$, $\alpha_2 = 0.3$ and $\alpha_3 = 0.6$. Therefore, each of the rows of H is going to have at most 3 nonzero entries corresponding to the coefficients α_1, α_2, α_3. Figure 5.3 depicts how one of the rows of the interpolation operator matrix H and the right hand side g are defined for the example in Figure 5.2.

If all 3 barycentric coordinates are in $[0, 1]$, the point is contained in the triangle, the barycentric coordinates are the sought interpolation coefficients, and they form one row of H. For efficiency, most points in the shapefile are excluded up front, based on a comparison of their coordinates with the vertices of the triangle, which is implemented by a fast binary search.

When there is more than one point of the shapefile in any triangle, these are condensed into a single constraint, obtained by averaging the interpolated rows of H inside the same triangle of grid nodes. Therefore, the matrix H is going to have only the relevant rows. This way, over constraining the fire arrival time near the perimeter is avoided, which should be avoided for the same reason as limiting the number of constraints in mixed finite elements to avoid locking, cf., e.g., [8]. Furthermore, the linearly dependent constraints are removed and the matrix H of the linear system of constraints has full row rank.

5.4.2 Initial approximation

In order to solve the nonlinear constrained optimization problem (5.7) using a descent method, as will be shown in the following sections, it is vital to have a reasonable initial approximation of the fire arrival time satisfying the previous linear system of constraints $HT = g$. Thus, the quadratic minimization problem

$$I(T) = \frac{1}{2}\int_{\Omega} \|(-\triangle)^{\alpha} T\|^2 \, dxdy \rightarrow \min_{T} \text{ subject to (5.5) and } T|_{\partial\Omega} = T_{\max} \tag{5.8}$$

is solved, where $\triangle = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}$ is the Laplace operator, $T_{\max}$ is the maximum value of the fire arrival time at the boundaries, and $\alpha > 1$ is generally non-integer.

The reason for choosing $\alpha > 1$ is that $\sqrt{I(T)}$ is the Sobolev $W^{\alpha,2}(\Omega)$ seminorm and in 2D, the space $W^{\alpha,2}(\Omega)$ is not embedded in continuous functions if and only if $\alpha > 1$. Consequently, $I(T)$ is not a bound on the value $T(\vec{x})$ at any particular point, only averages over some area can be controlled. Numerically, when $\alpha = 1$, minimizing $I(T)$ with a point constraint, such as an ignition point, results in T taking the shape of a sharp funnel at that point (Figure 5.13), which becomes thinner as the mesh is refined. This situation is definitely undesirable because it can result in a non-differentiable first approximation of the fire arrival time.

The discrete form of (5.8) is

$$\frac{1}{2}\langle ST, T\rangle \rightarrow \min_{T} \text{ subject to } HT = g \text{ and } T = T_{\max} \text{ at } \partial\Omega, \tag{5.9}$$

where $S = A^{\alpha}$ with $-A$ a discretization of the Laplace operator with Dirichlet boundary conditions.

To solve (5.9), the first step to make is a change of variable $U = T - T_b$ where T_b is the constant function with value $T_{\max}$. This shifts the solution to a Dirichlet zero boundary problem. Thus, the solution of the optimization problem (5.9) is equivalent to solve

$$\frac{1}{2}\langle S(U + T_b), U + T_b\rangle \rightarrow \min_{U} \text{ subject to } H(U + T_b) = g \text{ and } U = 0 \text{ at } \partial\Omega. \tag{5.10}$$

Since T_b is a constant function, then $ST_b = 0$. Applying this fact and rearranging the linear system of constraints give that solving the optimization problem in equation (5.10) is equivalent to solving

$$\frac{1}{2}\langle SU, U + T_b\rangle \rightarrow \min_{U} \text{ subject to } HU = g - HT_b \text{ and } U = 0 \text{ at } \partial\Omega.$$

Thus, expanding the inner product, Q is defined as

$$Q(U) = \frac{1}{2}\langle SU, U\rangle + \frac{1}{2}\langle SU, T_b\rangle \rightarrow \min_{U} \text{ subject to } HU = g - HT_b \text{ and } U = 0 \text{ at } \partial\Omega.$$

At this point, one can construct the Lagrangian of $Q(U)$ for the equality constraints $HU = g - HT_b$. So,

$$\mathcal{L}(U, \lambda) = Q(U) + \lambda^{\top}(HU - g + HT_b) = \frac{1}{2}U^{\top}SU + \frac{1}{2}T_b^{\top}SU + \lambda^{\top}(HU - g + HT_b).$$

For optimality reasons, $\nabla_U \mathcal{L}(U, \lambda) = 0$ and $\nabla_\lambda \mathcal{L}(U, \lambda) = 0$. Therefore,

$$\nabla_U \mathcal{L}(U, \lambda) = SU + \frac{1}{2} ST_b + \lambda^\top H = SU + \lambda^\top H = 0,$$

and

$$\nabla_\lambda \mathcal{L}(U, \lambda) = HU - g + HT_b = 0.$$

Consequently, solving the optimization problem in equation (5.9) is equivalent to solving the saddle point problem

$$\begin{cases} SU + \lambda^\top H = 0 \\ HU = g - HT_b. \end{cases} \tag{5.11}$$

To solve (5.11), a first feasible solution is found $U_0 = H^\top \left(HH^\top\right)^{-1} (g - HT_b)$, so that $HU_0 = g - HT_b$. Then, one can substitute in the saddle point problem in equation (5.11) $U = U_0 + V$ to get

$$\begin{cases} S(U_0 + V) + \lambda^\top H = 0 \\ H(U_0 + V) = g - HT_b. \end{cases}$$

Thus,

$$\begin{cases} SV + \lambda^\top H = -SU_0 \\ HV = 0 \end{cases} \tag{5.12}$$

Finally, the saddle point problem in equation (5.12) is solved approximately by preconditioned conjugate gradients for the equivalent symmetric positive definite linear system

$$PSPV + \rho (I - P) V = -SU_0, \tag{5.13}$$

where $P = I - H^\top \left(H^\top H\right)^{-1} H$ is the orthogonal projection on the nullspace of H, and $\rho > 0$ is an arbitrary regularization parameter.

Since S is the discretization of the Dirichlet problem, the preconditioner used is

$$M : r \mapsto PS^+ Pr,$$

where S^+ is the inverse of S on the complement of its nullspace.

Finally, the solution is recovered by

$$T = T_b + U_0 + PV.$$

The method only requires access to matrix-vector multiplications by S and S^+, which are readily implemented by sine FFT. To get a reasonable starting point for the descent method iterations it is only necessary to solve (5.13) to low accuracy, but the satisfaction of the constraint $HT = g$ to rounding precision is important.

5.4.3 Projected preconditioned gradient descent method

Once a first approximation is obtained, in order to solve the nonlinear constrained optimization problem (5.7), one can use the gradient descent method or steepest descent method projected to the constraints subspace. In order to apply this iterative nonlinear optimization method, it is necessary to compute the gradient of the objective function proposed in the equation (5.6), $\nabla J(T)$. In that particular case, the objective function defined is not differentiable because of the upwinding method used to compute $||\nabla T||_2^2$. However, in this section, a suitable approximation of ∇T is proposed.

At this point, one needs to know the components of the gradient matrix discretized in a squared $m \times n$ mesh. So, using the chain rule

$$(\nabla J(T))_{kl} = \frac{\partial J(u)}{\partial T_{kl}} = \left(\Delta x \cdot \Delta y \cdot \sum_{i=1}^{m}\sum_{j=1}^{n} \left|f_{ij}\right|^p\right)^{\frac{1-p}{p}} \sum_{i=1}^{m}\sum_{j=1}^{n} \left|f_{ij}\right|^{p-1} \text{sign}(f_{ij}) \frac{\partial f_{ij}}{\partial T_{kl}}, \quad (5.14)$$

where Δx and Δy are the x and y mesh spacings, $T_{kl} = T(x_{kl})$, and $f_{ij} = f(||\nabla T_{ij}||_2^2), R_{ij}^2)$.

In equation (5.14), everything is known except the term $\frac{\partial f_{ij}}{\partial T_{kl}}$ which needs an expansion by the chain rule

$$\frac{\partial f_{ij}}{\partial T_{kl}} = \frac{\partial f_{ij}}{\partial(||\nabla T_{ij}||_2^2)} \frac{\partial(||\nabla T_{ij}||_2^2)}{\partial T_{kl}} + \frac{\partial f_{ij}}{\partial(R_{ij}^2)} \frac{\partial(R_{ij}^2)}{\partial T_{kl}}, \quad (5.15)$$

where $R_{ij} = R(x_{ij})$.

The first factor of each term of the sum are known because they are the partial derivatives of f with respect the first and second components of f, respectively. For instance, if $f(x, y) = xy - 1$, then

$$\frac{\partial f(||\nabla T||_2^2, R^2)}{\partial(||\nabla T||_2^2)} = \frac{\partial f(x, y)}{\partial x} = y = R^2.$$

Thus,

$$\frac{\partial f_{ij}}{\partial(||\nabla T_{ij}||_2^2)} = R_{ij}^2.$$

Similarly,

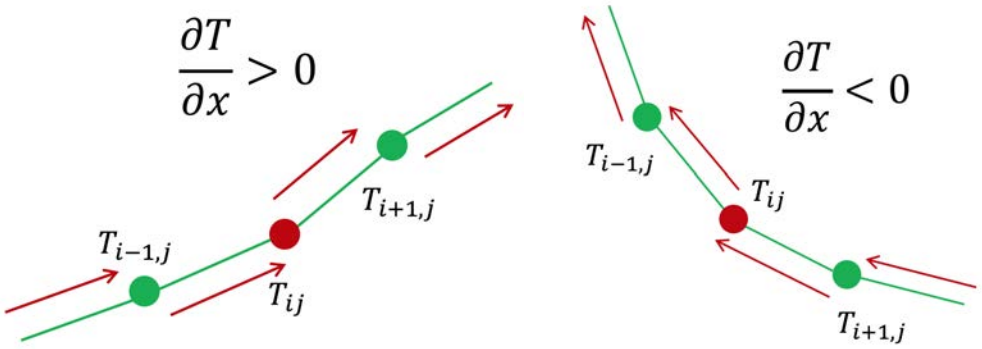
$$\frac{\partial f(||\nabla T||_2^2, R^2)}{\partial(R^2)} = \frac{\partial f(x, y)}{\partial y} = x = ||\nabla T||_2^2.$$

So,

$$\frac{\partial f_{ij}}{\partial R_{ij}^2} = ||\nabla T_{ij}||_2^2.$$

However, one needs to know in more detail how $||\nabla T_{ij}||_2^2$ and R_{ij}^2 are computed in order to interpret their derivatives with respect to T_{kl}. First,

$$||\nabla T||_2^2 = \left(\frac{\partial T}{\partial x}\right)^2 + \left(\frac{\partial T}{\partial y}\right)^2,$$



(a) (b)

Figure 5.4: Example of three consecutive nodes in the x direction of a fire arrival time function $T(x, y)$ with the y direction held constant. (a) The x partial derivative is always positive. (b) The x partial derivative is always negative.

and the partial derivatives $\frac{\partial T}{\partial x}$ and $\frac{\partial T}{\partial y}$ are computed numerically respecting the flow of information, i.e. locations with large value T depend on nearby locations with smaller value T but not the other way round. Therefore, the information flows from the past to the future.

In order to see this, for the remain of this section, the fire arrival time function $T(x, y)$ is going to be illustrated using three consecutive nodes in the x direction and keeping the y direction constant. Everything in this section can be done similarly for the y direction, keeping the direction x constant.

Suppose that $T_{i-1,j}$, T_{ij} and $T_{i+1,j}$ are the three consecutive nodes in Figure 5.4a. The partial derivative with respect to x is computed numerically using the left one-sided partial derivative, i.e.

$$\frac{\partial T_{ij}}{\partial x} \approx \frac{T_{ij} - T_{i-1,j}}{\Delta x} := \text{diff}_x^L(T_{ij}),$$

because as the previous statement says, it is necessary to carry the information from the past to the future.

Another similar example is using $T_{i-1,j}$, T_{ij} and $T_{i+1,j}$ from Figure 5.4b. In that case, one should use the right one-sided partial derivative, i.e.

$$\frac{\partial T_{ij}}{\partial x} \approx \frac{T_{i+1,j} - T_{ij}}{\Delta x} := \text{diff}_x^R(T_{ij}).$$

These previous cases don't have any trouble because

$$\frac{\partial T_{i-1,j}}{\partial x} \cdot \frac{\partial T_{i+1,j}}{\partial x} > 0.$$

However, a problem appears when dealing with borderline cases when the partial derivatives of the neighbors have different signs or are zero, i.e. when

$$\frac{\partial T_{i-1,j}}{\partial x} \cdot \frac{\partial T_{i+1,j}}{\partial x} \leq 0,$$

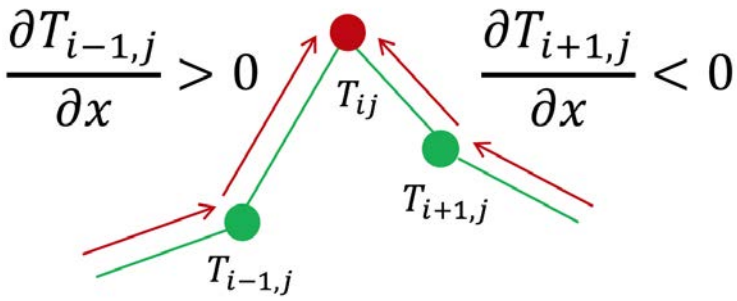


Figure 5.5: Example of three consecutive nodes in the x direction of a fire arrival time function $T(x, y)$ with the y direction held constant. The x partial derivatives have different signs on both sides.

as illustrated in Figure 5.5 where the x partial derivatives on both sides have different signs. In this case, one needs to decide which one-sided derivative should be taken through an appropriate upwinding method. Therefore, an upwinding method chooses

$$\frac{\partial T_{ij}}{\partial x} \approx \begin{cases} \text{diff}_x^L(T_{ij}) \\ \text{diff}_x^R(T_{ij}) \\ 0 \end{cases}$$

depending on the values of $\text{diff}_x^L(T_{ij})$ and $\text{diff}_x^R(T_{ij})$.

In WRF-SFIRE, the computation of $||\nabla T||_2^2$ is implemented using different upwinding methods. However, the ENO upwinding method is most frequently used. It is formulated as

$$\frac{\partial T_{ij}}{\partial x} \approx \begin{cases} \text{diff}_x^R, \text{ if } [\text{diff}_x^L < 0 \text{ and } \text{diff}_x^R \leq 0] \text{ or } [\text{diff}_x^L \leq 0 \text{ and } \text{diff}_x^R < 0] \\ \text{diff}_x^L, \text{ if } [\text{diff}_x^L > 0 \text{ and } \text{diff}_x^R \geq 0] \text{ or } [\text{diff}_x^L \geq 0 \text{ and } \text{diff}_x^R > 0] \\ \text{diff}_x^L, \text{ if } \text{diff}_x^L > 0 \text{ and } \text{diff}_x^R \leq 0 \text{ and } |\text{diff}_x^L| \geq |\text{diff}_x^R| \\ \text{diff}_x^R, \text{ if } \text{diff}_x^L \geq 0 \text{ and } \text{diff}_x^R < 0 \text{ and } |\text{diff}_x^L| < |\text{diff}_x^R| \\ 0, \text{ otherwise} \end{cases} \tag{5.16}$$

where $\text{diff}_x^L = \text{diff}_x^L(T_{ij})$ and $\text{diff}_x^R = \text{diff}_x^R(T_{ij})$. The ENO upwinding method can be defined similarly for the y partial derivative.

The derivative from equation (5.15) is expended by the chain rule, assuming that the formula for the derivative of a squared function is also valid for one-sided partial derivatives. Thus,

$$\frac{\partial(||\nabla T_{ij}||_2^2)}{\partial T_{kl}} = \frac{\partial\left(\left(\frac{\partial T_{ij}}{\partial x}\right)^2 + \left(\frac{\partial T_{ij}}{\partial y}\right)^2\right)}{\partial T_{kl}} = 2\left[\frac{\partial T_{ij}}{\partial x}\frac{\partial(\frac{\partial T_{ij}}{\partial x})}{\partial T_{kl}} + \frac{\partial T_{ij}}{\partial y}\frac{\partial(\frac{\partial T_{ij}}{\partial y})}{\partial T_{kl}}\right],$$

where $\frac{\partial T_{ij}}{\partial x}$ and $\frac{\partial T_{ij}}{\partial y}$ are computed using the ENO upwinding method of equation (5.16).

At this point, two questions arise. How can

$$\frac{\partial(\frac{\partial T_{ij}}{\partial x})}{\partial T_{kl}} \text{ and } \frac{\partial(\frac{\partial T_{ij}}{\partial y})}{\partial T_{kl}}, \tag{5.17}$$

be defined? Are one-sided partial derivatives from an upwinding method differentiable?

First, one can define these derivatives using perturbations of the neighbors, and the ENO upwinding method seen before. However, the one-sided partial derivatives from an upwinding method are not differentiable because sometimes the derivatives doing perturbations of a node at the negative and positive sides are not the same. Therefore, an approximation is proposed which defines for all the possible cases of three consecutive nodes $T_{i-1,j}$, T_{ij} and $T_{i+1,j}$, the derivatives from the negative and positive side. Then, the solution is defined as the average of both sides. The same thing is done for the y direction taking three consecutive nodes $T_{i,j-1}$, T_{ij} and $T_{i,j+1}$. In order to illustrate the procedure to compute all the possible cases, the example of the three consecutive nodes in the x direction in Figure 5.6a is explained. In that case, the ENO upwinding method of equation (5.16) decides that

$$\frac{\partial T_{ij}}{\partial x} \approx \text{diff}_x^L(T_{ij}) = \frac{T_{ij} - T_{i-1,j}}{\Delta x},$$

because

$$\text{diff}_x^L(T_{ij}) > 0,\ \text{diff}_x^R(T_{ij}) < 0 \ \text{ and } \ |\text{diff}_x^L(T_{ij})| = |\text{diff}_x^R(T_{ij})|.$$

At this point, in order to compute the derivatives of equation (5.17), one observes that the only derivatives of $\frac{\partial T_{ij}}{\partial x}$ different than 0 are those with respect to $T_{i-1,j}$, T_{ij} and $T_{i+1,j}$. Thus,

$$\frac{\partial(\frac{\partial T_{ij}}{\partial x})}{\partial T_{i-1,j}},\ \frac{\partial(\frac{\partial T_{ij}}{\partial x})}{\partial T_{ij}} \ \text{ and } \ \frac{\partial(\frac{\partial T_{ij}}{\partial x})}{\partial T_{i+1,j}},$$

can be nonzero.

This fact is because the derivative of T_{ij} with respect to x only can depend on its neighbors and itself because the upwinding method depends only on neighbors and the point itself. Therefore, one can start computing the derivative of $\frac{\partial T_{ij}}{\partial x}$ with respect to T_{ij} by perturbing this node T_{ij} (Figure 5.6b). When there is a perturbation to the positive and negative side of the node T_{ij}, one can observe in Figure 5.6b that the one-sided derivatives do not change. Therefore, the ENO upwinding method decides to use the same left one-sided derivative as in the initial problem. Therefore

$$\left(\frac{\partial(\frac{\partial T_{ij}}{\partial x})}{\partial T_{ij}}\right)_+ = \left(\frac{\partial(\frac{\partial T_{ij}}{\partial x})}{\partial T_{ij}}\right)_- \approx \frac{\partial(\text{diff}_x^L(T_{ij}))}{\partial T_{ij}} = \\ = \frac{\partial(\frac{T_{ij} - T_{i-1,j}}{\Delta x})}{\partial T_{ij}} = \frac{1}{\Delta x},$$

where $(\cdot)_+$ and $(\cdot)_-$ represents the derivatives from the positive and negative side, respectively. Therefore, in that particular case, both partial derivative sides are equal, implying that the partial derivative is differentiable with respect to T_{ij}. Thus, the final result is

$$\frac{\partial(\frac{\partial T_{ij}}{\partial x})}{\partial T_{ij}} = \left(\frac{\partial(\frac{\partial T_{ij}}{\partial x})}{\partial T_{ij}}\right)_+ = \left(\frac{\partial(\frac{\partial T_{ij}}{\partial x})}{\partial T_{ij}}\right)_- \approx \frac{1}{\Delta x}.$$

The next case considered is the derivative of $\frac{\partial T_{ij}}{\partial x}$ with respect to $T_{i-1,j}$. So, the node $T_{i-1,j}$ is perturbed to the positive and negative direction as before (Figure 5.7).

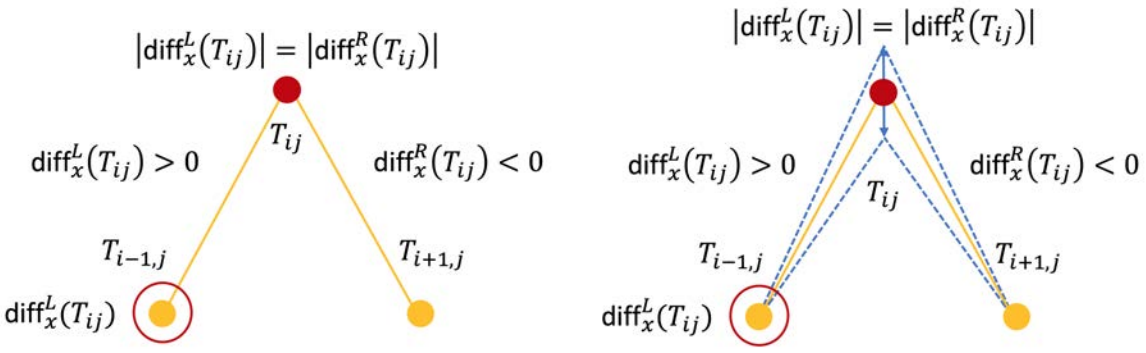


Figure 5.6: (a) Example of three consecutive nodes in the x direction of a fire arrival time function $T(x, y)$ with the y direction held constant. The x partial derivatives have different signs in both sides. The ENO upwinding method decides to take the left one-sided derivative for the x partial derivative of T_{ij}. (b) The same example as in Figure 5.6a, but perturbing the central node T_{ij} in the positive and negative direction. The ENO upwinding method decides to take the left one-sided derivative for the x partial derivative of T_{ij} after the perturbations.

In this case, the one-sided derivatives are different depending on the direction of the perturbation. Therefore, the case is split in two different computations. As one can see in Figure 5.7a, when $T_{i-1,j}$ is perturbed to the positive direction, the derivative of T_{ij} with respect to x according to the ENO upwinding method changes to $\text{diff}_x^R(T_{ij})$ from the original problem. Thus,

$$\left(\frac{\partial(\frac{\partial T_{ij}}{\partial x})}{\partial T_{i-1,j}}\right)_+ \approx \frac{\partial(\text{diff}_x^R(T_{ij}))}{\partial T_{i-1,j}} = \frac{\partial(\frac{T_{i+1,j}-T_{ij}}{\Delta x})}{\partial T_{i-1,j}} = 0.$$

On the other hand, Figure 5.7b shows that after perturbing $T_{i-1,j}$ to the negative direction, the derivative of T_{ij} with respect to x according to the ENO upwinding method is $\text{diff}_x^L(T_{ij})$. So,

$$\left(\frac{\partial(\frac{\partial T_{ij}}{\partial x})}{\partial T_{i-1,j}}\right)_- \approx \frac{\partial(\text{diff}_x^L(T_{ij}))}{\partial T_{i-1,j}} = \frac{\partial(\frac{T_{ij}-T_{i-1,j}}{\Delta x})}{\partial T_{i-1,j}} = -\frac{1}{\Delta x}.$$

Consequently, the result for that particular case is

$$\frac{\partial(\frac{\partial T_{ij}}{\partial x})}{\partial T_{i-1,j}} \approx \frac{1}{2}\left[\left(\frac{\partial(\frac{\partial T_{ij}}{\partial x})}{\partial T_{i-1,j}}\right)_+ + \left(\frac{\partial(\frac{\partial T_{ij}}{\partial x})}{\partial T_{i-1,j}}\right)_-\right] \approx \frac{0 - \frac{1}{\Delta x}}{2} = \frac{-1}{2\Delta x}.$$

Finally, one can analyze the derivative of $\frac{\partial T_{ij}}{\partial x}$ with respect to $T_{i+1,j}$ (Figure 5.8). In this case, both one-sided derivatives are different. So, in the case of the positive direction, one can observe in Figure 5.8a that

$$\left(\frac{\partial(\frac{\partial T_{ij}}{\partial x})}{\partial T_{i+1,j}}\right)_+ \approx \frac{\partial(\text{diff}_x^L(T_{ij}))}{\partial T_{i+1,j}} = \frac{\partial(\frac{T_{ij}-T_{i-1,j}}{\Delta x})}{\partial T_{i+1,j}} = 0,$$

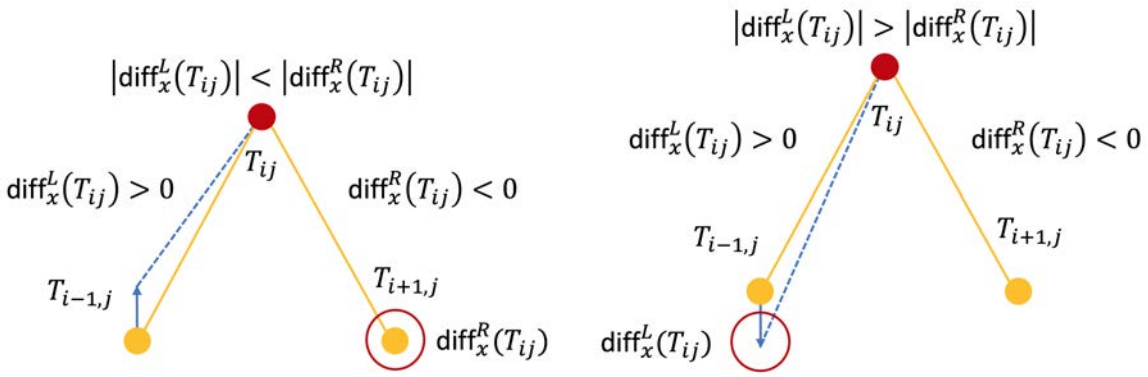


Figure 5.7: (a) The same example as in Figure 5.6a, but perturbing the left node $T_{i-1,j}$ in the positive direction. The ENO upwinding method decides to take the right one-sided derivative for the x partial derivative of T_{ij} after the perturbation. (b) The same example as in Figure 5.6a, but perturbing the left node $T_{i-1,j}$ in the negative direction. The ENO upwinding method decides to take the left one-sided derivative for the x partial derivative of T_{ij} after the perturbation.

and in Figure 5.8b, one can conclude that

$$\left(\frac{\partial(\frac{\partial T_{ij}}{\partial x})}{\partial T_{i+1,j}}\right)_- \approx \frac{\partial(\text{diff}_x^R(u_{ij}))}{\partial T_{i+1,j}} = \frac{\partial(\frac{T_{i+1,j}-T_{ij}}{\Delta x})}{\partial T_{i+1,j}} = \frac{1}{\Delta x}.$$

Consequently, the final result is

$$\frac{\partial(\frac{\partial T_{ij}}{\partial x})}{\partial T_{i+1,j}} \approx \frac{1}{2}\left[\left(\frac{\partial(\frac{\partial T_{ij}}{\partial x})}{\partial T_{i+1,j}}\right)_+ + \left(\frac{\partial(\frac{\partial T_{ij}}{\partial x})}{\partial T_{i+1,j}}\right)_-\right] \approx \frac{0+\frac{1}{\Delta x}}{2} = \frac{1}{2\Delta x}.$$

Then, computing the derivatives with respect to y similarly and analyzing all the cases, one can compute the partial derivatives in equation (5.17). However, in equation (5.15) there is still a gap. How can $\frac{\partial R_{ij}^2}{\partial T_{kl}}$ be computed? What are the dependencies of R_{ij}?

Assuming that formula for the derivative of a squared function is valid, using the chain rule gives

$$\frac{\partial(R_{ij}^2)}{\partial T_{kl}} = 2R_{ij}\frac{\partial R_{ij}}{\partial T_{kl}}.$$

However, it is necessary to know the dependencies of R_{ij} in order to compute the derivative $\frac{\partial R_{ij}}{\partial T_{kl}}$. As was shown in Section 5.2, the rate of spread R depends on many factors: the spatial position $\vec{x}$, the fire arrival time $T(\vec{x})$, and the gradient of the fire arrival time $\nabla T(\vec{x})$. Therefore, the previous derivative is complex. Consequently, it is proposed to assume that the rate of spread R is constant during the minimization, so it is independent of the fire arrival time T. Therefore, the derivative

$$\frac{\partial R_{ij}}{\partial T_{kl}} = 0.$$

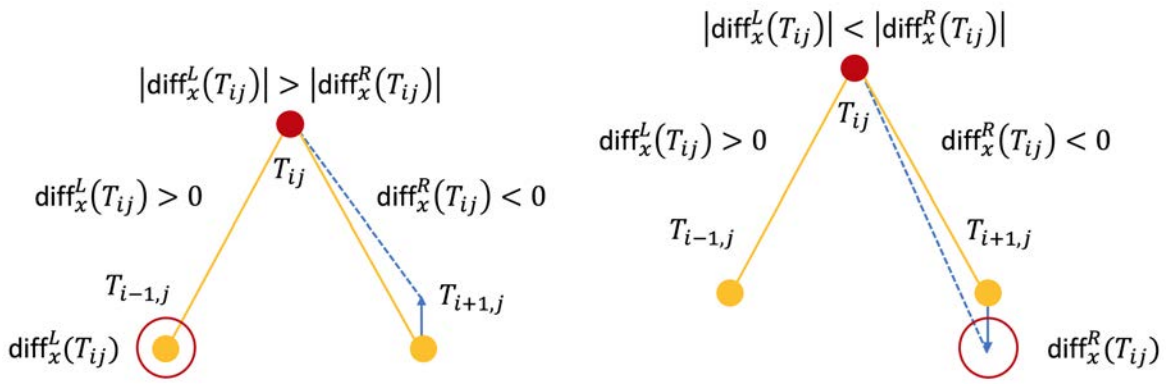


Figure 5.8: (a) The same example as in Figure 5.6a, but perturbing the right node $T_{i+1,j}$ in the positive direction. The ENO upwinding method decides to take the left one-sided derivative for the x partial derivative of T_{ij} after the perturbation. (b) The same example as in Figure 5.6a, but perturbing the right node $T_{i+1,j}$ in the negative direction. The ENO upwinding method decides to take the right one-sided derivative for the x partial derivative of T_{ij} after the perturbation.

However, the computation of this derivative could be line of a future research. Therefore, an approximation of the gradient of the objective function $J(T)$ is proposed. An approximation of the derivative of an upwinding method is also proposed because one can generalize this methodology for any upwinding method.

Once there is a method to approximate the gradient of the objective function, one can use this approximation in order to apply the projected gradient descent method also known as the projected steepest descent method. This method consists in finding a better solution for the nonlinear optimization method of equation (5.7) in an iterative fashion. From the initial approximation of Section 5.4.2, in each iteration, a new T^{k+1} is computed from the previous T^k using the equation

$$T^{k+1} = T^k - \alpha_k \nabla J(T^k),$$

where $\nabla J(T^k)$ is the approximation of the gradient of the objective function in equation (5.14) which is the steepest direction, i.e. the direction giving the best improvement for the next iteration and α_k is the step size, i.e. how far the new iteration is going in the previous direction. This step size is computed using the exact line search

$$\alpha_k = \min_{\alpha \in \mathbb{R}} \; J(T^k - \alpha \nabla J(T^k)).$$

Therefore, the idea is to find, in each iteration, a direction and a step size which moves to a better fire arrival time minimizing the objective function $J(T)$. This method converges, but it is very slow.

In order to improve the rate of convergence, the gradient is preconditioned using the matrix $A = -\triangle$ (discrete Laplace operator). In that manner, the convergence is faster and one can solve the optimization problem using

$$T^{k+1} = T^k - \alpha_k A^{-1} \nabla J(T^k),$$

where α_k was found using the exact line search

$$\alpha_k = \min_{\alpha \in \mathbb{R}} \; J(T^k - \alpha A^{-1} \nabla J(T^k)). \tag{5.18}$$

This method works for cases with a constant rate of spread R because a constant rate of spread was assumed in order to compute the gradient of the objective function. However, as was previously shown, the rate of spread depends on the fire arrival time. For this reason, the projected multigrid descent method is explored since it is not necessary to assume the rate of spread constant because the computation of the gradient of the objective function is not requiered. Therefore, one can use the method without taking into account any derivative for the rate of spread.

5.4.4 Projected multigrid descent method

First of all, a first approximation satisfying the constraints is found using the quadratic minimization problem of Section 5.4.2. After that, the residuals of the eikonal equation are minimized using the projected multigrid descent method. This method consists in combining line searches in the direction of changes to the value of the fire arrival time T at a single point, and linear combinations of point values as in [46].

Therefore, a bilinear coarse grid function φ is defined using different coarse mesh sizes. The bilinear coarse grid functions are defined as pyramids centered at all points (x, y) in the domain, using different mesh sizes s and with supports

$$[x - s, x + s] \times [y - s, y + s].$$

The formulation of the bilinear coarse grid functions in the previous square is

$$\varphi_s([x - s, x + s] \times [y - s, y + s]) = \max\left\{0, \left(1 - \frac{|\hat{x}|}{s}\right) \cdot \left(1 - \frac{|\hat{y}|}{s}\right)\right\},$$

for all $(\hat{x}, \hat{y}) \in [-s, s] \times [-s, s]$ and 0 in the rest of the domain. In Figure 5.9, there is an example of a bilinear coarse grid function in a 100×100 domain, centered at the point $(50, 50)$ with mesh size $s = 8$.

Then, φ_s is projected onto the $H\varphi_s = 0$ subspace, so that the constraints remain satisfied throughout the iterations. The projection is $P = I - H^\top (HH^\top)^{-1} H$ and is formulated as

$$\tilde{\varphi}_s = \varphi_s - H^\top (HH^\top)^{-1} H \varphi_s.$$

Figure 5.10a shows an example of a bilinear coarse grid function with mesh step size 16 and Figure 5.10b shows the same bilinear coarse grid function projected onto the $H\varphi_s = 0$ subspace showing also the constraints (perimeter shape points). One can observe that in all the points where the constraints lie, the bilinear coarse grid function is defined as 0 in order to keep the constraints invariant through the iterations.

These projected bilinear coarse grid functions are the different directions in which to apply the line search in all the following cases: changing the mesh step sizes s, centering at all the points which are required to interpolate, and applying the direction to the positive and negative sides $d_s = \pm\tilde{\varphi}_s$. The required points to apply the minimization

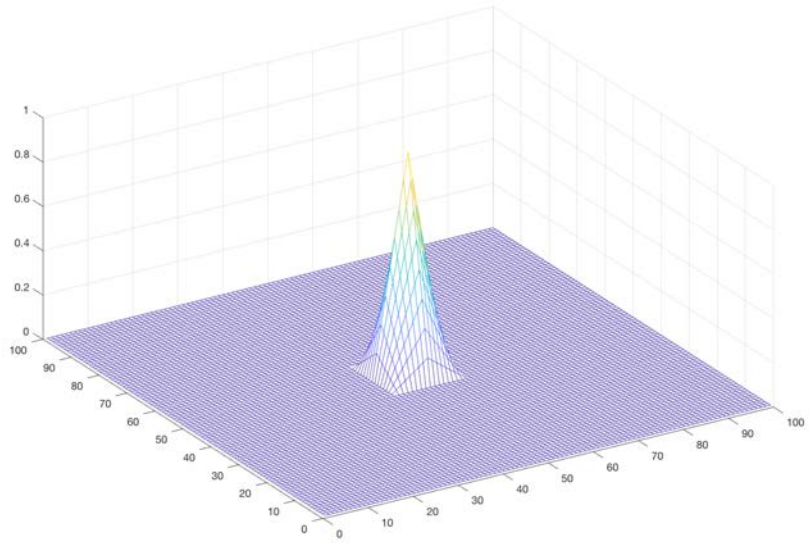
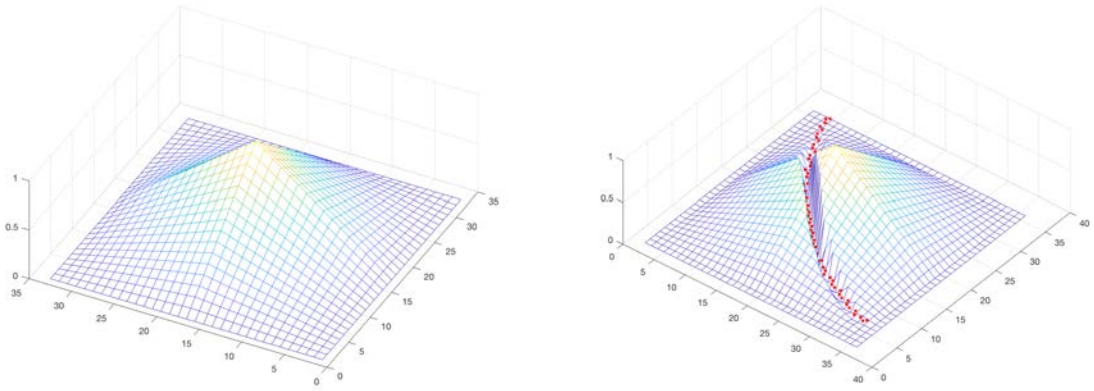


Figure 5.9: Example of a bilinear coarse grid function φ in a square 100×100 domain, centered at $(x, y) = (50, 50)$ with mesh size $s = 8$



(a) (b)

Figure 5.10: (a) Example of a bilinear coarse grid function φ with mesh size $s = 16$. (b) The same φ function projected to $H\varphi = 0$. The red dots represent the shape points of the perimeter.

to are the points between the two perimeters or between the ignition point and the first perimeter. In order to choose the points, for each perimeter Γ, the optimization problem

$$P_\Gamma(T) = \frac{1}{2}\int_\Omega \left\|(-\triangle)^\alpha T\right\|^2 dxdy \to \min_T$$
$$\text{subject to } T = T_\Gamma \text{ at } \Gamma,$$
$$T = T_i \text{ at } \vec{m} \text{ and}$$
$$T = T_{\max} \text{ at } \partial\Omega$$

is solved, where T_Γ is the fire arrival time at the perimeter Γ, T_i is the ignition time, $\vec{m}$ is an interior point from the intersection of the interior of all the perimeters as far as possible from the first perimeter, and $T_{\max}$ is the maximum value of the fire arrival time at the boundaries. The value α used in all the experiments is $\alpha = 1.4 > 1$.

The solution of this squared minimization T is found using the same strategy as in the squared minimization in Section 5.4.2. This new minimization gives an accurate approximation of a mask for each perimeter Γ. The masks are defined as

$$M_\Gamma = \begin{cases} 1, & T < T_\Gamma \\ 0, & \text{otherwise.} \end{cases}$$

Then, in the multigrid method, the bilinear coarse grid functions are centered only at the required points. In the case of interpolating between the ignition point and the first perimeter, the required points are those with value one in the first mask M_{Γ_1}. In the other cases interpolating between two consecutive perimeters Γ_k and Γ_{k+1}, the required points are those with value one in $M^c_{\Gamma_k} \cap M_{\Gamma_{k+1}}$. The expression $M^c_{\Gamma_k}$, means the complement of the mask, i.e. where the mask M_{Γ_k} is 0.

The idea of the multigrid method is to solve the minimization problem in the equation (5.7) iteratively. From the initial approximation of Section 5.4.2, in each iteration, a new T^{k+1} is computed from the previous T^k using the equation

$$T^{k+1} = T^k + \alpha_k d_s, \tag{5.19}$$

where α_k is found using the exact line search

$$\alpha_k = \min_\alpha \; J(T^k + \alpha d_s).$$

These equations mean that different directions d_s are tried using the best step sizes for each direction and if a direction and step size improves the fire arrival time, this solution is changed for the next step. Therefore, it is a trial and error technique that in each iteration tries to change the fire arrival time at a single point or in a linear combination of them.

The multigrid method is typically solved using at the beginning the maximum mesh step size permitted in the domain and then the mesh step size decreases by a factor of 2 at each step size iteration. Each step size iteration is repeated a different number of

times. These times will be called sweeps going forward. The smallest step size (step size 1) is done one time and the number of sweeps increases linearly as the step size increases at each iteration. All the step size iterations are repeated a fixed number of times or general cycles. In all the experiments, the general number of cycles is 4. Each cycle is ordered from the coarser step size iterations to the finest.

Finally, a relaxed projection of T^{k+1} onto the constraints subspace, $HT^{k+1} = g$, is done in order to smooth the iterated solution.

Therefore, the same projection $P = I - H^\top (HH^\top)^{-1} H$ is used as

$$\tilde{T}^{k+1} = T^{k+1} - \rho H^\top (HH^\top)^{-1} (HT^{k+1} - g),$$

where ρ is a fixed relaxation parameter. In all the experiments, $\rho = 0.1$. For the initial approximation, the projection is made completely by setting $\rho = 1$.

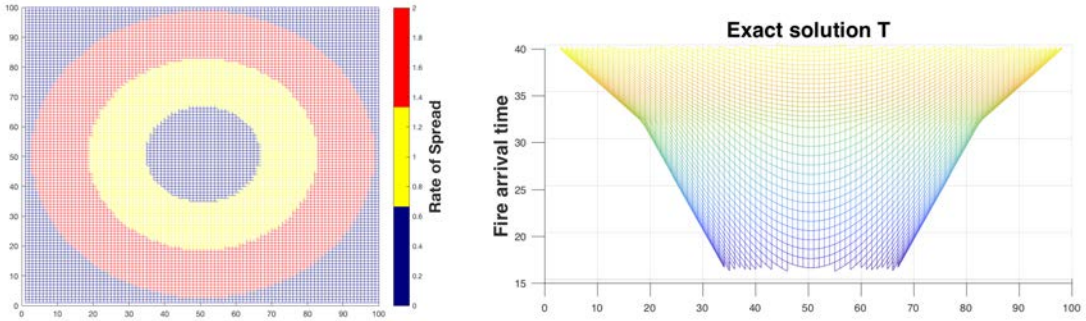
5.4.5 Dynamic rate of spread

This section proposes a solution for the numerical optimization of the residuals using the dynamic rate of spread idea from Section 5.3.3. The first important observation is that the projected preconditioned gradient descent method of Section 5.4.3 can not be used for this problem. The reason is that the projected preconditioned gradient descent method assumes constant rate of spread. Therefore, the dynamic rate of spread is going to be used inside the projected descent multigrid method of Section 5.4.4.

After each cycle of the projected multigrid descent method, the rate of spread is recomputed from the new fire arrival time T. In order to compute the new rate of spread $R(T)$, the x and y components of the wind in the fire mesh (UF and VF respectively) and the fuel moisture (FMC_G) are collected at all the time steps $(t_0, t_1, \ldots, t_k)$ of the first simulation of WRF-SFIRE run in the first step of the work-flow detailed in Section 5.1. These wind and fuel moisture variables are defined at all the simulation times τ and points $\vec{x}$ in the domain as $\mathrm{UF}_\tau(\vec{x})$, $\mathrm{VF}_\tau(\vec{x})$, and $\mathrm{FMC\_G}_\tau(\vec{x})$ respectively. Then, the rate of spread is computed from the fire arrival time T and the winds and fuel moisture from the simulation. Therefore, at each point $\vec{x}$ in the domain, the wind components $\mathrm{UF}_\tau(\vec{x})$ and $\mathrm{VF}_\tau(\vec{x})$, and the fuel moisture $\mathrm{FMC\_G}_\tau(\vec{x})$ are interpolated from all the simulation times $\tau = t_0, t_1, \ldots, t_k$ to the fire arrival time at the same point $\tau = T(\vec{x})$. So, for all the points $\vec{x}$ in the domain, the values $\mathrm{UF}_{T(\vec{x})}(\vec{x})$, $\mathrm{VF}_{T(\vec{x})}(\vec{x})$, and $\mathrm{FMC\_G}_{T(\vec{x})}(\vec{x})$ are obtained. Using these values of the wind and the fuel moisture, the fire spread model is computed using equation (2.1) in the normal direction

$$\frac{\nabla T(\vec{x})}{||\nabla T(\vec{x})||},$$

where $T(\vec{x})$ is the new fire arrival time obtained after each cycle of the projected multigrid descent method at point $\vec{x}$ in the domain.



(a) (b)

Figure 5.11: (a) Rate of spread in the concentric circle perimeters problem. (b) Exact solution T for the concentric circles problem.

5.5 Experimental results

5.5.1 Ideal case

The optimization problem defined in Section 5.3.2 is tested using both numerical solutions proposed in Sections 5.4.3 and 5.4.4 on an idealized case. This ideal case is defined using concentric circles as perimeters on a mesh with 100×100 nodes. The fire spreads equally in all directions from the center of the mesh. The propagation is set at different rates of spread in different sections of the domain. There is a jump in the rate of spread from $R_1 = 1$ to $R_2 = 2$ dividing the space between perimeters Γ_1 and Γ_2 in two different circular sectors of the same size (Figure 5.11a). Then, the fire arrival time at Γ_1 is defined and the fire arrival time at Γ_2 is computed from the given rates of spread. In this case there exists an exact solution shown in Figure 5.11b.

As a first step, the linear system of constraints is constructed from the barycentric method defined in Section 5.4.1. Figure 5.12 shows the two circular concentric perimeters defined for the linear system of constraints.

Then, an initial approximation of the fire arrival time is found by solving the quadratic minimization problem described in Section 5.4.2 with $\alpha = 1.4$. Figure 5.13 shows the initial approximation of the fire arrival time imposed by the ignition point and the two concentric circles in our particular case, using different values of α from 1 to 1.4. One can see how the unrealistic sharp funnel at the ignition point for $\alpha = 1$ disappears with increasing values of α.

From the first approximation using $\alpha = 1.4$, T^0, which satisfies the constraints (Figure 5.14a), the projected preconditioned gradient descent method in Section 5.4.3 can be used to solve the optimization problem and improve the initial objective function value of $J(T^0) = 8.4779$. In Figure 5.14b, one can observe an example of how the line searches work in order to find the best step size for each iteration. One can observe that in the final iteration of the line search, the improvement in the objective function value is negligible. After finding the best step size at each iteration, the fire arrival time can be changed if the objective function improves. In Figure 5.14c, one can observe how the objective function is improved at each iteration and also see that at the beginning

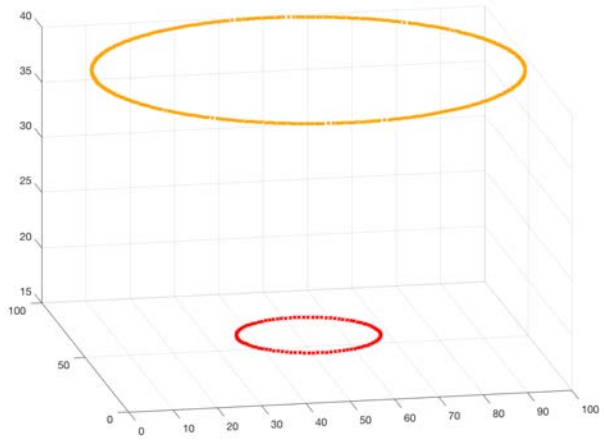


Figure 5.12: Graphic of the linear system of constraints generated using the barycentric method for the ideal case of the circular concentric perimeters.

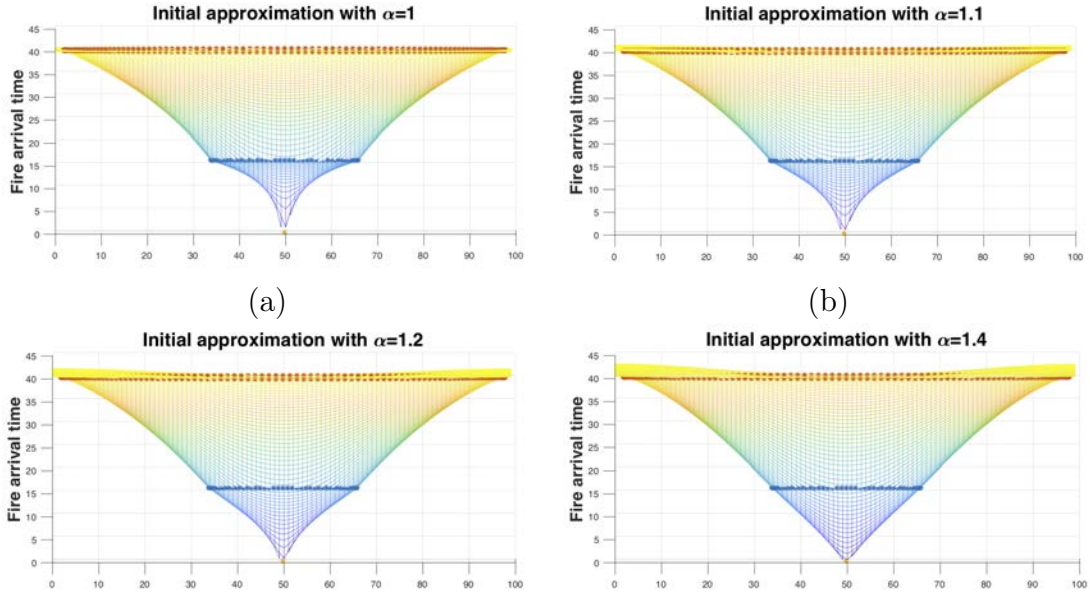


Figure 5.13: Initial approximation of the fire arrival time T in the two concentric circles perimeter case using different values of α.

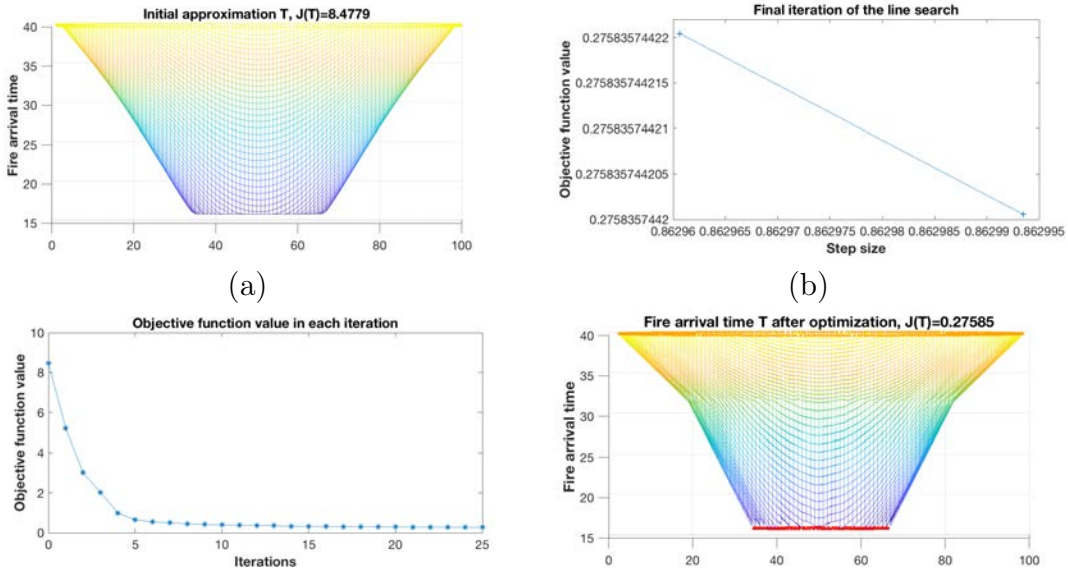


Figure 5.14: (a) Initial approximation from the first perimeter at $T_1 = 16$ to the second perimeter at $T_2 = 40$ obtained with $\alpha = 1.4$. (b) Final iteration of the line search looking for the best step size. (c) Values of the objective function after each iteration of the projected preconditioned gradient descent method. (d) Result of the fire arrival time interpolation after 100 iterations of the projected preconditioned gradient descent method.

the objective function values improve significantly with each iteration and then start becoming almost constant values. After 25 iterations, the resulting fire arrival time can be seen in Figure 5.14d with objective function value of $J(T^{25}) = 0.27585$. One can observe that the shape of the resulting fire arrival time is very similar to the exact solution in Figure 5.11b. Finally, the norm of the gradient of the objective function of the initial approximation is $||\nabla J(T^0)|| = 0.1593$ and using the projected preconditioned gradient descent method, the norm of the gradient is reduced to $||\nabla J(T^{25})|| = 0.0961$ after 25 iterations.

As an alternative method, the projected multigrid descent method proposed in Section 5.4.4 is run. The coarsening is done by the ratio of 2. The number of sweeps is linearly decreasing with the level. On the coarsest level, the mesh step is 16 and the sweep is done 5 times. The mesh step on the second level is 8 and the sweep is repeated 4 times, until resolution 1 on the original, finest grid, and the sweep repeats once. The same first approximation is used (Figure 5.15a) and an example of a bilinear coarse grid function with step size 16 is shown in Figure 5.15b. Figure 5.15c shows the decrease in the cost function with the number of line searches on any level. One can observe that the cost function decreases more in the first cycle, and then the curve becomes flatter. The final result after 4 cycles of 5 different resolutions (from 16 to 1 decreasing by powers of two) is shown in Figure 5.15d, which again is close to the exact solution in Figure 5.11b. The projected multigrid descent method with dynamic rate of spread of Section 5.4.5 can not be solved in this case because a constant rate of spread is defined.

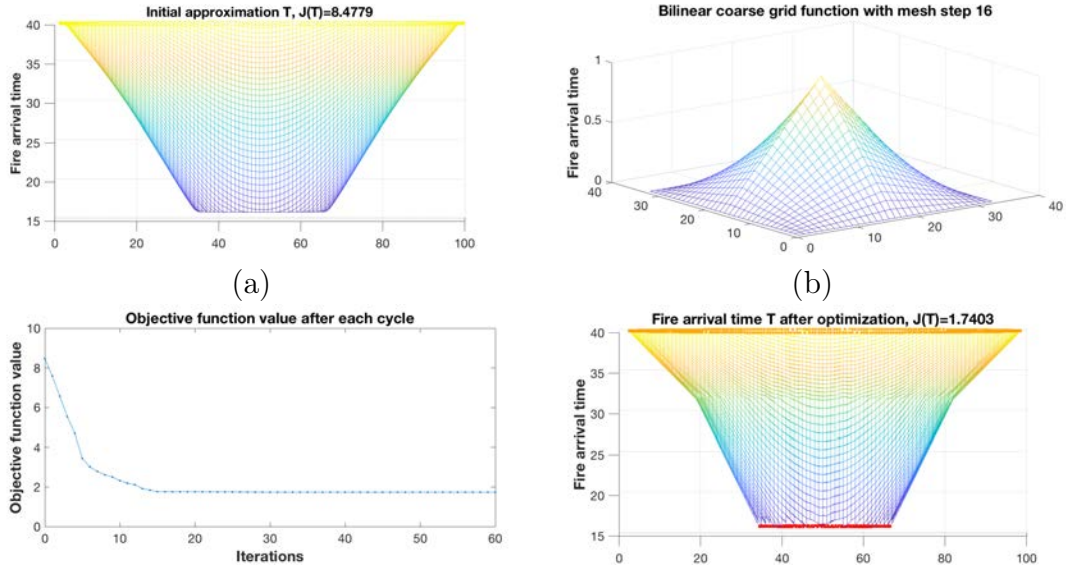


Figure 5.15: (a) Initial approximation from the first perimeter at $T_1 = 16$ to the second perimeter at $T_2 = 40$ obtained with $\alpha = 1.4$. (b) Example of a bilinear coarse grid function at mesh step 16. (c) Values of the objective function after each step size iteration of the projected multigrid descent method. (d) Result of the fire arrival time interpolation after 4 cycles of the projected multigrid descent method.

Finally, a comparison between the previous methods is performed. Therefore, the initial approximation, the projected preconditioned gradient descent method, and the projected multigrid descent method are compared.

As there is an exact solution, one can use the absolute percentage error of the fire arrival time

$$APE_{ij} = \frac{|S_{ij} - A_{ij}|}{|S_{ij}|} \cdot 100, \tag{5.20}$$

where S_{ij} is the (i, j) component of the exact solution of the fire arrival time and A_{ij} is the (i, j) component of the fire arrival time approximation from the three different optimization methods.

Then, the mean absolute percentage error is formulated as

$$MAPE = \frac{1}{M \cdot N} \sum_{i=1}^{M} \sum_{j=1}^{N} APE_{ij}. \tag{5.21}$$

Figure 5.16a, shows the absolute percentage error and how the error is concentrated in the rate of spread jump because the first approximation does not consider the effect of the rate of spread. The mean absolute percentage error of the first approximation is 2.11%. Using the projected preconditioned gradient descent method, the error is considerably reduced as shown in Figure 5.16b. In this case, the error is accumulated next to the first perimeter. This could be caused by the projection at the end of each iteration. However, the maximum error is less than 4% and the mean absolute

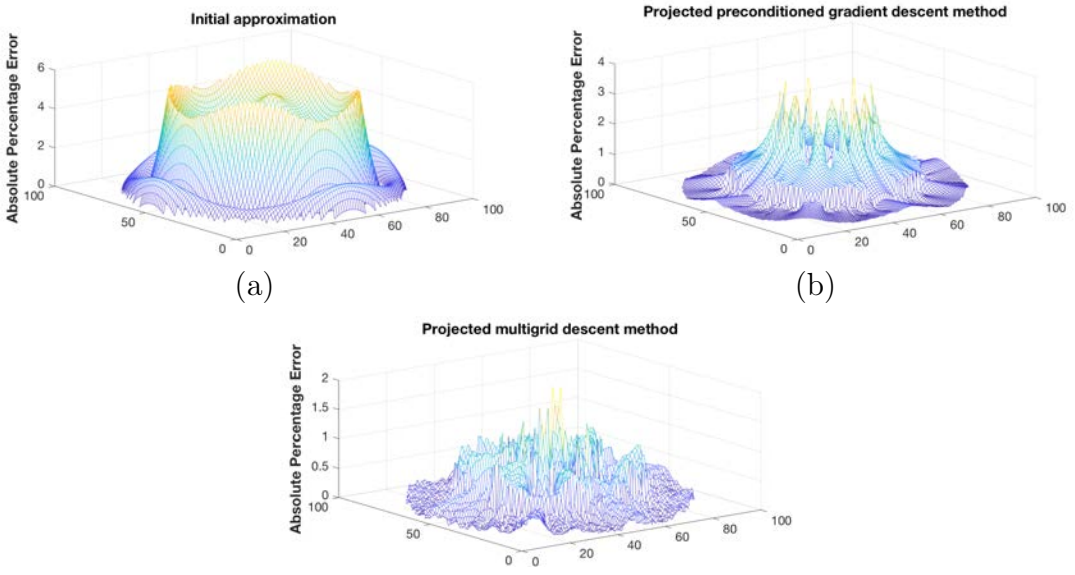


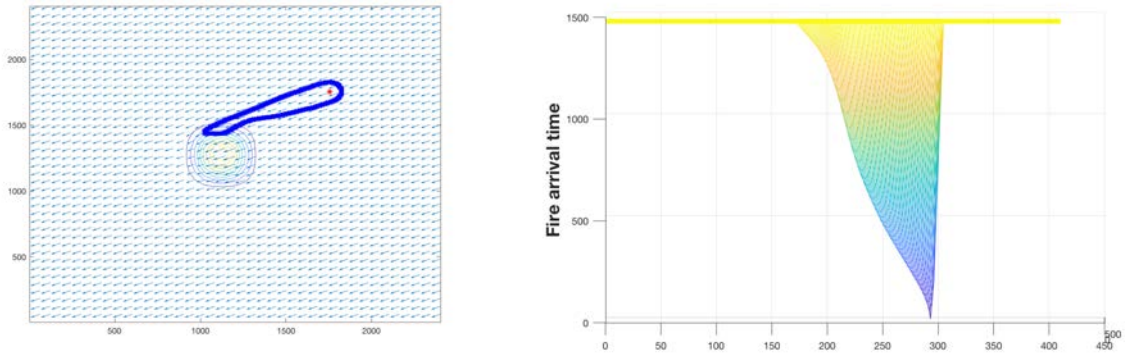
Figure 5.16: (a) Absolute percentage error of the initial approximation compared to the exact solution for the concentric circles experiment. (b) Absolute percentage error of the projected preconditioned gradient descent method result compared to the exact solution for the concentric circles experiment. (c) Absolute percentage error of the projected multigrid descent method compared to the exact solution for the concentric circles experiment.

percentage error is 0.52%, less than 1%. Figure 5.16c shows that using the projected multigrid descent method, the error in general is smaller than using both previous methods. However, near the first perimeter, there are still huge errors of 2%. The mean absolute percentage error of 0.26% is smaller than both previous methods and is a significant improvement to the initial approximation.

5.5.2 WRF-SFIRE ideal case

The next step is to try to reproduce the fire arrival time generated by the coupled atmosphere-fire simulator, WRF-SFIRE. Therefore, an ideal WRF-SFIRE case is simulated and used as a test case for the previous optimization methods (Sections 5.4.3 and 5.4.4). The inputs for the optimization are: the latitude-longitude coordinates of the ignition point and the ignition time, simulated perimeter and its associated time, and the prognostic variables (components of the wind, components of the slope, and fuel moisture) used to compute the rate of spread in the domain at all the output time steps of the simulation.

In the previous ideal example in Section 5.5.1, there was not any variable of the simulation affecting the rate of spread computation because a specific constant rate of spread was enforced. In this example, the prognostic variables previously mentioned are going to be involved and they are going to drive the optimization. Therefore, it is important to have a reliable rate of spread computed from these prognostic variables



(a) (b)

Figure 5.17: (a) Configuration of the hill experiment. The red dot represents the ignition point, the blue shape represents the simulated perimeter, the blue arrows represents the initial winds, and the colored contour plot represents the elevation (a hill in the middle of the domain). (b) WRF-SFIRE simulated fire arrival time for the hill experiment.

in order to drive the optimization to the optimal fire arrival time. The fire arrival time depends on the rate of spread and in Section 5.2 one can observe that the fire arrival time affects the rate of spread as well. Therefore, for each change of the fire arrival time in the optimization process, it is important to also change the rate of spread. As a first approximation of the rate of spread, one can assume a constant rate of spread computed using the prognostic variables in the WRF-SFIRE simulation. After this approximation, a way to dynamically change the rate of spread in order to take into account the fire arrival time dependency on the rate of spread is proposed.

The ideal WRF-SFIRE experiment used is the hill case. The domain is a squared 410×410 grid, the initial atmosphere conditions are a strong wind from north-east, constant fuel category 3 of the 13 Rothermel's fuel categories [4] (which is grass), and a hill in the middle of the domain (Figure 5.17a). The interpolation is done from the ignition point at time T_i to the perimeter after some iterations of WRF-SFIRE at time T_1. The fire arrival time simulated using WRF-SFIRE can be observed in Figure 5.17b.

As in Section 5.5.1, the interpolation operator of the ignition point and the perimeter shape points, H, and the vector of fire arrival times, g, are constructed using the barycentric method explained in Section 5.4.1 (Figure 5.18). Then, an initial approximation is computed using the method in the Section 5.4.2 giving an objective function value of $J(T^0) = 8684.0704$ (Figure 5.19a and Figure 5.20a).

Then, the preconditioned projected gradient descent method is used as in the previous ideal perimeter circles experiment using as a starting point the previous first approximation. Figure 5.19b shows the objective function values at the final iteration of the line search as a function of the step size α (equation (5.18)). In Figure 5.19c, one can observe that the objective function improves significantly in the first iterations and then it is almost constant. Therefore, almost all the improvement is accomplished in the first iterations and then the method converges slowly. Figure 5.19d shows the fire

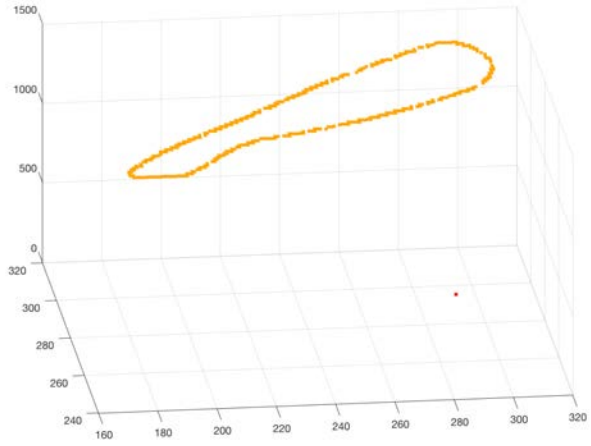


Figure 5.18: Graphic of the linear system of constraints generated using the barycentric method for the ideal WRF-SFIRE case of the hill.

arrival time after 1000 iterations with objective function value of $J(T^{1000}) = 487.7882$. Finally, the norm of the gradient of the objective function of the initial approximation is $||\nabla J(T^0)|| = 2.1758$ and using the projected preconditioned gradient descent method, the norm of the gradient is reduced to $||\nabla J(T^{1000})|| = 0.0279$ after 1000 iterations. Therefore, the value of the objective function, the norm of the gradient of the objective function, and the shape of the fire arrival time show an improvement from the initial approximation. However, the fire arrival time at the end of 1000 iterations is still far from the fire arrival time simulated with WRF-SFIRE at the same perimeter time (Figure 5.17b).

Next, the projected multigrid descent method of Section 5.4.4 is used. Figure 5.20b shows an example of one coarse bilinear function used for the multigrid minimization of mesh step size 16. In Figure 5.20c, one can observe that the solution improves faster in the first cycles and in the coarsest step size iterations in each of the general 4 cycles. The final solution in Figure 5.20d shows a substantial improvement over the initial approximation because of a similar shape of the fire arrival time compared with the fire arrival time simulated using WRF-SFIRE (Figure 5.17b).

Finally, the projected descent multigrid optimization method using dynamic rate of spread of Section 5.4.5 is used. In Figure 5.21b shows an example of one coarse bilinear function used for the multigrid minimization of mesh step size 16. In Figure 5.21c, one can observe that the solution improves faster in the first cycles and in the coarsest step size iterations in each of the general 4 cycles. The final solution in Figure 5.21d shows a substantial improvement over the initial approximation because of a similar shape of the fire arrival time compared with the fire arrival time simulated using WRF-SFIRE (Figure 5.17b). Using the dynamic rate of spread, one can construct in each step size iteration a more realistic rate of spread as well as a better fire arrival time. Therefore, the rate of spread computed using the first approximation of the fire arrival time T^0 shown in Figure 5.22a is improved same way as for the fire arrival time. Figure 5.22b shows the rate of spread at the last iteration of the projected multigrid descent method

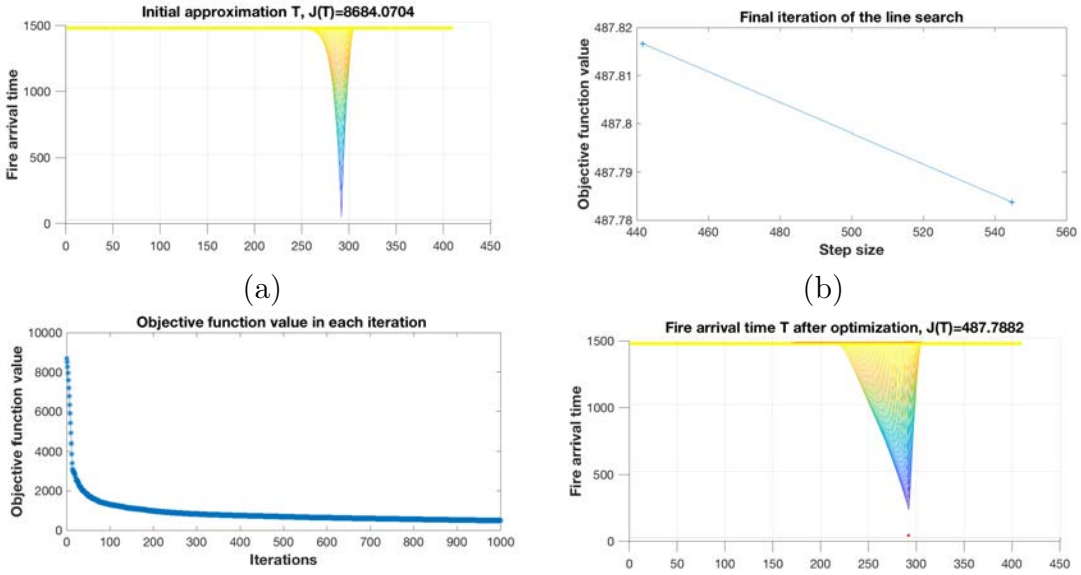


Figure 5.19: (a) Initial approximation from the ignition point to the first perimeter obtained with $\alpha = 1.4$. (b) Final iteration of the line search looking for the best step size. (c) Values of the objective function after each line search iteration of the preconditioned gradient descent method. (d) Fire arrival time interpolated after 1000 preconditioned gradient descent method iterations.

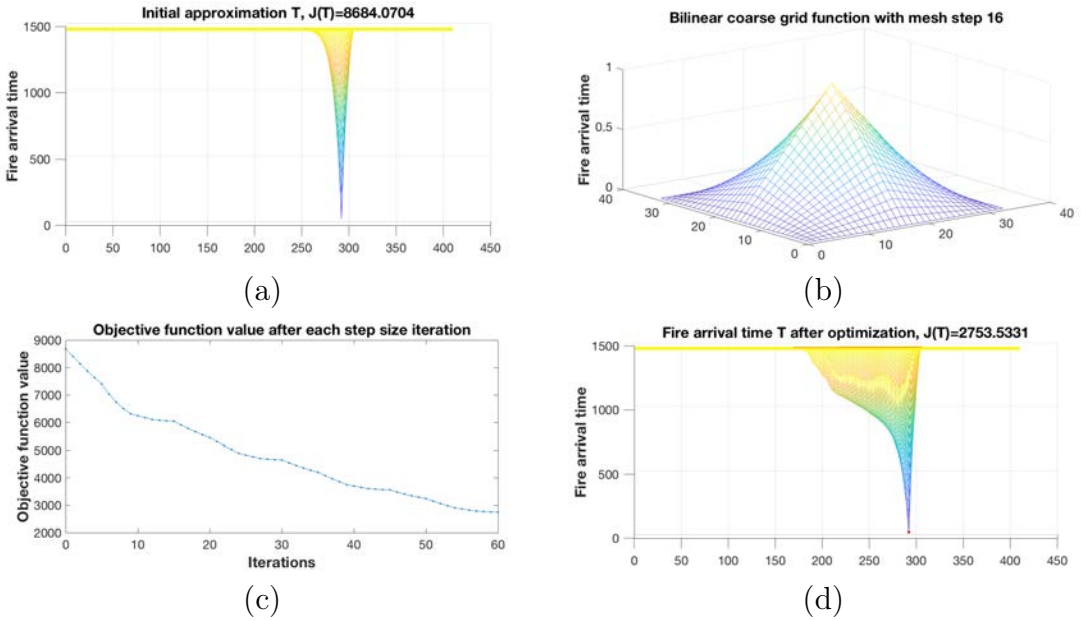


Figure 5.20: (a) Initial approximation from the ignition point to the first perimeter obtained with $\alpha = 1.4$. (b) Example of a bilinear coarse grid function at mesh step size 16. (c) Values of the objective function after each step size iteration of the projected multigrid descent method. (d) Result of the fire arrival time interpolation after 4 cycles of the projected multigrid descent method.

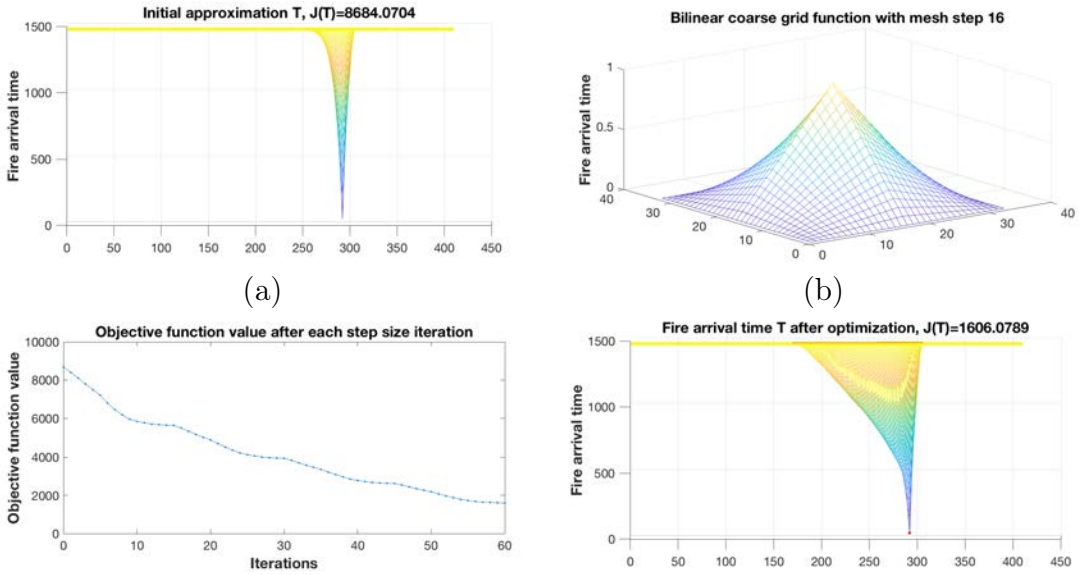
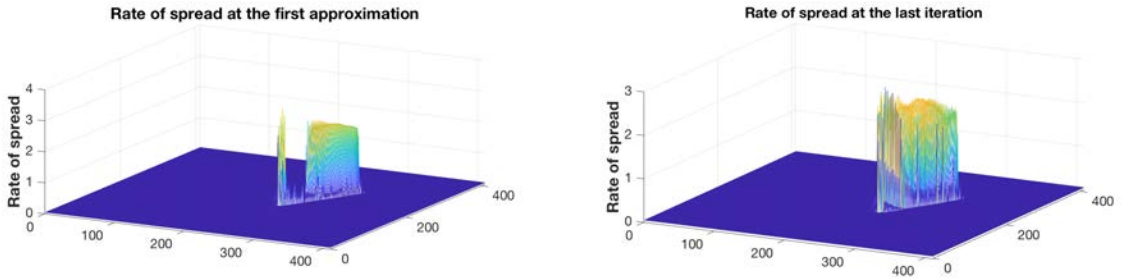


Figure 5.21: (a) Initial approximation from the ignition point to the first perimeter obtained with $\alpha = 1.4$. (b) Example of a bilinear coarse grid function at mesh step size 16. (c) Values of the objective function after each line search iteration of the projected multigrid method using dynamic rate of spread. (d) Result of the fire arrival time interpolation after 4 cycles of the projected multigrid method using dynamic rate of spread.

using dynamic rate of spread. One can observe that the hole in the middle of the rate of spread computed using the initial approximation of the fire arrival time (Figure 5.22a) is filled and well explained after all the iterations of the projected multigrid descent method using dynamic rate of spread (Figure 5.22b).

As in the previous ideal experiment, the initial approximation, the projected preconditioned gradient descent method and the projected multigrid descent method using or not using dynamic rate of spread are compared. The absolute percentage error is computed from equation (5.20) using the WRF-SFIRE simulated fire arrival time as the true state. In Figure 5.23a, the absolute percentage error of the initial approximation is plotted. One can observe a huge error surrounding the ignition point with an absolute percentage error close to 600%. Furthermore, the mean absolute percentage error computed from equation (5.21) of 103% indicates the necessity to improve this first guess. Using the projected preconditioned gradient descent method, one can observe a significant improvement near the ignition point and in general over the whole domain (Figure 5.23b). The surface plot of the absolute percentage error shows that the minimization improves the error in the diagonal of the perimeter where the winds drives fast the fire. Therefore, this fact and the mean absolute percentage error of 53% need still an improvement but show the improvement from the initial approximation. Then, Figure 5.23c shows the absolute percentage error using the projected multigrid descent method. A significant improvement from the previous two methods, the mean absolute percentage error of 51%, and no need to compute the gradient of the objective function



(a) (b)

Figure 5.22: (a) Rate of spread computed from the initial approximation of the fire arrival time T^0. (b) Rate of spread at the last iteration of the projected multigrid descent method.

show the capability of using a projected multigrid descent method. However, a mean absolute percentage error of 51% is still a significant amount of error. This gives the motivation to add the dependency of the fire arrival time on the rate of spread. Finally, Figure 5.23d shows the absolute percentage error using the projected multigrid descent method with dynamic rate of spread. A significant improvement from all the previous methods can be observed. The mean absolute percentage error is 48%. Therefore, the projected multigrid descent method using dynamic rate of spread is the best technique to solve the optimization problem proposed in this chapter.

5.5.3 WRF-SFIRE real case

The last step is to test the optimization method for a real case. Therefore, a real fire is simulated using WRF-SFIRE so, the ignition point, observed perimeters, and the necessary prognostic variables to compute the rate of spread are used in order to find the optimal fire arrival time. In these experiments, the exact or expected solution is not known. The method used in this section is the projected multigrid descent method with dynamic rate of spread.

Cougar Creek

The first real case to test is the Cougar Creek fire which occurred in Washington on the eastern slopes of Mount Adams in the late summer of 2015. In Figure 5.24, one can observe the ignition point and the first observed perimeter. Therefore, the main goal is to interpolate the fire arrival time between this ignition point $\vec{x}_i$ and the first observed perimeter Γ_1.

The first step is to interpolate the ignition point and Γ_1 as a linear system of constraints in the simulation grid. So, it is necessary to compute the interpolation operator matrix H and the right hand side g of the linear system of constraints $HT = g$ of the optimization problem using the aforementioned barycenter method (Section 5.4.1). Figure 5.25 shows the constraints of the minimization interpolated into the grid.

Then, a first approximation of the fire arrival time is computed using the squared minimization proposed in Section 5.4.2. In Figure 5.26, one can observe the result of

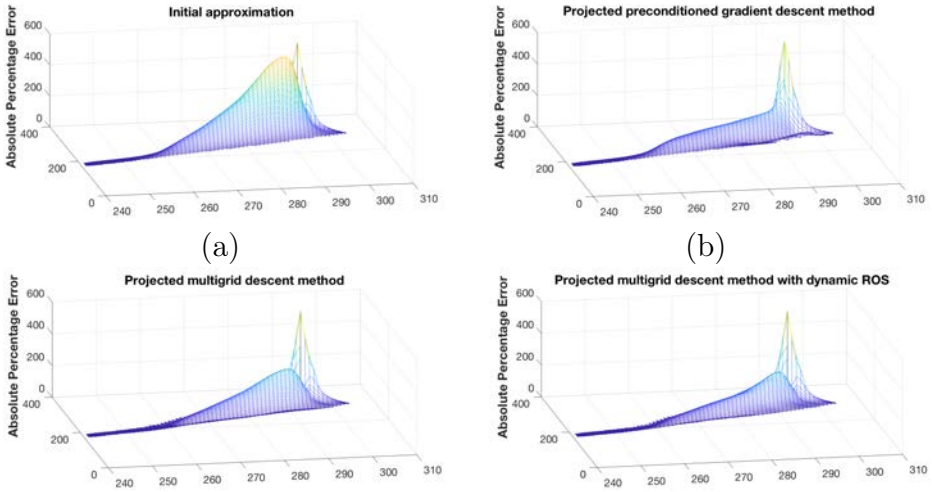


Figure 5.23: (a) Absolute percentage error of the initial approximation compared to the exact solution for the hill experiment. (b) Absolute percentage error of the projected preconditioned gradient descent method result compared to the exact solution for the hill experiment. (c) Absolute percentage error of the projected multigrid descent method compared to the exact solution for the hill experiment. (d) Absolute percentage error of the projected multigrid descent method using dynamic rate of spread compared to the exact solution for the hill experiment.

the first approximation which is going to be the first guess for the multigrid optimization with an objective function value of $J(T^0) = 608.3957$. The first approximation shows a slow progression at the beginning of the fire and then it speeds up at the end of the simulation. This fact should change in the projected multigrid descent method with dynamic rate of spread because of the heterogeneous rate of spread over the whole domain. Figure 5.27 shows the first rate of spread computed from the previous first approximation of the fire arrival time and the interpolated prognostic variables of Section 5.4.5.

The projected descent multigrid optimization method using dynamic rate of spread of Section 5.4.5 is used. In Figure 5.28b shows an example of one coarse bilinear function used for the multigrid minimization of mesh step size 16. In Figure 5.28c, one can observe that the solution improves faster in the coarsest step size iterations in each of the general 2 cycles. The final solution in Figure 5.28d shows an improvement over the initial approximation because of a less flat shape of the fire arrival time and the objective function value of $J(T) = 500.0837$.

Las Conchas

Trying to evaluate the projected multigrid descent method with dynamic rate of spread using another real case, the Las Conchas fire, reveals a difficulty. In this New Mexico fire which burned in 2011, different perimeters overlap (Figure 5.1). More precisely, the ignition point, the first perimeter and the second perimeter have some points lying in

Figure 5.24: Graphic of the ignition point $\vec{x_i}$ and first observed perimeter Γ_1 of the Cougar Creek fire.

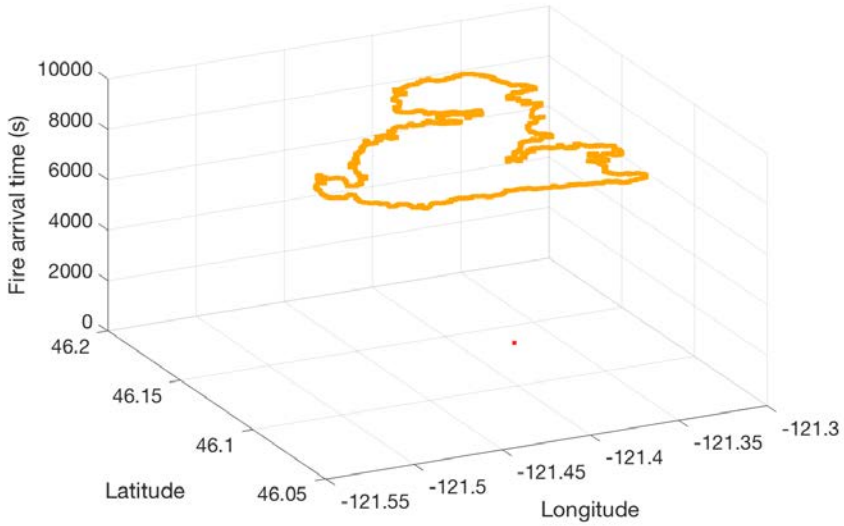


Figure 5.25: Graphic of the linear system of constraints generated by the barycenter method for the ignition point and the first perimeter of the Cougar Creek fire in Figure 5.24.

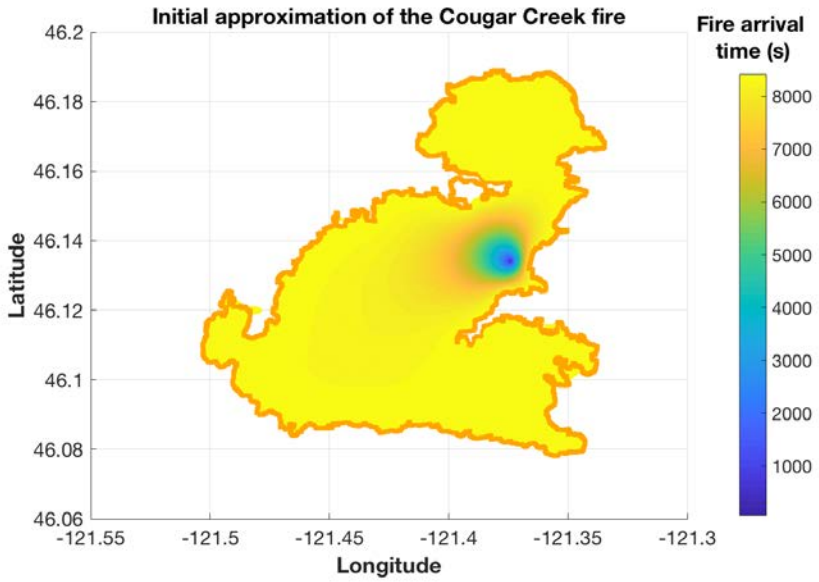


Figure 5.26: Graphic of the first approximation of the fire arrival time between the ignition point and the first observed perimeter of the Cougar Creek fire.

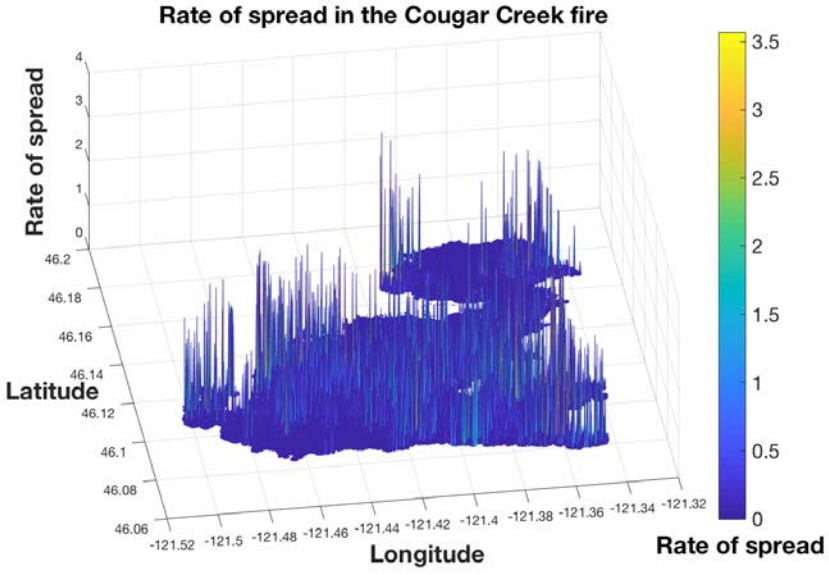


Figure 5.27: Graphic of the rate of spread between the ignition point and the first observed perimeter of the Cougar Creek fire.

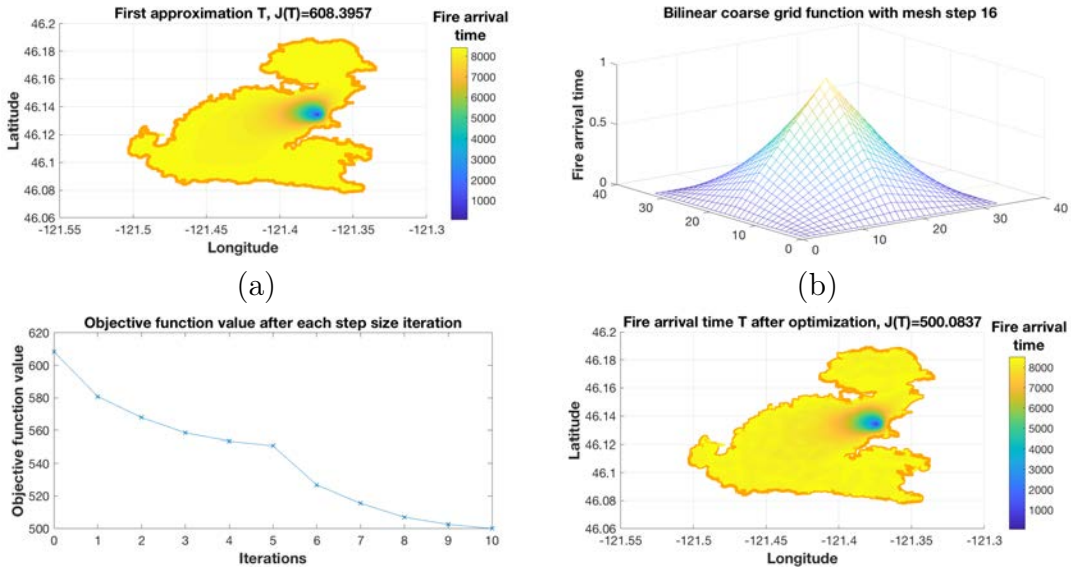


Figure 5.28: (a) Initial approximation from the ignition point to the first perimeter obtained with $\alpha = 1.4$ for the Cougar Creek fire. (b) Example of a bilinear coarse grid function at mesh step size 16. (c) Values of the objective function after each line search iteration of the projected multigrid method using dynamic rate of spread. (d) Result of the fire arrival time interpolation after 2 cycles of the projected multigrid method.

the same place. Therefore, when the barycenter method is applied, points from different perimeters lie in the same domain interpolation triangle. This makes it necessary to resolve an incompatibility because this fact means that at the interpolation triangle, the fire arrives at two different times. Mathematically, when two perimeters lie in the same triangle, the linear system of constraints $HT = g$ is inconsistent because the same linear combination of mesh nodes in H has a different right hand side g.

In this case, a solution where two very close perimeters are generated depending on the representatives of each perimeter in the grid in all the places where the perimeters coincide is proposed. This technique consists of a first step computing the squared minimization with Dirichlet boundary conditions proposed for the initial approximation in Section 5.4.2 but using an interior point of the first perimeter and the first perimeter as a constraints. In that way, one can obtain for all the points in the domain the T_g^k in Figure 5.29. Then, in each triangle where two perimeters coincide, the representative of each perimeter is determined using the minimum and maximum values of T_g^k in the triangle vertexes (Figure 5.29).

As one can observe in Figure 5.30, this new technique divides properly the constraints. However, when an ignition point and a perimeter lie in the same triangle, the first approximation in Section 5.4.2 is not capable of explaining how the solution expands from an ignition point very close to the perimeter and it generates local minima at different places (Figure 5.31a). Therefore, it tries to find the ignition point at different bad locations. For instance, if a synthetic ignition point is generated in the middle of the burned area, the solution obtained between the new ignition point and

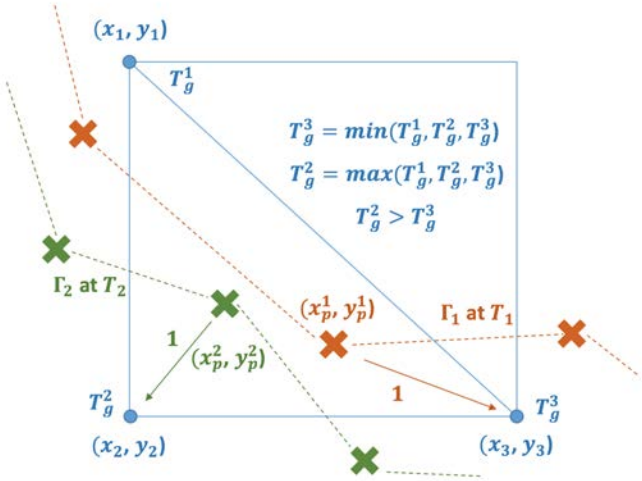


Figure 5.29: Graphical representation of the new technique which defines two new perimeter through the grid representatives of each perimeter in the cases where both perimeters lie at the same grid triangle.

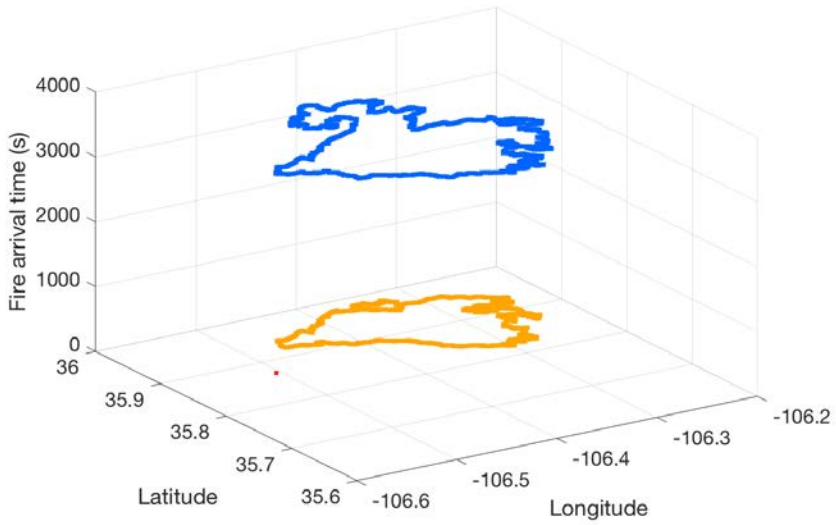
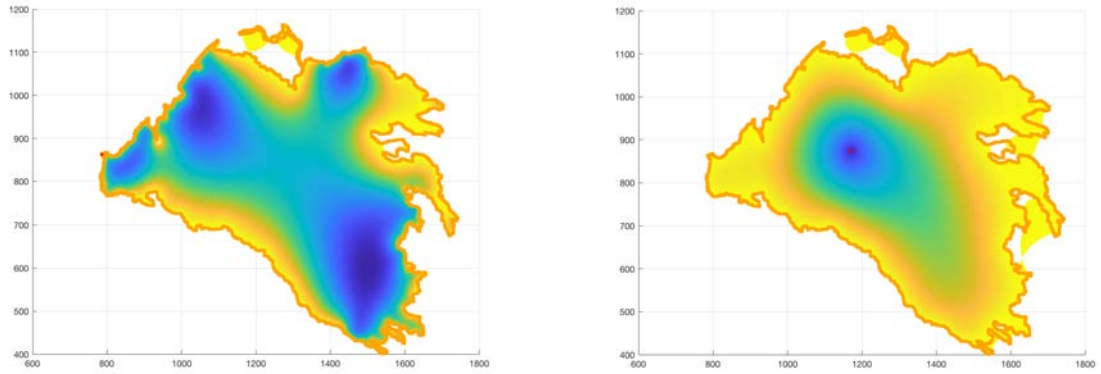


Figure 5.30: Graphic of the linear system of constraints generated by the barycenter method for the ignition point and the two first perimeters of the Las Conchas fire in Figure 5.1. The new method solving the problem when two perimeters lie in the same interpolation triangle is used (Figure 5.29).



(a) (b)

Figure 5.31: (a) Initial approximation between the ignition point and the first perimeter Γ_1 in the Las Conchas fire. (b) Initial approximation between a synthetic ignition point in the middle of the burned area and first perimeter Γ_1 in the Las Conchas fire.

the first perimeter Γ_1 is an acceptable first approximation to the optimization (Figure 5.31b). If in the square minimization of Section 5.4.2 a value of $\alpha = 1$ is used, the funnel shape ensures not violating the maximum principle. Using that and adding the first approximation between the first and the second perimeter, the result gives a logical first approximation (Figure 5.32).

In general, the situations where an ignition point is lying in a perimeter means that next to the ignition point there is a break like a road, river, etc. In that situations, one should not consider the fire arrival time because the fire can not propagate in that direction. Therefore, one should find a solution for that situations where these places have not to be computed (avoiding ignition process).

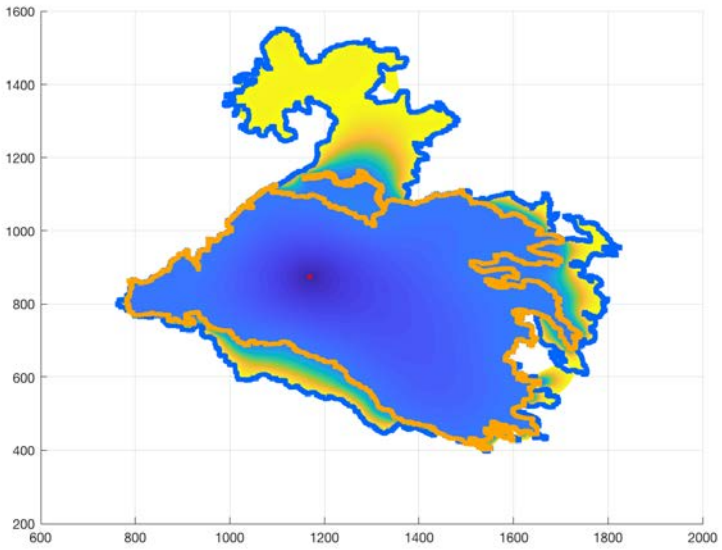


Figure 5.32: Initial approximation between the synthetic ignition point and the second perimeter Γ_2 going through the first perimeter Γ_1.

Chapter 6

Conclusions and Future Work

Wildfires are one of the major causes of environmental destruction every year worldwide. Any kind of prediction tool that can somehow accurately forecast the evolution of wildfires can help to contain them and plan appropriate damage mitigation strategies. One of the major challenges when simulating wildfires is the process of working with a multi-physical problem (combustion and atmosphere). In this direction, WRF-SFIRE is a coupled fire-atmosphere simulator which reproduces the interaction between the propagation of a fire and the atmosphere surrounding it (Chapter 2). This thesis proposes some improvements to this simulator in order to achieve a reliable and useful operational tool.

One can observe in Chapter 3 that the usage of a high resolution in the meteorological mesh is required in order to explain the coupling between the atmosphere and the fire. This makes it necessary to exploit high performance computing resources in order to give operational results. One could observe that the MPI parallel paradigm can give operational times when a large amount of MPI processes are used. However, the size domain restricts the amount of patches one can use to parallelize the simulation. OpenMP and MPI parallel paradigms can be combined. So, each squared distributed memory patch is divided in horizontal shared memory tiles. However, this configuration doesn't give better results than only using MPI parallelization. Therefore, the best parallel configuration is to use MPI distributed memory parallelism, using as many patches as the size of the domain allows. This contribution has generated the following publications:

- A. Farguell, J. Moré, A. Cortés, J.R. Miró, T. Margalef and V. Altava, *Reducing Data Uncertainty in Surface Meteorology using Data Assimilation: A Comparison Study*, Procedia Computer Science, Vol. 80, 1846-1855, 2016.

- Angel Farguell, Ana Cortés, Tomàs Margalef, Josep R. Miró, J. Mercader, *A Multi-physics wildfire Spread Model on Multi-core Systems*, Proceedings of the 17th International Conference on Computational and Mathematical Methods in Science and Engineering, CMMSE 2017, 853-861, 2017.

- A. Farguell, A. Cortés, T. Margalef, J.R. Miró, J. Mercader, *Data resolution effects on a coupled data driven system for wildfire propagation prediction*, Procedia Computer Science 108C 1562-1571, 2017.

- Angel Farguell, Ana Cortés, Tomàs Margalef, Josep R. Miró, J. Mercader, *Scalability of multi-physics system for wildfire spread prediction in multi-core platforms*, Journal of Supercomputing, DOI: 10.107/s11227-018-2330-9, 2018.

In this scalability study, possible future work could be fund by analyzing the bottlenecks of the MPI parallelization of WRF-SFIRE. Therefore, it would be interesting to perform analysis of the balance between communication and computation in order to determine if a parallel optimization of the MPI code in order to speed up the simulation is possible.

In Chapter 4, a solution for level set method with spatially highly variable rate of spread is proposed. The solution proposed consists in using a constraint for the level set method where local minima are avoided. It is shown that the proposed solution gives realistic results for the cases with spurious ignitions and is consistent with those did not have spurious ignitions. The proposed solution is implemented in the master branch of the WRF-SFIRE Git repository in order to avoid future spurious ignitions. The current version of the Git repository can be downloaded from [16]. However, the level set method has a conceptual problem dealing with the possibility of deforming the level set function in such a way that one can create spurious ignitions. This opens an interesting future line of research with the possibility of designing a new approach similar to the level set method.

Finally, in Chapter 5, a new method for fitting fire arrival time data by an approximate solution of the eikonal equation is proposed. In this context, the optimization problem of minimizing the residuals of the eikonal equation is formulated and solved using the projected preconditioned gradient descent method, giving an approximation for the gradient of an upwinding method. However, this method assumes rate of spread is constant. Therefore, the problem is also solved using the projected multigrid descent method both using and not using a dynamic rate of spread. These methods are illustrated on an idealized case, an idealized case from WRF-SFIRE simulation, and two real fire cases from WRF-SFIRE simulation. The method producing the best results in all the cases is the projected multigrid descent method using a dynamic rate of spread. For both idealized cases, the results are very close to the exact solutions.

For the Cougar Creek fire, it seems to give an accurate interpolation of the fire arrival time depending on the rate of spread. However, there is a problem when some constraints (ignition point or observed perimeters) lie in the same coordinates. The problem of different observed perimeters overlapping is solved using a new separation method. However, there is still a problem when the ignition point and the first observed perimeter coincide, which opens a new future work. This new method has been published in:

- Angel Farguell, James Haley, Adam Kochanski, Ana Cortés Fite, and Jan Mandel, *Assimilation of Fire Perimeters and Satellite Detections by Minimization of the Residual in a Fire Spread Model*, ICCS 2018, LCNS 10861, 711-723, 2018.

- Angel Farguell, James Haley, Jan Mandel, Adam Kochanski, Sher Schranz, *Assimilation of Fire Perimeters and Satellite Observations into a Coupled Fire-Atmosphere Model*, American Meteorological Society's 12th Fire and Forest Meteorology Symposium, 13-17 May 2018 in Boise, Idaho.

- James Haley, Angel Farguell, Adam Kochanski, Jan Mandel, Sher Schranz, *Data likelihood of Actives Fires Satellite Detection and Applicattions to Ignition Estimation and Data Assimilation*, VIII International Conference on wildfire Research (accepted) 2018.

As it has been shown along this chapter, the different contributions made in this work open new and interesting research lines. One current work, that clearly opens several computational and modeling challenges consists of including a third counter part in the multi-physics system used in this thesis work, the smoke. The key points that must be faced when including the smoke in the model are reported in:

- Adam Kochanski, Steven Krueger, Jan Mandel, Martin Vejmelka, Dalton Burke, James Haley, Angel Farguell, Sher Schranz, *Coupled fire-atmosphere-smoke forecasting: current capabilities and plans for the future*, VIII International Conference on wildfire Research (accepted) 2018.

Bibliography

[1] Wildfires burn more US forest. *Nature*, 538:292, oct 2016.

[2] Frank A Albini. A model for fire spread in wildland fuels by-radiation. *Combustion Science and Technology*, 42(5-6):229–258, 1985.

[3] D. H. Anderson, E. A. Catchpole, N. J. De Mestre, and T. Parkes. Modelling the spread of grass fires. *The Journal of the Australian Mathematical Society. Series B. Applied Mathematics*, 23(4):451–466, 1982.

[4] Hal E. Anderson. Aids to determining fuel models for estimating fire behavior. USDA Forest Service General Technical Report INT-122, 1982. `http://www.fs.fed.us/rm/pubs_int/int_gtr122.html`.

[5] P. L. Andrews. BehavePlus fire modeling system: past, present, and future. Paper J2.1, 7th Symposium on Fire and Forest Meteorology, 2007. `http://ams.confex.com/ams/pdfpapers/126669.pdf`, retrieved September 2011.

[6] Richard A. Anthes. Data assimilation and initialization of hurricane prediction models. *Journal of the Atmospheric Sciences*, 31(3):702–719, 1974.

[7] Stanley L Barnes. A technique for maximizing details in numerical weather map analysis. *Journal of Applied Meteorology*, 3(4):396–409, 1964.

[8] F. Brezzi and M. Fortin. *Mixed and Hybrid Finite Element Methods*. Springer-Verlag, New York – Berlin – Heidelberg, 1991.

[9] D.T. Butry, E.D. Mercer, J.P. Prestemon, J.M. Pye, and T.P. Holmes. What is the price of catastrophic wildfire? *Journal of Forestry*, 99(11):9–17, 2001.

[10] T. L. Clark, M. A. Jenkins, J. L. Coen, and D. R. Packham. A coupled atmosphere-fire model: Role of the convective Froude number and dynamic fingering at the fireline. *International Journal of Wildland Fire*, 6(4):177–190, 1996.

[11] Terry L. Clark, Janice Coen, and Don Latham. Description of a coupled atmosphere-fire model. *International Journal of Wildland Fire*, 13:49–64, 2004.

[12] Terry L. Clark, Marry Ann Jenkins, Janice Coen, and David Packham. A coupled atmospheric-fire model: Convective feedback on fire line dynamics. *J. Appl. Meteor*, 35:875–901, 1996.

[13] J. L. Coen. Simulation of the Big Elk Fire using coupled atmosphere-fire modeling. *International Journal of Wildland Fire*, 14(1):49–59, 2005.

[14] J. L. Coen, M. Cameron, J. Michalakes, E. G. Patton, P. J. Riggan, and K. Yedinak. WRF-Fire: Coupled weather-wildland fire modeling with the Weather Research and Forecasting model. *J. Appl. Meteor. Climatol.*, 52:16–38, 2013.

[15] Janice L. Coen. Modeling wildland fires: A description of the Coupled Atmosphere-Wildland Fire Environment model (CAWFE). NCAR Technical Note NCAR/TN-500+STR, `http://dx.doi.org/10.5065/D6K64G2G`, 2013.

[16] Open Wildland Fire Modeling Community. Github repository: A coupled weather-fire forecasting model built on top of weather research and forecasting (wrf). `https://github.com/openwfm/wrf-fire`. Accessed 2018-06-15.

[17] Open Wildland Fire Modeling E community Wiki. Open wiki to explain the state of the art about different wildland fire modeling approaches. `http://www.openwfm.org/wiki/Open_Wildland_Fire_Modeling_E_community_Wiki`. Accessed 2018-08-28.

[18] Philippe Courtier, E Andersson, W Heckley, D Vasiljevic, M Hamrud, A Hollingsworth, F Rabier, M Fisher, and J Pailleux. The ecmwf implementation of three-dimensional variational assimilation (3d-var). i: Formulation. *Quarterly Journal of the Royal Meteorological Society*, 124(550):1783–1807, 1998.

[19] George P Cressman. An operational objective analysis system. *Mon. Wea. Rev*, 87(10):367–374, 1959.

[20] John R Curry and Wallace L Fons. Rate of spread of surface fires in the ponderosa pine type of california. *Journal of Agricultural Research*, 57(4):239–267, 1938.

[21] Paulo M Fernandes and Hermínio S Botelho. A review of prescribed burning effectiveness in fire hazard reduction. *International Journal of Wildland Fire*, 12(2):117–128, jun 2003.

[22] J. B. Filippi, F. Bosseur, X. Pialat, P.A. Santoni, S. Strada, and C. Mari. Simulation of coupled fire/atmosphere interaction with the MesoNH-ForeFire models. *Journal of Combustion*, 2011:Article ID 540390, 2011.

[23] Jean-Baptiste Filippi, Frédéric Morandini, Jacques Henri Balbi, and David Rc Hill. Discrete event front-tracking simulation of a physical fire-spread model. *Simulation*, 86(10):629–646, 2010.

[24] Sarah Elise Finlay, Andrew Moffat, Rob Gazzard, David Baker, and Virginia Murray. Health Impacts of Wildfires. *PLoS Currents*, 4:e4f959951cce2c, nov 2012.

[25] Mark A. Finney. FARSITE: Fire Area Simulator – model development and evaluation. Res. Pap. RMRS-RP-4, Ogden, UT, USDA Forest Service, Rocky Mountain Research Station, 1998. `http://www.fs.fed.us/rm/pubs/rmrs_rp004.html`, retrieved December 2011.

[26] Mark A Finney. Calculation of fire spread rates across random landscapes. *International Journal of Wildland Fire*, 12(2):167–174, 2003.

[27] Wallace L Fons. Analysis of fire spread in light forest fuels. *Journal of Agricultural Research*, 72(3):93–121, 1946.

[28] SR Freitas, IC Menezes, R Stockler, R Mello, NA Ribeiro, JAM Corte-Real, and P Surovỳ. Application of a mesoscale atmospheric coupled fire model brams-fire to alentejo woodland fire and comparison of performance with the fire model wrf-sfire. In *AGU Fall Meeting Abstracts*, 2014.

[29] Generalitat de Catalunya, Departament d'Interior, and GRAF. Informe de l'incendi de cardona (08-07-05), 2005. `http://interior.gencat.cat/web/.content/home/030_arees_dactuacio/bombers/foc_forestal/consulta_incendis_forestals/informes_incendis_forestals/2000-2009/2005/20050708_REC_Cardona.pdf`. Accessed 2018-02-20.

[30] Klodiana Goga, Antonio Parodi, Pietro Ruiu, and Olivier Terzo. Performance analysis of wrf simulations in a public cloud and hpc environment. In Leonard Barolli and Olivier Terzo, editors, *Complex, Intelligent, and Software Intensive Systems*, pages 384–396, Cham, 2018. Springer International Publishing.

[31] James E. Hoke and Richard A. Anthes. The initialization of numerical models by a dynamic-initialization technique. *Monthly Weather Review*, 104(12):1551–1556, 1976.

[32] P. L. Houtekamer and Herschel L. Mitchell. Data assimilation using an ensemble kalman filter technique. *Monthly Weather Review*, 126(3):796–811, 1998.

[33] Shamoon Jamshed. *Using HPC for Computational Fluid Dynamics: A Guide to High Performance Computing for CFD Engineers*. Academic Press, 2015.

[34] H. Karimabadi, J. Driscoll, Y.A. Omelchenko, and N. Omidi. A new asynchronous methodology for modeling of physical systems: breaking the curse of courant condition. *Journal of Computational Physics*, 205(2):755 – 775, 2005.

[35] Inderpreet Kaur, Andrea Mentrelli, Frédéric Bosseur, Jean-Baptiste Filippi, and Gianni Pagnini. Turbulence and fire-spotting effects into wild-land fire simulators. *Communications in Nonlinear Science and Numerical Simulation*, 39:300 – 320, 2016.

[36] Adam Kochanski, M. A. Jenkins, S. K. Krueger, J. Mandel, J. D. Beezley, and C. B. Clements. Coupled atmosphere-fire simulations of FireFlux: Impacts of model resolution on its performance. Paper 10.2, Ninth Symposium on Fire and Forest Meteorology, Palm Springs, October 2011, 2011. Available at `http://ams.confex.com/ams/9FIRE/webprogram/Paper192314.html`.

[37] Adam K. Kochanski, Mary Ann Jenkins, Kara Yedinak, Jan Mandel, Jonathan Beezley, and Brian Lamb. Toward an integrated system for fire, smoke, and air quality simulations. *International Journal of Wildland Fire*, 25:534–546, 2016.

[38] Chris Lautenberger. Wildland fire modeling with an eulerian level set method and automated calibration. *Fire Safety Journal*, 62:289–298, 2013.

[39] R. Linn, J. Reisner, J. J. Colman, and J. Winterkamp. Studying wildfire behavior using FIRETEC. *Int. J. of Wildland Fire*, 11:233–246, 2002.

[40] Vivien Mallet, David E Keyes, and FE Fendell. Modeling wildland fire propagation with level set methods. *Computers & Mathematics with Applications*, 57(7):1089–1101, 2009.

[41] J. Mandel, S. Amram, J. D. Beezley, G. Kelman, A. K. Kochanski, V. Y. Kondratenko, B. H. Lynn, B. Regev, and M. Vejmelka. Recent advances and applications of WRF-SFIRE. *Natural Hazards and Earth System Science*, 14(10):2829–2845, 2014.

[42] Jan Mandel, Jonathan D. Beezley, Janice L. Coen, and Minjeong Kim. Data assimilation for wildland fires: Ensemble Kalman filters in coupled atmosphere-surface models. *IEEE Control Systems Magazine*, 29(3):47–65, June 2009.

[43] Jan Mandel, Jonathan D. Beezley, and Adam K. Kochanski. Coupled atmosphere-wildland fire modeling with WRF 3.3 and SFIRE 2011. *Geoscientific Model Development*, 4:591–610, 2011.

[44] Jan Mandel, Lynn S Bennethum, Jonathan D Beezley, Janice L Coen, Craig C Douglas, Minjeong Kim, and Anthony Vodacek. A wildland fire model with data assimilation. *Mathematics and Computers in Simulation*, 79(3):584–606, 2008.

[45] Jan Mandel and Bedřich Sousedík. Coarse space over the ages. In Yunqing Huang, Ralf Kornhuber, Olof Widlund, and Jinchao Xu, editors, *Domain Decomposition Methods in Science and Engineering XIX, Lecture Notes in Computational Science and Engineering 78, Part 2*, pages 213–220. Springer-Verlag, 2011.

[46] S. F. McCormick and J. W. Ruge. Unigrid for multigrid simulation. *Math. Comp.*, 41(163):43–62, 1983.

[47] W. Mell, M. A. Jenkins, J. Gould, and P. Cheney. A physics-based approach to modelling grassland fires. *Intl. J. Wildland Fire*, 16:1–22, 2007.

[48] Ana Cristina Mendes-Oliveira, Paulo Guilherme Pinheiro dos Santos, Oswaldo de Carvalho-Júnior, Luciano Fogaça de Assis Montag, Renata Cecília Soares de Lima, Suzanne Lúcia Silva de Maria, and Rogério Vieira Rossi. Edge effects and the impact of wildfires on populations of small non-volant mammals in the forest-savanna transition zone in Southern Amazonia. *Biota Neotropica*, 12:57 – 63, 09 2012.

[49] Francisco Moreira and Danilo Russo. Modelling the impact of agricultural abandonment and wildfires on vertebrate diversity in mediterranean europe. *Landscape Ecology*, 22(10):1461–1476, Dec 2007.

[50] Domingo Muñoz-Esparza, Branko Kosović, Pedro A. Jiménez, and Janice L. Coen. An accurate fire-spread algorithm in the Weather Research and Forecasting model using the level-set method. *Journal of Advances in Modeling Earth Systems*, 2018.

[51] State of California. Detwiler fire incident information. `http://cdfdata.fire.ca.gov/incidents/incidents_details_info?incident_id=1672`. Accessed 2018-08-30.

[52] Y.A. Omelchenko and H. Karimabadi. Hypers: A unidimensional asynchronous framework for multiscale hybrid simulations. *Journal of Computational Physics*, 231(4):1766 – 1780, 2012.

[53] Stanley Osher and Ronald Fedkiw. *Level Set Methods and Dynamic Implicit Surfaces*. Springer, New York, 2003.

[54] Stanley Osher and R P. Fedkiw. *The Level Set Methods and Dynamic Implicit Surfaces. Applied Mechanics Reviews*, volume 57. 01 2004.

[55] Stanley Osher and James A Sethian. Fronts propagating with curvature-dependent speed: algorithms based on hamilton-jacobi formulations. *Journal of computational physics*, 79(1):12–49, 1988.

[56] K. W. Outcalt and D. D. Wade. Fuels management reduces tree mortality from wildfires in southeastern United States. *Southern Journal of Applied Forestry*, 28(1):28–34, 2004.

[57] Travis B Paveglio, Hannah Brenkert-Smith, Troy Hall, and Alistair M S Smith. Understanding social impact from wildfires: advancing means for assessment. *International Journal of Wildland Fire*, 24(2):212–224, 2015.

[58] Florence Rabier, H Järvinen, E Klinker, J-F Mahfouf, and A Simmons. The ecmwf operational implementation of four-dimensional variational assimilation. i: Experimental results with simplified physics. *Quarterly Journal of the Royal Meteorological Society*, 126(564):1143–1170, 2000.

[59] Ronald G. Rehm and Randall J. McDermott. Fire-front propagation using the level set method. NIST Technical Note 1611, March 2009. `https://nvlpubs.nist.gov/nistpubs/Legacy/TN/nbstechnicalnote1611.pdf`, accessed May 2018.

[60] Weather Research and Forecasting model official website. User's guide for advanced research wrf modeling system version 3.7. [online], last visit 26 of March 2015. `www2.mmm.ucar.edu/wrf/users/docs/user_guide_V3/contents.html`.

[61] Gwynfor D Richards. An elliptical growth model of forest fire fronts and its numerical solution. *International Journal for Numerical Methods in Engineering*, 30(6):1163–1179, 1990.

[62] Gwynfor D Richards. A general mathematical framework for modeling two-dimensional wildland fire spread. *International Journal of Wildland Fire*, 5(2):63–72, 1995.

[63] M. C. Rochoux, S. Ricci, D. Lucor, B. Cuenot, and A. Trouvé. Towards predictive data-driven simulations of wildfire spread – part i: Reduced-cost ensemble kalman filter based on a polynomial chaos surrogate model for parameter estimation. *Natural Hazards and Earth System Sciences*, 14(11):2951–2973, 2014.

[64] Richard C. Rothermel. A mathematical model for predicting fire spread in wild land fuels, 1972.

[65] Richard C. Rothermel. A mathematical model for predicting fire spread in wildland fires. USDA Forest Service Research Paper INT-115, 1972. `https://www.fs.fed.us/rm/pubs_int/int_rp115.pdf`, accessed March 2018.

[66] Ian D. Rutherford. Data assimilation by statistical interpolation of forecast error fields. *Journal of the Atmospheric Sciences*, 29(5):809–815, 1972.

[67] Servei Meteorològic de Catalunya (METEOCAT). Official website. `http://www.meteo.cat`. Accessed 2018-02-20.

[68] William C. Skamarock, Joseph B. Klemp, Jimy Dudhia, David O. Gill, Dale M. Barker, Michael G. Duda, Xiang-Yu Huang, Wei Wang, and Jordan G. Powers. A description of the Advanced Research WRF version 3. NCAR Technical Note 475, 2008. `http://www.mmm.ucar.edu/wrf/users/docs/arw_v3.pdf`, retrieved December 2011.

[69] William C Skamarock, Joseph B Klemp, Jimy Dudhia, David O Gill, Dale M Barker, Wei Wang, and Jordan G Powers. A description of the advanced research wrf version 2. Technical report, National Center For Atmospheric Research Boulder Co Mesoscale and Microscale Meteorology Div, 2005.

[70] Thayjes Srivas, Raymond A. de Callafon, Daniel Crawl, and Ilkay Altintas. Data assimilation of wildfires with fuel adjustment factors in farsite using ensemble kalman filtering**this work is funded by nsf 1331615 under ci, information technology research and sees hazards programs. *Procedia Computer Science*, 108:1572 – 1581, 2017. International Conference on Computational Science, ICCS 2017, 12-14 June 2017, Zurich, Switzerland.

[71] Scott L. Stephens and Lawrence W. Ruth. Federal forest-fire policy in the United States. *Ecological Applications*, 15(2):532–542, 2005.

[72] C. Tymstra, R. Bryce, B. Wotton, S. Taylor, and O. Armitage. Development and structure of Prometheus: the canadian wildland fire growth simulation model. Information Report NOR-X-147, Northern Forestry Centre, Canadian Forest Service, 2010. `http://publications.gc.ca/collections/collection_2010/nrcan/Fo133-1-417-eng.pdf`, accessed March 2018.

[73] Martin Vejmelka, Adam K. Kochanski, and Jan Mandel. Data assimilation of dead fuel moisture observations from remote automatic weather stations. *International Journal of Wildland Fire*, 25:558–568, 2016.

[74] J Yoder, D Engle, and S. Fuhlendorf. Liability, incentives, and prescribed fire for ecosystem management. *Frontiers in Ecology and the Environment*, 2:361–366, Jan 2004.