

Control of spatio-temporal pattern formation governed by geometrical models of interface evolution

Adrian Bouaru



A thesis submitted to the University of Sheffield for the degree of Doctor of Philosophy

Department of Automatic Control and Systems Engineering

University of Sheffield

Sheffield, UK

2015

Abstract

Numerous natural phenomena are characterized by spatio-temporal dynamics which give rise to time evolving spatial patterns. Although studies that address the problem of modelling these complex dynamics exist, a model based control approach for such systems is still a challenging task. The work in this thesis is concerned with the development of control methods for such spatio-temporal systems, where interface growth is represented using a geometric evolution law. In particular, the focus is set on the control of dendritic crystal growth and wind aided wildfire spread.

In the first section of the thesis several modeling approaches used to describe spatio-temporal systems are discussed and partial differential equation models are selected as a suitable class of models for describing the system dynamics. Following this, the complex nonlinear dynamics found in dendritic crystal growth and wind aided wildfire evolution are analyzed through the scope of geometrical partial differential equation models of interface evolution.

In the next part of the thesis, a new method for controlling the shape of evolving interfaces, such as the boundaries of dendritic crystals, described by a temperature-dependent geometrical evolution law, where the normal velocity of the growing interface depends on both local curvature and temperature, is presented. A study of the control method in both two and three dimensions is provided, using both linear and non-linear interface dynamics, to demonstrate the effectiveness of the proposed method.

Following this, a novel model-based approach for controlling the evolving complex-shaped fire front using a level-set formulation of the model is presented. The work shows how key influencing parameters such as terrain topography and fuel loading, extracted directly from satellite imaging data, can be incorporated in the model. Simulation results show the numerical method is computationally efficient and examples are given with both synthetic and real data to illustrate the robustness of the containment method against changing wind speed, fuel loading and terrain topography.

Based on the developments made in the previous chapters, a new optimization method for controlling wildfire evolution is presented. Using a level set formulation of the evolving interface, a cost function is derived that allows the selection of a spatial location, such that the firefighting action has the best results with respect to a predefined target (minimum fire area increase etc.). Several tuning parameters are incorporated within the optimization algorithm, enabling a wide range of fire combat scenarios to be implemented. Numerical simulations provided demonstrate the effectiveness of the method.

Acknowledgments

I would like to thank my supervisor Daniel Coca for offering me the opportunity to become part of the exciting research area of spatio-temporal systems. I am grateful for his constant guidance and support and his extremely valuable insight into the study of spatio-temporal systems throughout the PhD.

I would also like to thank my colleagues from Office Room 316, with whom I've had numerous enjoyable discussions. My thanks and appreciations go to the Department of Automatic Control and Systems Engineering which has provided the support and resources I have needed to complete my thesis. I gratefully acknowledge that this work has been supported by the UK Engineering and Physical Sciences Research Council under the grant award EP/G042209/1.

In addition I would like to thank my friends and family who have supported me through the entire period. Most of all I would like to thank my loving wife, who helped me enormously and with whom I've shared most of the stress of going through a PhD. Her support has been invaluable throughout this entire period.

Table of contents

Abstract

Acknowledgements

List of figures

List of tables

1. Introduction	1
1.1. Background to the project and Motivation	3
1.2. Dendritic crystal growth	3
1.3. Wild fire propagation	5
1.4. Aims and objectives	9
1.5. Overview of the thesis	9
2. Mathematical representations of spatio-temporal systems	13
2.1. Introduction.....	13
2.2. Cellular automata models.....	13
2.3. Coupled map lattice models.....	16
2.4. Partial differential equations	17
2.5. Conclusions.....	21
3. Models of dendritic crystal growth	23
3.1. Introduction.....	23
3.2 Models of interface evolution	23
3.2.1. Geometrical models of dendritic growth.....	23
3.2.2. The Stefan problem.....	27
3.2.3. Cellular automata models for dendritic crystal growth	29
3.3. Numerical methods for solving interface evolution problems.....	30
3.4. Numerical studies of crystal growth in 2D	34
3.4.1. Time and space discretization	34
3.4.2. Simulation of geometric models of dendritic crystal growth	35
3.4.2.1. Motion of level sets in the normal direction	35
3.4.2.2. Motion of level sets under curvature flow	38
3.4.2.3. Motion of level sets under curvature flow with advection in the normal direction	39

3.4.2.4. Motion of level sets under the second derivative of curvature.....	40
3.4.2.5. Numerical difficulties of geometric models of dendritic crystal growth	42
3.4.3. Simulation of the modified Stefan problem	43
3.4.3.1. Velocity extension.....	43
3.4.3.2. Re-initialization of the level set function	44
3.4.3.3. Temperature update.....	45
3.4.3.4. Simulation results.....	47
3.4.4. CML simulation of dendritic crystal growth	49
3.4.4.1. Model description	49
3.4.4.2. Model simulations	50
3.5. Conclusions.....	52
4. Models of wildfire propagation	53
4.1. Introduction.....	53
4.2. Wind aided fire evolution models.....	53
4.3. Numerical studies of forest fire propagation.....	62
4.3.1. Simulation of geometric models of wildfire evolution	62
4.3.1.1. Basic wildfire evolution	65
4.3.1.2. Evolution under wind speed magnitude and wind direction variation.....	67
4.3.1.3. Fire islands and multiple fire fronts	69
4.3.1.4. Effects of fuel density on fire front propagation.....	70
4.3.1.5. Terrain elevation effects on wildfire propagation.....	72
4.4. Conclusions.....	77
5. A new approach for controlling crystal growth using a level set formulation	78
5.1. Introduction.....	78
5.2. Level set formulation of dendritic crystal growth.....	79
5.3. Control of evolving interfaces governed by geometric laws of evolution	81
5.4. Discretization	82
5.5. Extension to three dimensions	83
5.6. Numerical simulations of the crystal growth control method.....	85
5.6.1. Synthetic data simulations	86
5.6.2. Simulations using experimental data	92
5.6.3. Controller performance comparison	94
5.7. Conclusions.....	97

6. A model-based control approach for managing wildland fires	99
6.1. Introduction.....	99
6.2. Model-based containment of wildfires using geometrical models of interface evolution	100
6.3. Numerical implementation of the wildfire suppression method	102
6.3.1. Synthetic data simulations	103
6.3.2. Satellite terrain data simulations	105
6.3.3. Terrain elevation from DTED models	111
6.4. Conclusions.....	116
7. Optimized boundary control for wildland fire fronts	117
7.1. Introduction.....	117
7.2. Optimized boundary control of wildfires using geometrical models of interface evolution.....	118
7.3. Numerical implementation of the optimized control problem.....	121
7.3.1. Numerical discretization of the control algorithm	121
7.3.2. Synthetic data simulations	122
7.3.3. Satellite and DTED data simulations	128
7.4. Conclusions.....	129
8. Conclusion	130
8.1. Introduction.....	130
8.2. Thesis summary	130
8.3. Future research directions	133
Bibliography	135

List of Figures

<i>Fig. 1.1.</i> Growth pattern of NH_4Br dendrite crystal.....	4
<i>Fig. 1.2.</i> Forest fire life cycle	8
<i>Fig. 2.1.</i> CA lattice diversity (two dimensional CA) – square, hexagonal and triangular lattice structures	14
<i>Fig. 2.2.</i> CA neighborhood – 1D. a) radial, $R = 1$ b) radial, $R = 2$	14
<i>Fig. 2.3.</i> CA neighborhood – 2D. a) von Neumann neighborhood b) Moore neighborhood c) simple hexagonal neighborhood.....	15
<i>Fig. 2.4.</i> Volume element $dv = dx \cdot dy \cdot dz$ undergoing heat transfer	19
<i>Fig. 3.1.</i> Time variation of contour $C(t)$	24
<i>Fig. 3.2.</i> Solid phase $\Omega_S(t)$, liquid phase $\Omega_L(t)$ and the free boundary $\Gamma(t)$ in the domain Ω	27
<i>Fig. 3.3.</i> Embedding of a circle as the zero level-set of a higher dimensional function	32
<i>Fig. 3.4.</i> Evolution of a star shaped curve under positive normal velocity	37
<i>Fig. 3.5.</i> Evolution of a star shaped curve under negative normal velocity	37
<i>Fig. 3.6.</i> Evolution of a star shaped curve under curvature flow	39
<i>Fig. 3.7.</i> Evolution of a star shaped under speed equal to the curvature combined with a normal speed term. Perturbations on the curve are diminished in time, and the shape tends to turn circular....	40
<i>Fig. 3.8.</i> A circle under curvature flow given by k_{SS} , remains stationary (after 20 000 iterations)	41
<i>Fig. 3.9.</i> A star shape curve evolving under curvature flow given by k_{SS} , transforms approximately into a circle after 20 000 iterations.....	42
<i>Fig. 3.10.</i> a) Evolution of a circular seed with radius $r = 0.1$ over a time interval $\Delta T = 0.8s$.The time levels are shown in increments of 0.1 seconds. b) The temperature distribution at final time ΔT	48
<i>Fig. 3.11.</i> a) Evolution of a square seed with $l = 0.1$ over a time interval $\Delta T = 0.8s$.The time levels are shown in increments of 0.1 seconds. b) The temperature distribution at final time ΔT	48
<i>Fig. 3.12.</i> a) Evolution of a four sided star shape seed over a time interval $\Delta T = 0.8s$.The time levels are shown in increments of 0.1 seconds. b) The temperature distribution at final time ΔT	49
<i>Fig. 3.13.</i> a) Crystal growing in undercooled melt after $\Delta T = 1.5$ seconds using the normal Von Neumann capture rule b) Temperature distribution at final time ΔT	51

<i>Fig. 3.14. a)</i> Crystal growing in undercooled melt after $\Delta T = 1$ second using the normal Moore capture rule <i>b)</i> Temperature distribution at final time ΔT	51
<i>Fig. 4.1.</i> Time variation of contour $\Gamma(s, t)$ in time Δt	55
<i>Fig. 4.2.</i> Fire front evolution as a function of wind speed U and wind direction θ . Initial wildfire is considered to be a circle or radius R	58
<i>Fig. 4.3.</i> Fire front evolution initial boundary and computational domain	66
<i>Fig. 4.4.</i> Fire front evolution of a circular initial fire front (red), under $\Theta = [0.2, 0.5, 0.5]$ with constant wind speed $U = 10$ blowing in the S-N direction. The time level sets are shown in increments of $\Delta t_{plot} = 0.25$	66
<i>Fig. 4.5.</i> Time evolution of the area enclosed by the fire front under constant wind speed	67
<i>Fig. 4.6.</i> Time evolution of fire front arc length under constant wind speed	67
<i>Fig. 4.7.</i> Time evolution of fire front under changing wind speed	68
<i>Fig. 4.8.</i> Time evolution of the area enclosed by the fire front under changing wind speed	69
<i>Fig. 4.9.</i> Time evolution of the fire front arc length under changing wind speed	70
<i>Fig. 4.10.</i> Time evolution of fire front under changing wind speed and wind direction	71
<i>Fig. 4.11.</i> Time evolution of the fire front area under changing wind speed and wind direction	71
<i>Fig. 4.12.</i> Time evolution of the fire front arc length under changing wind speed and wind direction ...	72
<i>Fig. 4.13.</i> Time evolution of multiple independent fire fronts	73
<i>Fig. 4.14.</i> Time evolution of multiple fire fronts with unburnt biofuel islands	74
<i>Fig. 4.15. a)</i> Biofuel density <i>b)</i> Initial wildfires under varying fuel density terrain – green, black, magenta, cyan regions are defined according to a)	75
<i>Fig. 4.16.</i> Wildfire evolution under varying fuel density	75
<i>Fig. 4.17.</i> Terrain slope map (e^μ), $\mu_{down} = -0.7, \mu_{up} = 0.2$	76
<i>Fig. 4.18.</i> Wildfire evolution under varying terrain topography	76
<i>Fig. 5.1.</i> Initial interface (regular octahedron)	85
<i>Fig. 5.2.</i> Evolution of a regular octahedron under normal velocity given by curvature and advection in the normal direction.....	86
<i>Fig. 5.3.</i> Setup for the 2D control problem	87
<i>Fig. 5.4.</i> Time evolution of the initial crystal seed. Time plots (blue) shown in increments of $\Delta t_{plot} = 0.1$ seconds	88
<i>Fig. 5.5.</i> Arc length error 2D synthetic seed	88
<i>Fig. 5.6.</i> Area error 2D synthetic seed	89
<i>Fig. 5.7.</i> Global error 2D synthetic seed	89
<i>Fig. 5.8.</i> Initial conditions for the 3D control problem	90

<i>Fig. 5.9.</i> 3D controlled growth: a) system evolution after time T_{sim} b) Reference shape	90
<i>Fig. 5.10.</i> Surface area error – 3D control	91
<i>Fig. 5.11.</i> Enclosed volume error – 3D control	91
<i>Fig. 5.12.</i> Growth pattern of NH_4Br dendrite crystal	92
<i>Fig. 5.13.</i> a) Segmented image (central dendrite) b) Boundary of NH_4Br dendrite crystal	92
<i>Fig. 5.14.</i> Initial control setup using NH_4Br dendrite crystal reference	93
<i>Fig. 5.15.</i> Time evolution of the controlled system using NH_4Br dendrite crystal reference. Time plots are shown in increments of $\Delta t_{plot} = 0.1$ seconds	94
<i>Fig. 5.16.</i> Arc length error NH_4Br dendrite crystal reference	95
<i>Fig. 5.17.</i> Area error NH_4Br dendrite crystal reference	95
<i>Fig. 5.18.</i> Global error NH_4Br dendrite crystal reference	96
<i>Fig. 5.19.</i> Controller comparison initial setup	96
<i>Fig. 5.20.</i> Response comparison of P, PI and PID controllers	97
<i>Fig. 6.1.</i> Profiles obtained by varying the power term m	102
<i>Fig. 6.2.</i> Initial setup for wildfire evolution	103
<i>Fig. 6.3.</i> Uncontrolled wildfire evolution	104
<i>Fig. 6.4.</i> Controlled wildfire evolution	104
<i>Fig. 6.5.</i> Multiple wildfire detection	105
<i>Fig. 6.6.</i> Multiple wildfires – a) uncontrolled evolution b) suppression active	105
<i>Fig. 6.7.</i> Region of interest – Color background view	106
<i>Fig. 6.8.</i> Region of interest – Water vapor view	106
<i>Fig. 6.9.</i> Region of interest – Infrared enhancement view	107
<i>Fig. 6.10.</i> Satellite acquired terrain data	108
<i>Fig. 6.11.</i> Fuel loading map – a) Terrain 1, b) Terrain 2, c) Terrain 3	109
<i>Fig. 6.12.</i> Wildfire evolution under suppression agent, level sets are shown in time increments of $\Delta t_{plot} = 30$ seconds – a) Terrain 1, b) Terrain 2, c) Terrain 3	110
<i>Fig. 6.13.</i> Terrain elevation map	112
<i>Fig. 6.14.</i> Neighborhood of 8 points used to estimate the slope of the central point Z_0	113
<i>Fig. 6.15.</i> Terrain slope estimate of $Reg_{observed}$	113
<i>Fig. 6.16.</i> Terrain topography satellite image – http://goo.gl/maps/uc1D8	114
<i>Fig. 6.17.</i> Terrain topography – fuel loading map	114
<i>Fig. 6.18.</i> Unconstrained wildfire evolution for the data fusion model. Time plots are shown in increments of $\Delta t_{plot} = 15$ seconds until final time $T_{sim} = 150$ seconds	115

<i>Fig. 6.19.</i> Wildfire evolution under suppression agent for the data fusion model. Time plots are shown in increments of $\Delta t_{plot} = 40$ seconds until final time $T_{sim} = 400$ seconds	115
<i>Fig. 7.1.</i> Initial setup for wildfire evolution.....	123
<i>Fig. 7.2.</i> Sector division at time $t = 0$	124
<i>Fig. 7.3.</i> Level set functions $\varphi_c^1, \varphi_c^2, \varphi_c^3, \varphi_c^4$ attached to the four sectors at time $t = 0$	124
<i>Fig. 7.4.</i> Normalized distance function $\bar{d}_{\Gamma_r}(0)$	125
<i>Fig. 7.5.</i> a) Controlled fire front evolution b) Uncontrolled fire front evolution.....	125
<i>Fig. 7.6.</i> a) Normalized distance function \bar{d}_{Γ_r} at time $t = 20$ - Controlled case b) Normalized distance function \bar{d}_{Γ_r} at time $t = 20$ - Uncontrolled case	126
<i>Fig. 7.7.</i> Suppressed fire evolution under varying δ_c	126
<i>Fig. 7.8.</i> Suppressed fire evolution under varying division numbers M	127
<i>Fig. 7.9.</i> Suppressed fire evolution under varying δ_{min}	128
<i>Fig. 7.10.</i> a) DTED model controlled evolution b) DTED model unconstrained evolution.....	129

List of Tables

Table 6.1. Digital Terrain Data accuracy	111
--	-----

1. Introduction

1.1. Background to the project and Motivation

Dynamical systems where the evolution depends on both the spatial interactions and the time dynamics of the neighboring spatial locations of the system are called spatio-temporal systems. Therefore, such systems evolve in both space and time, and the dynamics at a unique spatial location taken at a specific point in time is dependent on the dynamics of the system at other spatial locations and the evolutionary history back in time. In contrast, purely temporal systems involve measurements of a variable over time and are not concerned with the spatial interactions within the system. As a result, spatio-temporal systems can describe a wide range of highly complex natural phenomena, where the spatial interactions within the system produce time evolving patterns. Examples of spatio-temporal systems include crystal growth, fluid flows, combustion and flame propagation (wild land fire spread), climate systems, self-organized cellular systems.

In recent years, numerical modelling, simulation and control of interface evolution problems that appear in such spatio-temporal systems has become of great interest. This current study is focused on simulation and control of such interface evolution problems, with application in two major areas: crystal growth and wind-aided forest fire spread. Although these phenomena are quite opposite by nature, given the fact that one implies solidification at relatively low temperatures, and the other is related to burning of biofuels (dried grass, shrub, timber etc.), both their dynamics can be modelled using similar techniques. In both cases, an initial interface (initial crystal seed, initial fire front) evolves in time, depending on numerous environment variables, to create complex patterns. Developing a systematic methodology of controlling the mode in which these interfaces evolve poses significant challenges to researchers and is an area of great interest.

Solidification processes are amongst the most common phase change processes, where an interface separating two phases (e.g. liquid and solid) evolves in time to create spatio-temporal patterns, such as dendrite crystals. Dendritic crystal structures are found in numerous systems in nature, both organic and inorganic, where many complex patterns emerge from simple conditions. One of the most common examples of dendrite is the snow crystal (Nakaya, 1954, Kobayashi, 1993), but dendrite-like patterns have been observed in other substances such as pure metals (Warren, 1995), alloys (J.F. McCarthy and N.W. Blake, 1996) or non-metallic substances, such as helium (Rolley et. al, 1994). Crystal growth has

been a topic of study in numerous areas, such as physics, material science, physical chemistry, and pharmaceutical or food industries. Thus, crystal growth is a multidisciplinary field and there is currently a great interest in understanding the principles governing pattern formation in such systems. Solution crystallization is used to obtain many pharmaceutical products, where the morphology and structure of the crystals influence chemical properties of the drugs and their therapeutic performance (Lee and Myerson, 2006). Another example is in the field of material science, in the case of steel casting, where during solidification dendrite structures form in the melt with a specific growth rate, shape and size, which determine the properties of the solidified material, such as mechanical strength and electrical conductivity. In many cases metal solidification causes formation of dendrite structures that affect the quality of the final material (Govindaraju, 2002). Hence, for such processes where the properties of materials are determined by crystal structures, control and prediction of dendritic formation is of great interest. The control of crystal growth poses many difficulties due to the speed of the process and the numerous parameters that determine the evolution of the growth.

The second application of spatio-temporal models that is considered in this work is the modeling and control of forest fires. Wild land fires are a natural occurring part of the environment and have been a topic of study for several decades, and the search for a mathematical model of evolution for fire spread has been a topic of intense research (Johnson et al, 2001).

Wild fires can occur by both natural and artificial (manmade) causes, and have devastating effects. Regions where the climate is predominantly dry favor the ignition and spread of fires. In these conditions the sun's heat or a lightning strike is enough to spark a fire. This can be considered as natural path of recycling, in the sense that the elimination of dead, useless vegetation can make space for a new repopulation of healthy vegetation. However, while there are few benefits of eliminating dried flora, in most cases destruction of vegetation often is correlated with decreases in fauna population. Human related activities and negligence, in particular, can also cause wild fires, and the most common causes are unattended campfires, inappropriate waste burning, fireworks, smoking or intentional arson.

Wild land fires are responsible for numerous types of overwhelming damage. A direct and obvious example is the massive loss of forests and shrub, which in turn causes several chain effects. In general, areas covered by forests, create a perfect setup for wildlife habitat and the loss of forests causes large drops in environmental biodiversity, favors greenhouse effects and increases soil deterioration. The damages are not limited solely to uninhabited areas. The dynamic nature of wildfires implies that evolution behavior is difficult to predict, especially when the fires reach the so called wild land-urban interface (Rehm et al 2009). In such cases, damages which are measured in loss of human lives and financial properties, are considerably greater (National Interagency Fire Center, 2012).

As a direct result of these highly unwanted outcomes, development of mathematical and numerical models of interface evolution for the rate of spread of wild fires and front movement prediction is crucial for a better understanding, management and prevention of these unwanted events. However, this is not an easy task, due to the fact that fires are influenced by a large number of factors, including the biofuel type, fuel loading, terrain slope factors, ignition factors, environment moisture, wind speed etc. (Rothermel, 1972).

1.2. Dendritic crystal growth

In this section a short presentation of the physical process that describes the formation of crystalline solids from melts is given. The process of crystallization is complex and will not be captured in detail in this section. Crystallization begins with the creation of a cluster of atoms of crystalline structure that grow in the melt to form nuclei, a process known as nucleation. The phase transformation (solidification) requires rearrangement of the atoms, which causes small atomic-scale movements. This process is due to extraction of heat from the melt in a more or less controlled manner. Once nucleation has occurred, the nuclei continue to grow, forming the macroscopic crystal, influenced by the kinetics of atom attachment to the interface, capillarity and diffusion of heat and mass (Dantzig and Rappaz, 2009).

To explain the formation of dendritic crystals, consider a container of a pure substance in liquid state, heated to a certain temperature, high enough to ensure that there are no crystals in the melt. Consider the temperature is lowered below the freezing point of the liquid. If the cooling is done in such a manner that the liquid does not change phase, e.g. the system is in metastable equilibrium, any small perturbations, such as introducing a seed in the liquid (heterogeneous nucleation), or the spontaneous formation of a crystalline nucleus (homogeneous nucleation) cause a rapid and unstable phase change, namely dendritic crystal growth. The evolving solid-liquid interface takes a tree-like form (called a dendrite), consisting of a primary dendrite arm with branches that extend outward (secondary dendrite arms). This is the case of isolated growth (unconstrained growth) for a pure substance, where an isolated dendrite grows from an undercooled melt, and the growth is controlled mostly by heat flow (Fig 1.1.). When the heat flows from the solid into the melt (latent heat), the dendrites grow in a free manner, with a speed imposed mostly by the undercooling. The process is naturally unstable and the crystal will continue to grow until constrains due to surface tension or interface kinetic effects stabilize the growth.

During the last decades, considerable theoretical effort was expended with the purpose of providing a better understanding of the mechanisms that control the pattern formation during dendritic crystal growth (Glicksman et al, 1976, Langer, 1980, Brower et. al, 1984, Kessler et. al, 1985, Kessler et. al, 1986,

Kessler and Levine, 1986, Martin and Goldfield, 1987, Strain, 1990, Sethian and Strain, 1992). Despite many previous efforts to analyze and model the pattern formation in dendritic crystal growth, the mechanism by which the system evolves is not yet fully understood.

Several mathematical models have been proposed to simulate dendritic growth. Two of the most common approaches to model crystal growth are the local geometrical models of evolution (Kessler et al., 1984, Zhao et al., 2011, Zhao et al., 2007), which take into account properties of the solid-liquid interface, such as curvature; another method is to model growth by non-local models that take into account other global parameters of the system (Gibou et al., 2003). Other approaches to model solidification exist in literature, that are somewhat a different approach from local and non-local models, such as cellular automata models (Zhao et al., 2009).

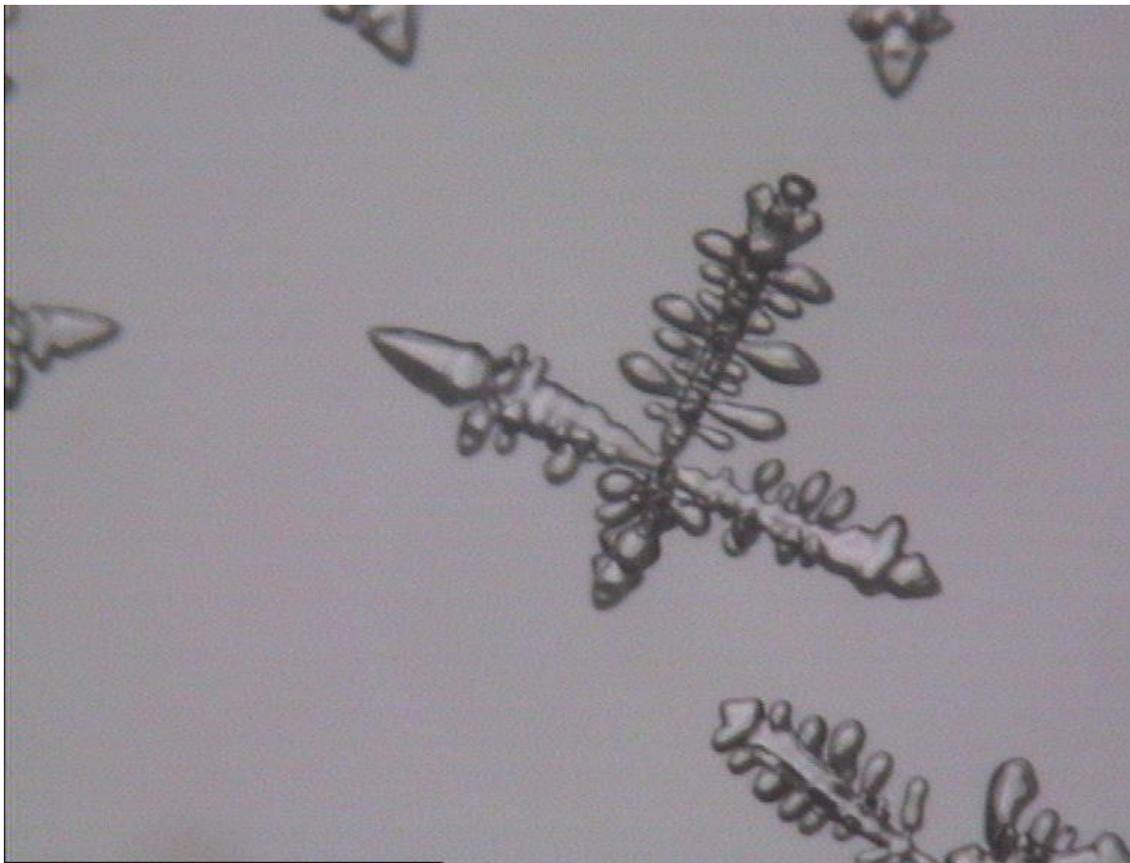


Fig. 1.1. Growth pattern of NH_4Br dendrite crystal

Even though modeling and numerical simulations of dendritic crystal growth have shown an increase in scientific research interest, there are very few articles that point towards control of dendritic solidification processes. Most work done towards control has been done in the framework of non-local models, using a simplified geometry that cannot describe dendritic growth – the classical Stefan

problem. Another approach to control the shape of the crystal involves solving an inverse design problem, in the same framework of non-local models (Zabaras et. al. 1992).

1.3. Wild fire propagation

This section contains a brief description of the fundamental physical processes that are involved in the evolution of a typical (naturally occurring) fire together with a description of related mathematical models and fire management approaches. Wild land fires are the effect of released heat as a result of the thermochemical decomposition of biofuels, and the transport of the produced energy to the surrounding spatial locations which contain unburnt fuel. The transported heat causes a sustained reaction and consequently the ignition of the surrounding fuel occurs, driving the fire growth.

The main phases of a combustion process can be divided in preheating (Hottel et. al, 1971), ignition, flaming, smoldering and glowing combustion (Williams, 1977); each phase represents a different physical phenomenon and implicates different chemical reactions. Although many of the ignition factors can be controlled (decreased human factor influence on the environment), natural causes, such as lightning strikes are not easy to predict. In recent years, fuel ignition due to lightning strikes has received more attention. Reports show that in Canada alone, 45% of all wild land fires are caused mainly by lightning strikes and the damage done represents 81% of the total areas burned (Environment Canada, 2013). Studies have shown that lightning fires can affect the lightning environment through smoke, in some cases inverting the charge structures of cloud-ground lightning strikes.

The two main forms of combustion are flames and smoldering, which represent the means of heat transfer that drive the fire front evolution. Combustion of a material is generally a fuel oxidation process, which generates as output thermal energy, accompanied by various products of complete combustion (carbon dioxide, water) and incomplete combustion (carbon monoxide, soot etc.). This stage of the fire evolution is particularly difficult to model, due to the complex nature of the biofuel and the environment conditions in which the combustion actually occurs (Ward, 2001, Sullivan, 2007a).

Fire front movement after combustion is associated to another class of process modelling, which brings together the various processes (chemical, physical) that describe the fire, and relates them to the environmental factors (wind speed, terrain elevation etc.) to describe the time evolution of the fire front. The rate of spread describes how fast the fire front is burning through unburnt biofuels. Consequently, the rate of spread (or fire front velocity) is generally given by the heat flux that is generated from the active combustion of existing fuel cells in report to the heat that is required to ignite the surrounding fuel (Webber, 2001).

The development of models for fire spread goes back as early as the 1970', when a large number of

papers were published on this matter (Bruce et. al, 1961, Rothermel et. al, 1966, Van Wagner, 1967, Anderson, 1968, 1969, Emmons and Shen, 1971, Rothermel, 1972, Albini, 1976, Rothermel, 1983), a trend that has continued to present times (Rothermel 1991, Larini et. al,1998, Pastor, 2003, Balbi et. al, 2009, Mallet et. al, 2009, Stepanov et al., 2012, Lautenberger, 2013). This variety in modelling techniques and approaches is mainly due to the interactions between the extremely large number of factors that influence the fire. Several types of models categories can be mentioned, depending on the modelling technique used (Sullivan, 2007a and 2007b):

- Physical models
- Quasi-physical models
- Empirical models
- Quasi-empirical models

The fuel type is crucial in analyzing fires, mainly due to the fact that a first categorization is made according to the location of the most abundant and more likely fuel to burn – ground fuels, surface and crown fuels. The first category includes organic matter that is decomposing or dead, plant roots, duff etc. The immediate layer next to the ground fuels is the surface fuel layer, which is composed of fallen foliage, tree bark, twigs, branches, vines, grass, shrub etc. The top fuel layer is the crown layer, which consists of the mid-upper part of the fully-grown trees, where the canopy is located. Each layer has its unique physical properties, and contributes to a certain degree to the development of a fire, depending on the amount of heat produced and transferred. To facilitate the analysis of fire evolution, development of methodic classification and mathematical models of behavior for a wide range of fuel types was addressed (Scott and Burgan, 2005).

The moisture content in these fuels may act in helping or delaying the fire ignition and propagation, and consequently production of smoke (Nelson, 2001). Humidity is mainly affected by the amount of water fall in a specific region, making the local weather climate an important factor for wildfires. The humidity percentage is influenced by the amount of solar radiation, the atmospheric pressure, wind speed, oceanic currents, terrain etc. Thus, another factor that needs to be taken into account is the coupling ensemble of meteorology and climatology conditions and local terrain topography with wildfire development and spread (Flannigan and Wotton, 2001, Coen et. al, 2013).

To minimize the extent to which fires cause damage to both populated environments and natural ecosystems, fire management, control and suppression strategies have been developed. These techniques are managed in general by specialized organizations, which assess and plan the proper action to be taken. The effectiveness of these organizations depends on factors as available technology and resources available for fire suppression, the expertise and knowledge of people within the organization and external natural influences (i.e. bad weather, low visibility, strong winds etc.). Suppression strategies vary

depending on the size and strength of the fire, the terrain topography, fuel available and weather conditions. These approaches act by affecting the main factors that drive fire growth:

- heat reduction – by using water, fire-fighting foam and gels
- limit the amount of available oxygen (combustion inhibition)
- limit the amount of biofuel available – by organized burning various sites situated ahead of the fire

The main suppression strategies can be divided in the following categories:

- Direct control – is used when the fire is detected in its early stages or in regions of large, growing fires that can be rapidly and safely tackled. This method usually leads to relatively fast and complete extinguishment.
- Perimeter control – implies that the fire evolution is fully or partially suppressed in such a way that the regions of fire front are contained inside a specified safety region. This approach is the most common approach for surface fires.
- Prescription control (controlled burning) – is a particular suppression technique in which the fire ignition may be done deliberately. This approach is planned in time slots when forest fires do not naturally occur, and is used as a tool to regulate the possible fuel accumulation and to stimulate the growth of new vegetation.

The typical life cycle of a forest fire can be described in terms of time delay intervals associated to key events from ignition to extinction, which is illustrated in Fig 1.2. (Martell, 2001). By combining these four stages, a simple mathematical model that doesn't take into account the spatial pattern of the evolving fire can be derived as (Parks, 1964):

$$\frac{du}{dt} = Gr_D + at \quad (1)$$

where u represents the total burned area, u 's time derivative equals the fire growth rate, t is the elapsed time since the fire detection event (t_D), Gr_D is the actual growth rate at the detection time and a is the acceleration of fire growth. The initial condition of (1) is $u(0) = u_D$ and represents the size of the area burned at detection time t_D .

The fire size u_A when the actual suppression action begins (t_A) is given by

$$u_A = u(\tau_A) = u_D + Gr_D\tau_A + \frac{1}{2}a\tau_A^2 \quad (2)$$

$$\tau_A = t_A - t_D$$

which will continue to grow at a rate of

$$Gr_A = Gr_D + a\tau_A \quad (3)$$

When the suppression resource is present, Parks assumed the growth rate to be decelerated by a linear

function factor that is dependent on the efficiency of the suppression e_f and the time interval on which it has been active :

$$\frac{du}{dt} = Gr_D + at - e_f(t - \tau_A) = Gr_A - (e_f - a)(t - \tau_A) , \quad t \geq \tau_A \quad (4)$$

The fire evolution then will be fully contained (i.e. growth no longer occurs and $\frac{du}{dt} = 0$) when

$$t - \tau_A = \frac{Gr_A}{(e_f - a)} = \tau_c \quad (5)$$

Although the modelling approach derived by Parks does not provide information about the front shape and does not include any meteorological or topographical aspects of the environment it offers an explicit relationship between the burned area covered by fire evolution and the effectiveness of the suppression method, being one of the first approaches to containment strategies.

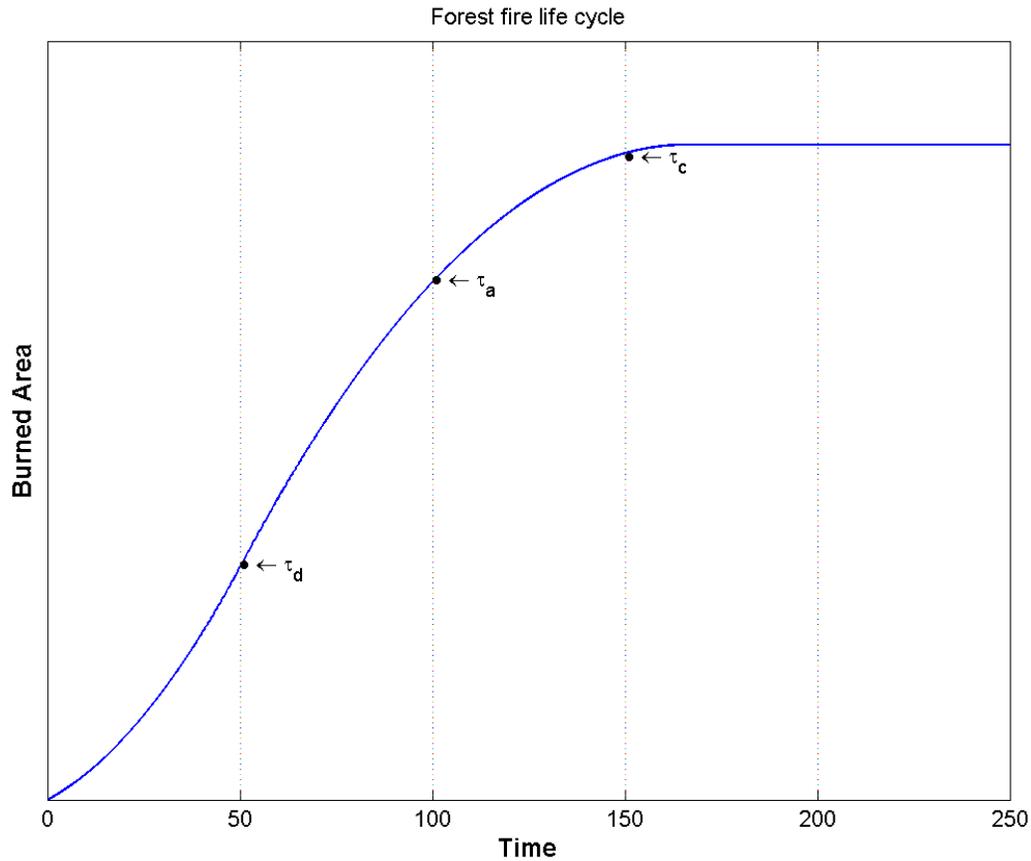


Fig. 1.2. Forest fire life cycle: $0 - t_{ignition}$, $[0 + \Delta t, 50] - \Delta t_{detection}$,
 $[50 + \Delta t, 100] - \Delta t_{arrival}$, $[100 + \Delta t, 150] - \Delta t_{control}$,
 $[150 + \Delta t, 200] - \Delta t_{under\ control}$, $[200 + \Delta t, 250] - \Delta t_{out}$

1.4. Aims and objectives

This thesis aims to develop control strategies for systems with complex spatio-temporal dynamics, focusing on interface evolution problems linked to dendritic crystal growth and wind-aided fire spread. The control laws are derived based on a level set formulation of the geometrical model of the evolving interface.

The main objectives of this work can be summarized as follows:

- Extend the geometrical model of interface evolution for dendritic crystal growth to include effects of undercooling (temperature dependence) in the pattern formation process
- Develop, implement and evaluate through numerical simulation new techniques to control the evolution of the boundary of the dendritic crystals
- Validate the novel methodology using both synthetic data and experimental data, gathered from time lapse observation of dendritic crystal growth
- Advance current wind-aided fire evolution models to include interactions with terrain topography and fuel loading estimation to improve the fire front evolution prediction over a general terrain under known meteorological conditions
- Develop efficient control strategies for the purpose of containment and suppression of wild land fires using a geometrical representation of the fire front interface dynamics.

The control of spatio-temporal systems has numerous practical applications, but poses significant theoretical and implementation challenges. The results of this work will help advance our understanding of these systems as well as improve our ability to control them.

1.5. Overview of the thesis

The work undertaken in this thesis is organized into eight chapters, which provide a comprehensive background material needed for a control strategy definition together with independent chapters that contain the original research contributions.

Chapter 2 makes an introduction to modelling approaches used for capturing spatio-temporal dynamics. Therefore, a comparison of several mathematical concepts is presented, with focus on the main dynamical models used throughout literature. This results in the selection of the most appropriate method to describe the two spatio-temporal systems of interest, dendritic growth and wildfire spread, in the form of partial differential equations. The Chapter provides a brief overview of geometrical models of interface evolution, which in this work are used to describe the dynamics of the two spatio-temporal

systems considered. In particular, the curvature driven evolution of a simple closed curve described by a partial differential evolution equation is studied in more detail.

Chapter 3 provides a thorough review of local and non-local interface evolution models used to describe dendritic crystal growth. In the scope of local evolution models, geometric laws of evolution that are based on curvature flows are presented. Non-local models such as the modified two-phase Stefan problem are considered. Although growth models described by PDE's are the main focus of the current work, coupled dynamical systems are shown to be able to capture general characteristics of dendritic crystal growth.

The Chapter also provides an overview of the most used techniques for tracking the moving boundary. The method adopted in this work is the implicit level set method as the preferred approach. The mathematical formulation of the approach is provided and illustrated through several numerical examples.

Chapter 4 introduces the framework for modeling the spatio-temporal dynamics of propagating wildfires, where the rate of spread is determined by the combined effects of atmospheric, topographic and terrain dependent parameters. Semi-empirical approaches are chosen as the representative mathematical formulation, and fire interface dynamics are derived in a level set formulation. Several aspects of numerical implementation and fire front prediction are studied in a two dimensional space.

Chapter 5 introduces a new strategy to control crystal growth, i.e. impose a desired shape, based on the level set formulation of the geometrical model of the interface evolution.

A new methodology to derive a distributed closed loop control strategy is presented, starting from an arbitrary initial interface and the desired reference. By representing the interfaces using a level set formulation, the main components needed to define the time evolution of a controlled variable are derived. The closed loop control method ensures the convergence of any initial simple and non-intersecting curve to the desired reference.

The control method is formulated for interfaces evolving in both two and three-dimensions, providing the basis for practical implementation. The numerical implementation of the proposed algorithm is presented in detail for 2D and 3D problems.

To validate the new approach, several scenarios are studied, in both two and three dimensional cases. The novel technique is first used to control the nonlinear dynamics with artificially generated curve (surface) sets. Several key performance indicators are defined to measure the performance of the closed loop controller, with excellent results.

To illustrate the robustness of the algorithm, experimentally observed data of a dendritic growth process is processed and the complex spatio-temporal pattern is then extracted and used as a reference. As a result, it was seen that the results of the proposed method allow arbitrary initial seeds to evolve into

highly complex shapes.

Chapter 6 proposes a new fire suppression method for wild land fires, based on both static and dynamic data that can be measured or is readily available for the observed region where an ignition event has been detected. The new control method blends a-priori, slow varying, information of a region of interest with the rapidly changing measurements of atmospheric conditions, providing an accurate estimate of the system dynamics. The method is based on containing the evolving fire within a certain area by gradually reducing the temperature at the boundary between the burnt and unburnt biofuel layer. To this aspect, the suppression method is a parametric control problem, which idealizes the resources available.

A nonlinear semi-empirical geometrical model of evolution is used to predict the growth of the fire interface and, based on key influencing parameters and the region defined by the containment zone, a specific control action is taken.

The new suppression strategy is shown to be robust against several real-life scenarios. To this scope, the effects of artificially ignited fire fronts is analyzed under varying atmospheric and terrain conditions, together with their un-controlled counterparts, clearly showing the advantages of introducing the controlled dynamics.

To further extend the method, satellite observations in the form of terrain imaging and digital terrain data are processed and quantitative information related to biofuel loading and terrain topography are estimated and incorporated into the evolution model, providing real-time accurate data for the prediction of a potential wildfire.

Chapter 7 introduces a novel optimization strategy for controlling the evolution of wildland fires, based on a geometrical representation of the evolving interface. The method is aimed at providing a solution to a resource allocation problem in firefighting scenarios. Using known information about an area of interest where a fire is detected, a cost function, which incorporates the fire dynamics, is used to determine the spatial location where the available (limited) amount of firefighting resource is most effective, given a specific control target.

The optimized boundary control strategy provides several tuning parameters that deliver a direct link between the evolving fire interface and the fire-fighting process, enabling the analysis of a vast number of scenarios. As with the approach presented in Chapter 6, the method can incorporate real-world data and containment sectors, thus being able to provide accurate predictions of the evolving dynamics.

Chapter 8 provides a summary of the work undertaken and highlights future directions of research.

As a direct consequence of the work undertaken in this thesis, the following conference papers have been published to date. Two journal papers are in preparation.

Bouaru A & Coca D (2014) Control of spatio-temporal pattern formation governed by geometrical models of interface evolution. *2014 European Control Conference, ECC 2014* (pp 1993-1998)

Bouaru A & Coca D (2014) Modelling and control of wildfires using geometrical models of interface evolution. *2014 Mathematics and Engineering in Marine and Earth Problems, MEME 2014*

2. Mathematical representations of spatio-temporal systems

2.1. Introduction

Numerous natural phenomena such as dendritic crystal growth and wildfire propagation, exhibit highly complex patterns that fall in the category of spatio-temporal systems. To better understand the underlying dynamics in such systems an accurate mathematical representation of the spatial interactions and time dynamics in such systems, which enables numerical simulation and analysis, prediction and practical implementation of control strategies is crucial.

This chapter provides an overview of different mathematical representations of spatio-temporal models. The most important classes of models that are used in practice to describe spatio-temporal dynamical systems are cellular automata (CA) models, coupled map lattice (CML) models and partial differential equation (PDE) models. This chapter succinctly reviews these approaches that are used to model spatio-temporal dynamics with focus on PDE models that will form the basis for the work. The Chapter is organized as follows. Section 2.2 provides a description of CA models present in literature. The third part, Section 2.3, provides details related to the structure and properties of CML models. In this thesis spatio-temporal systems will be represented using PDEs and therefore Section 2.4 is devoted to a brief theoretical overview of PDE models and interface evolution equations.

2.2. Cellular automata models

Cellular automata (CA) models represent an important tool in the study of spatio-temporal systems and were first introduced in (Neumann, 1951, 1966). A CA model is a discrete system that evolves in both discrete space and discrete time steps over a predefined spatial geometry. The main structure of a CA model is defined by a discrete grid structure, cell states, a neighborhood and a local transition rule.

The discrete lattice (grid of cells) on which the model is defined, can be n -dimensional in size. The cells, which represent the atomic building blocks of the CA, can be chosen to have various shapes (triangle, square, hexagonal), usually under the assumption of homogeneity (Fig. 2.1). The selection between shapes is usually related to the nature of the system of interest and the type of evolution law that is selected. The most common lattice structure is the square lattice, mainly due to its relatively simple numerical implementation, but hexagonal and triangular lattice structures have been used in literature (Morita et. al, 1995, Pagnutti et. al, 2005, Wuensche et. al, 2006).

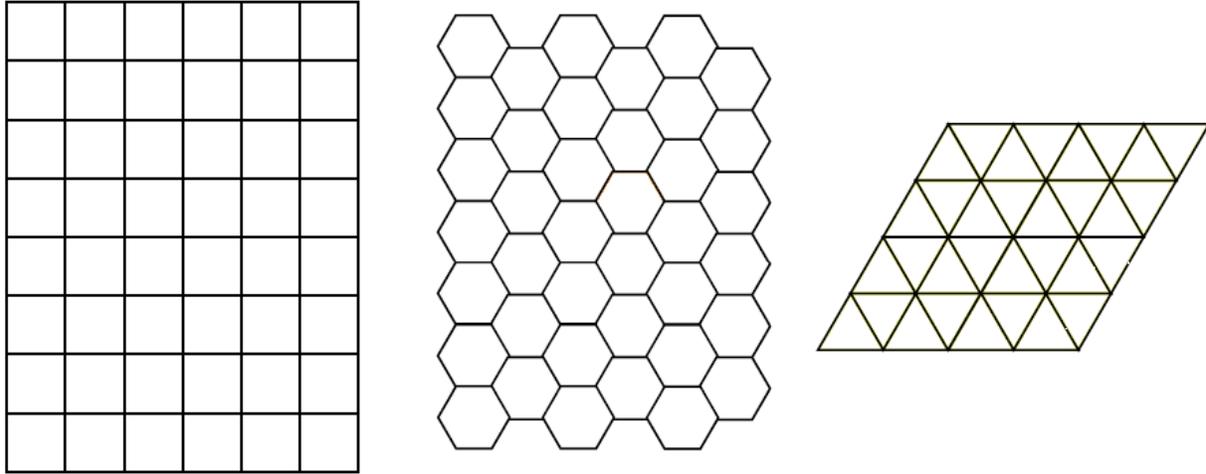


Fig 2.1. CA lattice diversity (two dimensional CA) – square, hexagonal and triangular lattice structures

The cell state, which is the value assigned to each cell, belongs to a set of allowed states, i.e. $s \in S = \{s_0, s_1, \dots, s_k\}$, with finite cardinality ($k < \infty$), where s_k can only take a finite number of discrete values. Binary CA are the most common, where the allowed states are $\{0,1\}$. In the simplest CA models, the state of a cell at time step t depends on the states of its neighboring cells at time step $t - 1$, which in turn depend on their corresponding neighborhoods at time step $t - 2$ etc.

The neighborhood N of a cell, represents the set of surrounding positions that influence the new value of the current cell. The neighborhood can be defined on different spatial locations and temporal steps, together with a variety of stencil size, allowing a large diversity in terms of evolution laws. Consider a one dimensional CA model, defined over a square lattice structure, where the spatial position of a cell is given by an index i at a discrete time step t and is denoted by $x_i(t)$. The neighborhood can then be a typical radial structure (Fig 2.2) or a combination of various spatial or temporal positions within the one dimensional CA (Wolfram, 1983, 1994). Considering the cell of interest is green, for the structures shown in Fig. 2.2, the neighborhoods can be defined as:

$$R = 1 : N_{x_i(t)} = \{x_{i-1}(t-1), x_{i+1}(t-1)\}$$

$$R = 2 : N_{x_i(t)} = \{x_{i-2}(t-1), x_{i-1}(t-1), x_{i+1}(t-1), x_{i+2}(t-1)\}$$

where here all the neighborhoods contain cells from the same temporal period.

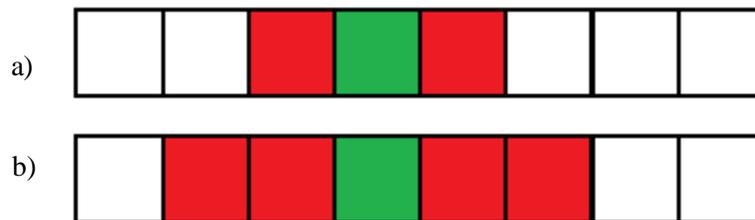
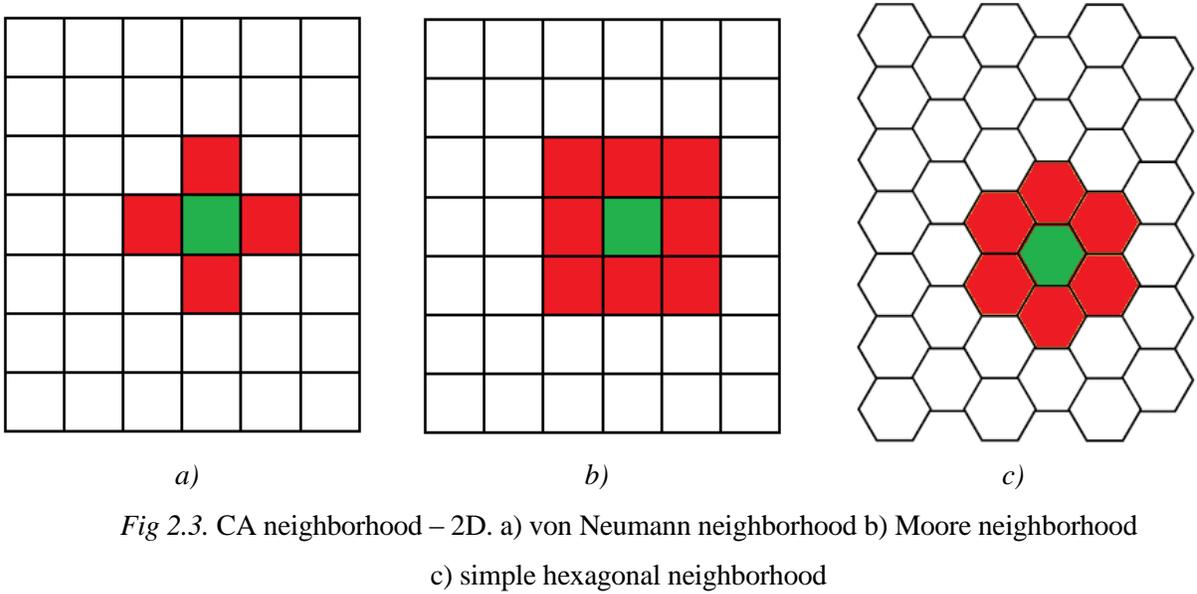


Fig 2.2. CA neighborhood – 1D. a) radial, $R = 1$ b) radial, $R = 2$



Given a two dimensional CA, the spatial location of an individual cell is defined by the set off indices (i, j) at a discrete time step t and is denoted by $x_{i,j}(t)$. In the two dimensional space, the variety of neighborhoods that can be considered is significantly larger than the previous case, given the higher number of spatial and temporal location combinations available. Among the most popular neighborhood structures are the von Neumann, Moore and hexagonal neighborhoods (Fig. 2.3), but more complex structures can be considered (Billings, 2013). As in the one dimensional case, considering the cell of interest is green, for the structures shown in Fig. 2.3, the neighborhoods can be defined as:

- von Neumann

$$N_{x_{i,j}(t)} = \{x_{i,j-1}(t-1), x_{i-1,j}(t-1), x_{i-1,j+1}(t-1), x_{i,j+1}(t-1)\}$$

- Moore

$$N_{x_{i,j}(t)} = \{x_{i+ik,j+jk}(t-1) \mid \|(ik, jk)\| \leq 1\}$$

The transition function is the governing law that dictates the evolution process of the CA, i.e. how an individual cell's state changes depending on its current value and the values of its current neighbors. At each time step, each cell is updated to time $t + 1$ according to a function $f: S^n \rightarrow S$ that maps the selected neighborhood evaluated at time step t to S . The transition rule can take various equivalent forms, depending on the nature of the system modelled by CA. Therefore, the local rule can be in the form of a truth table (Ilachinski, 2001), a Boolean function (Billings and Yang, 2003), probabilistic transition rule (Billings, 2013) or a polynomial function (Yang and Billings, 2000).

CA models are among the most basic spatio-temporal systems, and poses relatively simple structures and construction rules. Despite their simplicity CAs can be used to describe the behavior of numerous phenomena that exhibit pattern formation and have been widely used to model and identify the structure

of spatio-temporal models (Zhu et al., 2001, Yeh et al., 2002, Karafyllidis and Thanailakis, 1997, Zhao et al., 2009, Guo et. al., 2008, Zhao et. al., 2012).

2.3. Coupled map lattice models

The coupled map lattice model was first introduced in the 1980s (Kaneko, 1985, 1989) as a spatial counterpart of lumped discrete dynamical systems, and represents a class of discrete models that can be used to represent various phenomena described by spatio-temporal dynamics (Billings and Coca, 2002). CML models have gained increasing popularity in modelling spatio-temporal phenomena in recent years.

Coupled map lattice models (CML) possess several similarities with CA models, having the same discrete time and space characteristics, but unlike CA, the state space is continuous (Pan and Billings, 2008). In CA the information stored in the system states is discrete in nature, which limits the capability of CA to produce a high qualitative representation of a continuous system. Since CML models have the property of continuous state variables, they are able to better capture the system dynamics, since less information is lost through the discretization process.

Let L^d represents a d -dimensional computational lattice, such that $k = (k_1, k, \dots, k_d) \in \mathbb{Z}^d$ represent the set of coordinates over the map L^d . The general form for an input-output CML defined over L^d is then given by a combination of a local interaction component $f_l(\cdot)$ and a spatial coupling component $f_c(\cdot)$ (Coca and Billings, 2001):

$$z_k(t) = f_l(\mathbf{q}^{-n_z} z_k(t), \mathbf{q}^{-n_u} u_k(t)) + f_c(\mathbf{q}^{-n_z} z_k(t), \mathbf{q}^{-n_u} u_k(t), \mathbf{s}^{m_z} \mathbf{q}^{-n_z} z_k(t), \mathbf{s}^{m_u} \mathbf{q}^{-n_u} u_k(t)) \quad (2.1)$$

$$z_k(t) = f(\mathbf{q}^{-n_z} z_k(t), \mathbf{q}^{-n_u} u_k(t), \mathbf{s}^{m_z} \mathbf{q}^{-n_z} z_k(t), \mathbf{s}^{m_u} \mathbf{q}^{-n_u} u_k(t))$$

where \mathbf{q}^{-n} represents a temporal back-shifting operator

$$\mathbf{q}^{-n} = (q^{-1}, q^{-2}, \dots, q^{-n}) \quad (2.2)$$

So that

$$\mathbf{q}^{-n_z} z_k(t) = (z_k(t-1), z_k(t-2), \dots, z_k(t-n_z)) \quad (2.3)$$

$$\mathbf{q}^{-n_u} u_k(t) = (u_k(t-1), u_k(t-2), \dots, u_k(t-n_u)) \quad (2.4)$$

with $\{n_u, n_z\}$ representing the maximum temporal lags associated with the input u and the output z . In (2.1), \mathbf{s}^m is a multi-valued spatial translational operator with

$$\mathbf{s}^{m_z} = (s^{p_z^1}, s^{-p_z^1}, \dots, s^{p_z^{m_z}}, s^{-p_z^{m_z}}) \quad (2.5)$$

$$\mathbf{s}^{m_u} = (s^{p_u^1}, s^{-p_u^1}, \dots, s^{p_u^{m_u}}, s^{-p_u^{m_u}}) \quad (2.6)$$

Such that

$$\mathbf{s}^{m_z} z_k = (z_{k+p_z^1}, z_{k-p_z^1}, \dots, z_{k+p_z^{m_z}}, z_{k-p_z^{m_z}}) \quad (2.7)$$

$$s^{m_u} u_k = \left(u_{k+p_u^1}, u_{k-p_u^1}, \dots, u_{k+p_u^{m_u}}, u_{k-p_u^{m_u}} \right) \quad (2.8)$$

where the parameters $\{p_u^j, p_z^j\} \in \mathbb{Z}^d$ denote the spatial translation multi-indices with $\{m_u, m_z\}$ being the maximum allowed spatial radius attached to input u and the output z respectively.

In equation (2.1), $f: Z^{n_z+m_z} \times U^{n_u+m_u} \rightarrow Z$ is a differentiable map that depends on the local input $u_k \in U$ and state $z_k \in Z$, evaluated at node k , and the variables at neighboring sites. Usually the nearest neighbor map is the most commonly used (Bunimovich, 1995, Kaneko, 1989, Richter, 2008), but other types of neighborhood can be adopted, and the result considerably changes the spatio-temporal patterns obtained.

CML have been successfully used to model complex spatio-temporal processes such as the dynamics in phase transitions in boiling (Yanagita, 1992), crystal growth (Reiter, 2005, Kessler et. al, 1990) or formation and dynamics of clouds (Yanagita and Kanebo, 1997).

Although CML models are able to represent spatio-temporal information with higher fidelity than CA models, their main limitation is that they are discrete in space and time and thus cannot describe in full spatio-temporal processes that are continuous in space and time.

2.4. Partial differential equations

Partial differential equations (PDEs) are continuous time-space models that are widely used to accurately model an extensive range of natural phenomena (crystal growth, wildfire spread, fluid dynamics, heat transfer, acoustics etc.). PDEs represent a very powerful tool in modeling spatio-temporal systems.

Consider an open subset $\Omega \subset \mathbb{R}^n$; the general form of a PDE of order $k \geq 1$ is given by:

$$L(D^k u(\mathbf{x}), D^{k-1} u(\mathbf{x}), \dots, Du(\mathbf{x}), u(\mathbf{x}), \mathbf{x}) = 0, (\mathbf{x} \in \Omega) \quad (2.9)$$

where $L: \mathbb{R}^{n^k} \times \mathbb{R}^{n^{k-1}} \times \dots \times \mathbb{R}^n \times \Omega \rightarrow \mathbb{R}$ is known and $u: \Omega \rightarrow \mathbb{R}$ is considered the unknown. The partial derivatives $D^j u(\mathbf{x}), j \geq 1$ are given by

$$D^j u(\mathbf{x}) = \frac{\partial^{j_1} u}{\partial x_1^{j_1}} \frac{\partial^{j_2} u}{\partial x_2^{j_2}} \dots \frac{\partial^{j_N} u}{\partial x_N^{j_N}} \quad (2.10)$$

with $Du = D_x u = (u_{x_1}, \dots, u_{x_n})$ denotes the gradient of u with respect to the variable $\mathbf{x} = (x_1, \dots, x_n)$, $j_i \in \mathbb{N}, i = \overline{1, N}$ and $j_1 + j_2 + \dots + j_N = j$. In spatio-temporal systems one of the independent variables represents the time variable ($x_1 = t$) and the others are spatial coordinates.

Let $x \in \Omega \subset \mathbb{R}$ be the spatial variable, $t \in \mathbf{T} \subset \mathbb{R}^+$ denotes time and $u(x, t): \Omega \times \mathbf{T} \rightarrow \mathbb{R}$ satisfies the following PDE:

$$L\left(x, t, u, \frac{\partial u}{\partial x}, \frac{\partial u}{\partial t}, \frac{\partial^2 u}{\partial x^2}, \frac{\partial^2 u}{\partial t^2}, \frac{\partial^2 u}{\partial x \partial t}, \dots\right) = 0 \quad (2.11)$$

The PDE in (2.11) is said to be linear if (2.9) can be written in the form:

$$\sum_{|\alpha| \leq k} a_\alpha(x) D^\alpha u = f(x), x \in \Omega, \alpha = (\alpha_1, \alpha_2, \alpha_3, \dots, \alpha_N) \quad (2.12)$$

for given functions $a_\alpha(x)$, where $|\alpha| = \alpha_1 + \alpha_2 + \alpha_3 + \dots + \alpha_N$ and $f(\cdot)$ is a known function. Examples of linear spatio-temporal PDE's include the heat equation, the wave equation or the linear transport equation (Evans, 1998). If $L(\cdot)$ is of the form

$$\sum_{|\alpha|=k} a_\alpha(D^{k-1}u, \dots, Du, u, x) D^\alpha u + a_0(D^{k-1}u, \dots, Du, u, x) = 0 \quad (2.12a)$$

i.e. the coefficients of the highest order terms do not depend on $D^k u$, the PDE is categorized as quasilinear. Examples of such systems include the p -Laplacian or the minimal surface equation (Evans, 1982). If the highest order derivatives of u appear nonlinearly in (2.11), the PDE is considered fully nonlinear. Popular examples of nonlinear equations describing a space-time evolution are the Hamilton-Jacobi equation and the reaction-diffusion equation (Osher and Fedkiw, 2003).

PDEs are defined on bounded domains which often include several conditions to be specified within the domain. One necessary condition is the initial condition for the PDE that describes the state of the system at time $t = 0$, i.e.

$$u(\mathbf{x}, 0) = u_0(\mathbf{x}), \forall \mathbf{x} \in \Omega \quad (2.13)$$

Depending on the problem under analysis additional conditions need to be imposed on u at $\mathbf{x} \in \partial\Omega$ if Ω is bounded (physical constraints), as the PDE evolves over a time period $t = (0, T]$, to specify a unique solution, given a known function $\psi(\mathbf{x}, t): \Omega \times \mathbf{T} \rightarrow \mathbb{R}$:

- Dirichlet condition: $u(\mathbf{x}, t) = \psi(\mathbf{x}, t), \forall \mathbf{x} \in \partial\Omega, t \geq 0$,
- Neumann condition: $\frac{\partial u(\mathbf{x}, t)}{\partial n} = \psi(\mathbf{x}, t), \forall \mathbf{x} \in \partial\Omega, t \geq 0$
- Mixed boundary conditions, which require $u(\mathbf{x}, t)$ to take certain values on a subset $\Omega_1 \subset \partial\Omega$ and the derivative of $u(\cdot)$ in the normal direction to take certain values on $\Omega_2 \subset \partial\Omega$, with Ω_1, Ω_2 two disjoint parts such that $\partial\Omega = \Omega_1 \cup \Omega_2$,
- Robin boundary condition which requires imposing a linear combination of Dirichlet and Neumann conditions on the boundary $\partial\Omega$, i.e.

$$a \cdot u(\mathbf{x}, t) + b \cdot \frac{\partial u(\mathbf{x}, t)}{\partial n} = \psi(\mathbf{x}, t), \forall \mathbf{x} \in \partial\Omega, t \geq 0 \quad (2.14)$$

where a, b may be scalars, vectors or functions.

In the case of crystal growth, the phase change is mainly driven by the temperature distribution in the considered domain where the growth is observed. Let $u(x, y, z, t): \Omega \subset \mathbb{R}^3 \times \mathbf{T} \rightarrow \mathbb{R}$ be a three

dimensional temperature field defined over a time period $t = [0, T]$, $t \in \mathbf{T} \subset \mathbb{R}^+$. Fourier's law expresses the relation between thermal rate of flow \dot{q} and the temperature gradient ∇u across a unit surface S :

$$\dot{q} = \frac{\Delta q}{dt} \quad (2.15)$$

$$\dot{q} = -\mathbf{k}\nabla u \quad (2.16)$$

where Δq is the amount of heat that passes through S in time dt and \mathbf{k} is the thermal conductivity. Considering the components of the flow in three spatial dimensions, the flow can be expressed as:

$$\dot{q} = q_x \vec{i} + q_y \vec{j} + q_z \vec{k} \quad (2.17)$$

$$q_x = -k_x \frac{\partial u}{\partial x}, q_y = -k_y \frac{\partial u}{\partial y}, q_z = -k_z \frac{\partial u}{\partial z} \quad (2.18)$$

where k_x, k_y, k_z are the thermal conductivities in each spatial dimension. Considering an element with volume dv , which is part of a material undergoing conductive heat transfer (Fig 2.4) and assuming that within the volume element there are a series of uniformly distributed heat sources that generate an equal and constant amount of heat q_v throughout the volume, the material is homogenous and isotropic ($k_x = k_y = k_z = k$) and physical deformations due to temperature changes are neglected, the expression for the heat equation is obtained as (Evans,1998, Arfken, 2013):

$$\frac{\partial u}{\partial t} = \frac{k}{\rho \cdot c_p} \cdot \nabla^2 u + \frac{\dot{q}_v}{\rho \cdot c_p} \quad (2.19)$$

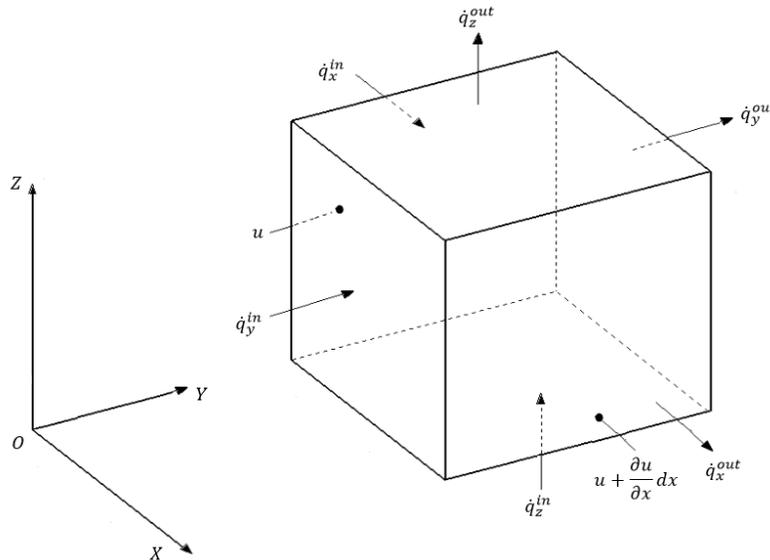


Fig 2.4. Volume element $dv = dx \cdot dy \cdot dz$ undergoing heat transfer

where ρ is the material density and c_p is the specific heat capacity. The term $\frac{k}{\rho \cdot c_p}$ is a measure of the thermal inertia of the material, and characterizes the ability of the material to store thermal energy. Equation (2.19) expresses that the speed at which the temperature changes within a material, under the action of a

conductive heat flow and the influence of internal heat sources depends on the temperature gradient and the thermal inertia of the material.

Depending on the nature of the system observed there are several variations of the heat equation, which lead other well-known PDEs:

- Poisson's equation, which is obtained by considering steady state heat transfer, with internal heat sources

$$\left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2} \right) u = f(x, y, z) \quad (2.20)$$

and has a wide-ranging use in engineering and theoretical physics.

- Fourier's equation, which describes the transient heat transfer in the absence of internal heat sources

$$\rho \cdot c_p \cdot \frac{\partial u}{\partial t} = k \cdot \nabla^2 u \quad (2.21)$$

- Laplace's equation, which describes any heat conduction problem where the temperature distribution is independent of the time variable

$$\left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2} \right) u = 0 \quad (2.22)$$

Although one of the main uses of equation (2.19) lies in the field of thermal analysis and simulation, it is often used in differential geometry, in particular mathematical modeling of geometric flows in Riemannian manifolds, with the parameter of interest chosen to be the curvature (Eisenhart, 1947, Do Carmo, 1976, Brakke, 2015).

Let $\gamma: \Omega \subset \mathbb{R}^2 \times [0, T] \rightarrow \mathbb{R}^2$ be a smooth convex closed plane curve in \mathbb{R}^2 . Assuming the curve γ is parameterized by arc length s then the curvature is defined as:

$$k(s, t) = \frac{\partial^2}{\partial s^2} \gamma(s, t) \quad (2.23)$$

If $\gamma(s, t)$ is evolving under its mean curvature the following PDE is satisfied:

$$\frac{\partial}{\partial t} \gamma(s, t) = -k(s, t) \cdot N \quad (2.24)$$

where N represents the outward unit normal to the curve. The curvature flow is one of the most studied types of curvature driven evolution and has found applications in numerous areas (crystal growth, flame propagation, image processing etc.). With respect to curvature, there can be defined a large variety of geometric flows, with two main categories being intrinsic flows such as Ricci flow (Ivancevic, 2011, Morgan and Tian, 2007), Calabi flow (Chen and He, 2008), Yamabe flow (Ye, 1994); or extrinsic flows such as the mean curvature flow or the surface tension flow.

The evolution law (2.24) can be written in the form of a quasilinear PDE as:

$$\left(\frac{\partial}{\partial t} - \frac{\partial^2}{\partial s^2}\right)\gamma = 0 \quad (2.25)$$

which reveals the parabolic nature of curvature flow and similarities with the heat equation. The curvature flow (2.25), also called curve shortening flow, describes the relation between the normal velocity of an interface in relation to its local curvature. This type of geometric evolution has a smoothing effect over time, reducing irregularities on the curve, therefore reducing its arc length and area (Gage, 1984, Gage and Hamilton, 1986). If $L(t)$ represents the arc length of $\gamma(s, t)$, in the case of curvature flow, its time variation is given by:

$$\frac{d}{dt}L(t) = - \int_{\gamma(s,t)} k^2 ds \quad (2.25)$$

which implies that the curve evolution is in the direction that reduces the arc length, as noted. To exemplify this let us consider a two dimensional circle C of time varying radius $R(t)$, and initial radius $R(0) = R_0$ evolving under curvature flow. At each point $p \in C$, the curvature is then defined by $k(t) = \frac{1}{R(t)}$. The time evolution of the circle is then given by:

$$\frac{d}{dt}R(t) = - \frac{1}{R(t)} \quad (2.26)$$

By solving the simple curvature driven evolution equation in (2.26), the time variation of the radius $R(t)$ is found to have the form:

$$R(t) = (R_0^2 - 2t)^{\frac{1}{2}} \quad (2.27)$$

which describes the time evolution of the circle, which is continuously shrinking, eventually collapsing to a single point and developing a singularity at time $t = \frac{R_0^2}{2}$.

It is often the case that the dynamics of a continuous spatio-temporal system can be described by a system of partial differential equations. However, for the purpose of system identification, numerical simulations and control, a discretization of the linear (quasilinear, nonlinear) PDE under consideration is needed. Therefore, for such systems, measurements are usually taken in the form of time snapshots at regular time intervals, which can then be used to obtain a discrete-time representation of the system. The most popular way used to discretize PDEs over a domain are finite difference methods, which are used to obtain an approximation of the spatial and temporal derivatives, thus obtaining a discrete time continuous state approximation that can be used in simulations and the design of control laws.

PDEs represent one of the most important classes of models, mainly due to their continuous nature. This special characteristic that distinguishes PDE models from discrete type approaches allows to better capture real-world dynamics of spatio-temporal systems.

2.5. Conclusions

This chapter included a brief introduction to the approaches that are used to model spatio-temporal dynamics, namely cellular automata (CA) models, coupled map lattice (CML) models and partial differential (PDE) models. By doing so, basic concepts that will be frequently referenced throughout the thesis were introduced. In this study the focus is set on representing the dynamics of systems using partial differential equations, mainly due to the fact that PDE models typically provide a clear relation between model parameters and physical behavior of the real-world system, thus creating an intuitive and suitable description of the system under analysis. Coupled map lattice models and CA models are also referenced throughout the next chapters, and several numerical studies are presented to illustrate their ability to capture the temporal dynamics together with the spatial interactions that produce complex evolving patterns found in spatio-temporal systems. Chapter 3 will put this outline in background of mathematical approaches used to model the remarkable non-linear dynamics found in dendritic crystal growth, followed by Chapter 4 which will give an overview of wild fire spatio-temporal models.

3. Models of dendritic crystal growth

3.1. Introduction

The growth of a crystal can be viewed as a moving boundary of a closed curve. Growth models developed are usually differentiated based on the approach used in defining the law of evolution of the moving boundary. Several modeling approaches were developed in literature. These can be categorized in two main types: local models (Eden, 1961, Wettlaufer et. al, 1994, Ben-Jacob et. al, 1984) and non-local models (Kobayashi, 1993, Chen, 2002). This chapter provides a review of crystal growth models together with a detailed numerical analysis of the interface growth models for dendritic crystal growth.

The chapter is organized as follows. Section 3.2 is devoted to a review of dendritic crystal growth interface evolution models, namely local and non-local models of interface evolution, with emphasis given to (local) geometrical evolution laws. The third part, Section 3.3, provides an overview of the most common numerical methods used in interface evolution problems. The level set method is selected as the preferred numerical framework, due to its numerous advantages over other simulation methods (Osher and Fedkiw, 2003). Following this, the model classes presented in Section 3.2 are analyzed numerically in more depth in Section 3.4, which covers an analysis of curvature driven interface evolution models, the modified two-phase Stefan problem and CML dendritic growth models.

3.2. Models of interface evolution

3.2.1. Geometrical models of dendritic growth

Models where growth speed is determined only by local parameters of the solid-liquid interface, such as curvature, or other shape-dependent properties are called geometrical models (Zhao et. al., 2007). This class of models usually doesn't include long-range influences such as change in solute concentration, or heat flow, but have been used successfully to accurately model early stages of dendritic crystal growth (Brower et al, 1984). Let us consider a typical growth process, in which a crystal grows in a liquid solution. The crystal usually grows in all three dimensions, but for simplicity only growth in the $x - y$ plane will be considered.

In two dimensions, the crystal growth process can be viewed as the evolving boundary of a curve (interface), such that the curve separates at any given time the solid surface from the liquid melt.

Let C be a closed curve, and $\tilde{x}: [0, S] \times [0, T) \rightarrow \mathbb{R}^2$ is a continuous function such that $\tilde{x}(s, t) = (x(s, t), y(s, t))$ is a point on the curve, with $s \in [0, S]$ a parameter along the curve (e.g. arc length) and x, y are Cartesian coordinates.

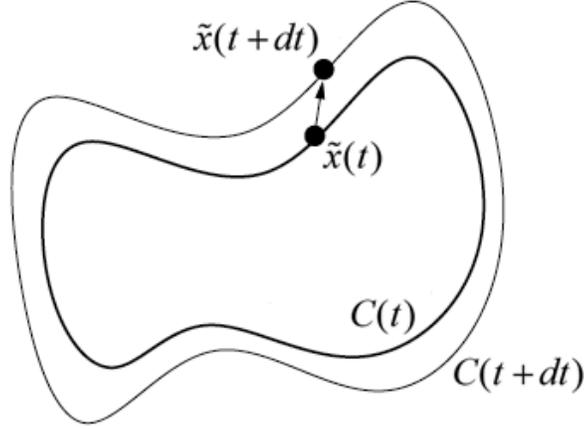


Fig. 3.1. Time variation of contour $C(t)$

Further, it is supposed that the curve C is injective at each time step, i.e. the curve does not intersect itself and does not have missing points, and the periodic boundary condition $\tilde{x}(0, t) = \tilde{x}(S, t)$ holds. As the crystal grows, its boundary undergoes deformations, generating various shapes (Fig. 3.1). The speed of the interface can be written as:

$$\begin{cases} \frac{\partial \tilde{x}}{\partial t} = \alpha_1(s, t) \cdot \vec{N} + \alpha_2(s, t) \cdot \vec{T} \\ \tilde{x}(s, 0) = C_0(s) \end{cases} \quad (3.1)$$

where \vec{N} is the outward curve unit normal, \vec{T} is the tangent vector, s is the curve parameterization, t is the time parameter and α_1 and α_2 are components of some speed function V in the normal and tangential directions. Usually the tangential component of the velocity does not affect the shape of the closed curve so that the last term in the right-hand side of equation (3.1) is equal to 0. The normal speed is typically a curvature dependent function, but it can also include other terms, like orientation angle, θ with respect to an axis. The equation of motion of the interface becomes:

$$\begin{cases} \vec{N} \cdot \frac{\partial \tilde{x}}{\partial t} = \alpha_1(k, \theta) \\ \tilde{x}(s, 0) = C_0(s) \end{cases} \quad (3.2)$$

where k is the curvature. This type of motion is usually referred to as curvature based front evolution. The speed function $\alpha_1(k, \theta)$ can consist of several terms: a polynomial of curvature, with powers up to the n^{th} order, terms proportional to derivatives of the curvature with respect to some curve parameter, such as arc length, and terms related to the curve orientation.

The issue of front propagation in the normal direction under curvature-dependent flow is characteristic in modeling a wide range of physical phenomena that exhibit pattern formation, and among the most studied in literature crystal growth and propagation of flames. A typical curvature speed, given by $\alpha_1 = 1 - \varepsilon k$, where ε is an arbitrary constant, was studied in a series of papers by Sethian (1985, 1989, 1990). This particular form of α_1 is a specific representation of the Gibbs-Thomson energy conservation condition on the boundary of the solid (Almgren and Wang, 2000), in the case of crystal growth. Sethian presented a comparison of numerical methods to tackle the problem of front tracking under different parameterizations of the evolving curve for α_1 . The numerical algorithms presented are based on solving general Hamilton-Jacobi equations (Shu and Osher, 1989, Osher, 1993, Jiang and Peng, 2000), which approximate the movement of the boundary under its curvature.

A first noticeable model used to describe the geometrical evolution of crystal growth was the one developed by (Brower et al., 1983), which was the starting point for many current works. They proposed a general model of the following form:

$$\begin{cases} \vec{N} \cdot \frac{dx}{dt} = \mathbf{F}(k, \nabla k, \dots) \\ \mathbf{F} = F(k) + \gamma \frac{\partial^2 k}{\partial s^2} \end{cases} \quad (3.3)$$

where ∇k represents the covariant derivative on the surface, and $\mathbf{F}(k, \nabla k, \dots)$ is a general function of k and its derivatives. The term $\gamma \frac{\partial^2 k}{\partial s^2}$ in (3.3) has a stabilizing effect on the system, with $\gamma > 0$ representing a measure of surface tension. The function F was chosen to take into account the aspects of physical solidification, such as interface shape and minimum nucleation size:

$$F(k) = k + \alpha k^2 - \beta k^3 \quad (3.4)$$

The equation (3.4) is a polynomial function of curvature, where the coefficients $\alpha > 0$ and $\beta > 0$ are related to the degree of undercooling and the minimum bubble size for nucleation. This model was further developed in (Brower et al, 1984), where kinematics of geometrical evolution in two dimensions is investigated in terms of speed function \mathbf{F} . By adding small perturbations to the initial front, they compute and analyze the effects of the surface tension parameter γ on the system.

In the work of Kessler et. al, (1984, 1985) further improvement of the model is made by adding effects of crystalline anisotropy. The model (3.3) was modified into:

$$\vec{N} \cdot \frac{d\vec{x}}{dt} = \left[F(k) + \gamma \frac{\partial^2 k}{\partial s^2} \right] \cdot [1 + e \cdot \cos(m\theta)] \quad (3.5)$$

The extra term proportional to e was introduced in order to include the effects of repeated side-branching characteristic to dendritic crystal growth. This effect is reflected in enhanced growth in the

direction given by the angle θ , made by the curve normal and the y axis along preferred m fold symmetry. The anisotropy term e was shown to be related to the growth velocity of a dendrite. For a value of e greater than a threshold value $e_c = 0.15$ the growth speed was shown to remain constant, and the amplitude of oscillations that affect the tip velocity drops. This means that for anisotropy strength smaller than the threshold value, the dendritic growth is unstable.

Zhao et. al. (2007) introduced a new method of identifying the growth rule in crystal growth, based on the local geometry of the evolving boundary of the crystal. He proposed a polynomial form of the speed function F of the form:

$$F = a_0 + a_1k + a_2k^2 + \dots + a_nk^n \quad (3.6)$$

where k designates the curvature and n is the order of the polynomial model. The coefficient vector $\{a_0, a_1, \dots, a_n\}$ is estimated using a least squares method using collected data points. The data points are obtained from a sequence of images of a growing NH_4Br dendrite crystal from melt, taken at different time steps. The model was attained using regions of the boundary and not the whole boundary of the crystal, leading to a multi-order polynomial speed function of the form:

$$F = 1.2105 + 129.34k - 957.85k^2 \quad (3.7)$$

The model (3.7), although simpler than the one proposed in (3.3) is able to approximate the real system to some degree and it captures the main properties of the crystal boundary evolution for regions with relatively large curvature growths.

A similar and more accurate modeling approach is done by Zhao et. al. (2011). Using the same NH_4Br solution, the 2-D growth patterns of the crystal were recorded using a digital camera, and then processed to extract information regarding curvature and velocity of the solid-liquid interface. Using these measurements, they proposed modeling the growth of the crystal using a model of the form:

$$\vec{N} \cdot \frac{d\vec{x}}{dt} = \sum_{i=0}^n a_i k^i + \gamma \frac{\partial^2 k}{\partial s^2} \quad (3.8)$$

This model closely resembles the model given by (3.3), but the main difference is that crystalline anisotropy is not taken into account. Also, given the polynomial order, not all the terms are required to be present, but only the ones that influence the evolution dynamics are chosen. Furthermore zero-power terms can be present in the model, which means that planar surfaces (regions on the boundary with zero curvature) are not stationary. The estimated model structure was found to be:

$$\vec{N} \cdot \frac{d\vec{x}}{dt} = 0.071 + 3.349k - 1709.49k^2 + 3.655 \frac{\partial^2 k}{\partial s^2} \quad (3.9)$$

The model (3.9) can reproduce the behavior of the real system, but its accuracy is limited to short time periods - limited relatively to the early stages of dendritic growth, when global parameters of the system can be ignored. This is due to the fact that the model does not take into account long range interactions,

such as change in solution concentration or temperature, which have a more significant impact on the growth on longer time scales.

3.2.2. The Stefan problem

Solidification models discussed above, that describe the growth dynamics of the crystal interface using local geometric properties, like curvature and its derivatives at the interface represent an effective method of describing the early stages of crystal growth (Zhao et. al. 2007). One of the key parameters that influences the crystal growth process over time is the change in temperature in the melt (Dantzig and Rappaz, 2009). The geometrical approach to model the growth does not directly include the effects that variation in heat flow has on the evolving boundary.

In order to take into account the changes in temperature (and possibly other parameters, such as solute concentration, pressure) that occur in the melt in which the crystal is growing, a different way to model the process is necessary. The model for solidification of a liquid, as mentioned earlier, can be viewed as a moving boundary problem, often called a sharp interface-model. The phase transition process can be described by the classical Stefan problem (Vuik, 1993, Evans, 1951, Hill, 1987).

In essence, the temperature distribution for a phase changing medium on a specified domain is obtained by solving a specific type of boundary value problem for the heat equation, where the evolving phase boundary is unknown and has to be obtained as part of the solution (Kim et. al, 2000, Crank, 1984).

For simplicity, consider a two-dimensional domain, $\Omega \subset \mathbb{R}^2, \Omega: [0, l] \times [0, l]$ in which a crystal is growing from melt; given this, a function $u: \Omega \times [0, T] \rightarrow \mathbb{R}$ (the temperature) is defined over a time interval $0 \leq t \leq T$, and an interface $\Gamma(t)$ that separates Ω in the solid and liquid phases denoted by two subdomains $\Omega_S(t)$ and $\Omega_L(t)$ (Fig. 3.2).

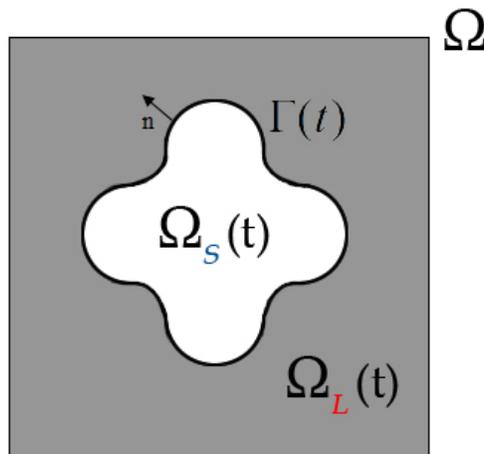


Fig. 3.2. Solid phase $\Omega_S(t)$, liquid phase $\Omega_L(t)$ and the free boundary $\Gamma(t)$ in the domain Ω

The classical two-phase Stefan problem requires finding the temperature distribution $u(x, y, t > 0)$ and the boundary $\Gamma(t)$, given the set of equations:

$$\rho c_S \frac{\partial u}{\partial t}(x, y, t) = k_S \nabla(\nabla u(x, y, t)) \text{ in } \Omega_S(t) \quad (3.10)$$

$$\rho c_L \frac{\partial u}{\partial t}(x, y, t) = k_L \nabla(\nabla u(x, y, t)) \text{ in } \Omega_L(t) \quad (3.11)$$

$$u(x, y, 0) = u_0(x, y) \text{ in } \Omega_S(0) \cup \Omega_L(0) \quad (3.12)$$

$$u(x, y, t) = u_c \text{ on } \partial\Omega \quad (3.12a)$$

$$u(x, y, t) = u_\Gamma \text{ on } \Gamma(t) \quad (3.13)$$

where ρ is the material density, c_S , c_L and k_S , k_L are the specific heat capacities and the heat conductivities in the solid and liquid subdomains, u_0 is the initial temperature distribution, u_c is the Dirichlet boundary condition on the boundary of the domain, $\partial\Omega$ and u_Γ is the temperature on the interface. The heat equations in $\Omega_S(t)$ and $\Omega_L(t)$ can be written in a more compact form as:

$$\frac{\partial u}{\partial t}(x, y, t) = D_{S/L} \nabla(\nabla u(x, y, t)) \quad (3.14)$$

where $D_{S/L} = \frac{k_{S/L}}{\rho c_{S/L}}$ is the thermal diffusivity.

In addition to the set of equations (3.10) – (3.13) a conservation law on $\Gamma(t)$ is needed in order to balance the heat absorbed on the boundary and the latent heat generated through solidification:

$$\rho L V_\Gamma \cdot n = k_S \Delta u_S \cdot n - k_L \Delta u_L \cdot n \quad (3.15)$$

Here n is the outward normal to the interface and L is the latent heat. Equation (3.15) is called the Stefan condition and states that the normal velocity on the interface V_Γ is prescribed by the jump of the temperature gradient over $\Gamma(t)$.

In order to link the Stefan problem to crystal growth and dendritic solidification, the interface condition (3.13) needs to be modified in order to consider the influences of surface tension, molecular kinetics and anisotropy. In such situations the isothermal condition on the interface (3.13) is modified by the Gibbs-Thomson relation:

$$u(x, y, t) = -\sigma_c k - \sigma_v V_\Gamma \quad (3.13a)$$

where σ_c and σ_v represent the surface tension and the kinetic coefficients (which are constant in the case of isotropic growth) and k is the curvature on $\Gamma(t)$.

The Stefan problem is one of the most common methods of modeling dendritic crystal growth (Gupta, 2003). Although it provides a more accurate description of the process than the previous discussed class of geometric models, solving the Stefan problem it is computationally intensive and it poses difficulties in numerical simulation (Rubinshteĭn, 1971, Javierre et al, 2006, Tan and Zabaras, 2006). Numerical simulations have been done using different types of front tracking methods. The interface location is

tracked either explicitly (Juric and Tryggvason, 1996) or implicitly, using a phase field model (Kobayashi, 1993) or the level set method (Chen et. al, 1997; Gibou et. al., 2003; Sethian and Strain, 1992; Leung et. al., 2004). A review of these numerical methods used to capture and track the evolution of an interface will be detailed in Section 3.3.

3.2.3. Cellular automata models for dendritic crystal growth

A recently new method of modeling dendritic crystal growth was proposed in the form of cellular automata (CA) models. In order to simulate the previous types of models – geometric models and the Stefan problem – the equations that describe the growth must be discretized over a finite mesh. To obtain a more precise approximation to the models and to ensure numerical stability in numerical simulations, the spatial and temporal steps taken must be chosen relatively small. The CA modeling approach provides a mathematical approximation of the system dynamics where the time, space, as well as the state variables belong to a discrete set.

Most commonly, the transition rule for the CA model is coupled with the diffusion equation (3.14) in the whole domain, and the Gibbs-Thomson correction (3.16) on the cells that represent the interface, in order to include effects typical to dendritic growth. Such models have been simulated by Wei et.al. (2010), Sakaguchi and Ohtaki (1999), Reiter (2004) and show a good resemblance to the standard model of solidification (the Stefan problem) in two dimensions.

Zhao et. al. (2009) proposed a new method of determining the transition rule for a binary CA, in the case of dendritic growth, directly from experimental data. By observing the patterns formed by a growing NH_4Br dendrite, a transition rule was determined based on the speed at which the boundary evolves as:

$$c(x, y, t + 1) = \max \left(m, c(x, y, t) + \frac{1}{M} \sum_{i=1}^M |\vec{V}_{(x_i, y_i)}| \right) \quad (3.16)$$

where $c(x, y, t + 1)$ represents the state of cell $c(x, y)$ at time $t + 1$, $\vec{V}_{(x_i, y_i)}$ denotes the velocities of significant occupied cells in the considered neighborhood (Moore), M denotes the total number of such cells and m is a noise reduction parameter. Although the model can exhibit features similar to models coupled with diffusion, it is somewhat limited, in the sense that the transition rule was determined from a relatively small amount of experimental data. The main advantage of this modeling approach is that the behavior of real crystal growth can be translated in the transition rule for a relatively simple model, which to some degree can predict dendritic growth.

The significant advantage of CA models is that they are far less expensive computationally to

implement than other solidification models, one drawback being the large number of cells needed in order to capture more accurately the growth dynamics; the resolution is needed because CA are usually affected by artificial anisotropy, which is more obvious when the mesh has a low resolution.

3.3. Numerical methods for solving interface evolution problems

The model equations presented in the previous sections (curvature models, the Stefan problem) in general do not have an analytical solution. Analytical solutions are only available under very restrictive conditions that are not found in realistic cases. Analysis of more realistic solidification scenarios requires numerical methods. When simulating numerically these evolution models a challenging problem is keeping track of the moving interface. Methods used in tracking the evolution of an interface can be divided into two categories: front-tracking methods, where the position of the interface needs to be explicitly updated at each time step, and implicit methods, which allow the front to move through a fixed lattice without explicitly dealing with the interface position. (Javierre-Perez, 2003). In this thesis, implicit methods will be deployed in the numerical studies presented.

Front tracking methods use an explicit representation of the moving interface. This is done by adding marker particles on the interface, and then moving the particles under the velocity on the interface. These methods can be used with success in cases where the interface is relatively simple. The cases where the interface complexity is high (such as dendritic shapes) need special care and can lead to instability. This is due to the fact that particles need to be added/removed, redistributed on the interface and reconnected at each time step, a process that becomes more complicated in higher dimensions. Also topological changes of the interface (merging/braking) are also difficult to treat. Although front tracking methods have a number of disadvantages, Juric and Tryggvason (1996) developed such a method for solidification using a front-tracking algorithm. Their method was able to describe two dimensional dendritic growth using the Stefan problem. Marker particles were added to the growing interface in order to maintain a high resolution and stability.

Implicit methods use a global parameterization of the interface, rather than a local one (such as in front tracking methods). The most used implicit methods are the phase field method and the level set method.

The phase-field method is one of the most popular methods for simulating solidification (Caginalp and Fife, 1986, Suzuki et al., 2002, Fabbri and Voller, 1997). This method avoids tracking the interface by using a function $\varphi(\mathbf{x}, t)$, called the order parameter, which is coupled with the temperature. The function $\varphi(\mathbf{x}, t)$ is defined over the entire computational domain and has values in the interval $[0,1]$. The phase field function takes constant values in each phase: $\varphi(\mathbf{x}, t) \rightarrow 0$ in the solid region, $\varphi(\mathbf{x}, t) \rightarrow 1$ in the liquid region far away from the interface, and the region where the relationship $0 < \varphi(\mathbf{x}, t) < 1$ holds, defines

the interface:

$$\varphi(\mathbf{x}, t) = \begin{cases} 1 & \text{if } \mathbf{x} \text{ is outside the interface} \\ 0 & \text{if } \mathbf{x} \text{ is on the interface} \\ -1 & \text{if } \mathbf{x} \text{ is inside the interface} \end{cases} \quad (3.17a)$$

This phase field variable is coupled with the heat equation by:

$$\begin{cases} \varepsilon^2 \frac{\partial \varphi}{\partial t} = -\frac{\delta F}{\delta \varphi} \\ \frac{\partial u}{\partial t} = k \Delta u - \frac{L}{2} \frac{\partial \varphi}{\partial t} \\ F(\varphi, u) = \int \left[\frac{1}{2} \varepsilon^2 (\nabla \varphi)^2 + f(\varphi, u) \right] \end{cases} \quad (3.17b)$$

Here ε is a parameter that describes the width of the interface (i.e. how fast the transition $0 \rightarrow 1$ is made), $F(\varphi, u)$ is an energy functional, $f(\varphi, u)$ is a free energy function that constitutes the difference between phase field models, $\frac{\delta F}{\delta \varphi}$ denotes the derivative of $F(\varphi, u)$ with respect to the phase field variable, k and L represent the thermal diffusivity and the latent heat and u is the temperature. Notice that there are no boundary conditions imposed on the interface and in order for this model to describe the Stefan problem special conditions must be imposed. To obtain this similarity, an asymptotic analysis of the model is required. This analysis consists in expanding the phase-field equations in a parameter proportional to the interface width ε and as $\varepsilon \rightarrow 0$ the phase-field model reproduces the dynamics of the Stefan problem. This method is one of the most popular used to simulate solidification and has been studied and implemented numerically in numerous papers. Simulations performed by Kobayashi (1993) were the first simulations of dendritic growth that are in agreement with the sharp interface model. Wang and Sekerka (1996) studied the effects of the parameter ε on the model and the convergence to the Stefan problem. They showed that the grid spacing needs to be very small in order for the phase field model to converge to the sharp interface model, which makes the phase field method impractical to use on uniform grids; a solution to this limitation would be a grid adaptation to solve the problem as $\varepsilon \rightarrow 0$.

The level set method is implicit method for front tracking that is based on embedding the evolving interface into a higher dimensional function. The method was first introduced by Osher and Sethian (1988) to study the effects of evolving curves under normal speed equal to the local curvature. Since then, it has been widely used in areas such as image processing, computer vision, optimization problems and mainly computational physics (Stefan problems, liquid-gas interactions, liquid flow, heat flow, flame propagation etc.). One important property of the level set approach is that it can solve the interface location exactly and there is no interface width as in the phase-field model. Because of this, there is no need for asymptotic analysis of the model. Another major advantage of the level set method is that topological changes of the interface, such as braking and merging are very easy to handle.

The main idea of the level set approach is to use a higher dimensional function, φ , called a level set

function, to parameterize the domain in which the interface is evolving, so that the zero level set of the function φ corresponds to the interface. The level set function is chosen with the following property:

$$\varphi(\mathbf{x}, t) = \begin{cases} +d & \text{if } \mathbf{x} \text{ is outside the interface} \\ 0 & \text{if } \mathbf{x} \text{ is on the interface} \\ -d & \text{if } \mathbf{x} \text{ is inside the interface} \end{cases} \quad (3.18)$$

where d represents the minimum distance from the interface Γ and the point $\mathbf{x} \in \mathbb{R}^n$ ($d = \min|\Gamma - \mathbf{x}|$). If $\Gamma(t)$ represents an interface moving in a domain $D \subset \mathbb{R}^2$, and $\varphi: \mathbb{R}^2 \times [0, T] \rightarrow \mathbb{R}$ is the level set function, then the interface at any given time is defined by $\Gamma(t) = \{\mathbf{x}: \varphi(\mathbf{x}, t) = 0\}$ (Fig. 3.3).

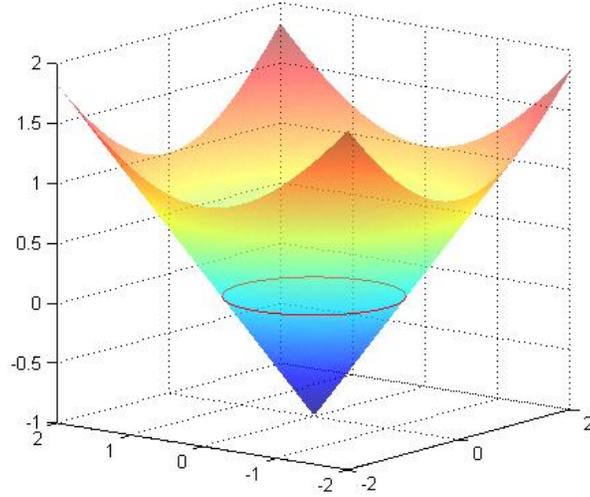


Fig. 3.3. Embedding of a circle as the zero level-set of a higher dimensional function

In order to track the movement of the interface, the zero level set of the function φ has to evolve in the same way as the interface. Therefore any point $\mathbf{x}(t)$ on the interface must satisfy the condition:

$$\varphi(\mathbf{x}(t), t) = 0 \text{ for } 0 \leq t \leq T \quad (3.19)$$

Taking the derivative of equation (3.19) with respect to the time parameter t results in:

$$\frac{\partial \varphi}{\partial t}(\mathbf{x}(t), t) + \nabla \varphi(\mathbf{x}(t), t) \cdot \dot{\mathbf{x}}(t) = 0 \text{ for } 0 \leq t \leq T \quad (3.20)$$

where $\dot{\mathbf{x}}(t) = V(t)$ represents the velocity of the point $\mathbf{x}(t)$. Equation (3.20) is called the level set equation and is valid only on the interface. By defining a function F as a continuous extension of $V(t)$ over the whole computational domain, the level set equation can be written as:

$$\varphi_t + \vec{F} \nabla \varphi = 0 \quad (3.21)$$

Notes on algorithms used to construct velocity extensions off the interface can be found in (Adalsteinsson and Sethian, 1997) and (Chopp, 2009).

This implicit representation in the form of a level set of a function has a series of attractive geometric properties (Osher and Fedkiw, 2003):

a) the outward normal vector to $\Gamma(t)$ is simply given by: $n = \frac{\nabla\varphi}{|\nabla\varphi|}$

b) the curvature of the interface can be computed from: $k = \nabla \cdot n$. The curvature is signed, so that it takes positive values in convex regions, negative values in concave regions and it is equal to zero for planar regions.

c) by taking $\varphi(\mathbf{x}, 0)$ as a signed distance function, the interior/exterior of the interface can be easily accessed.

d) the level set method can be easily extended to higher dimensions.

Equation (3.20) can be written in a more general form by using property (a):

$$\varphi_t + V \cdot n |\nabla\varphi| = 0 \quad (3.22)$$

This form of the level set equation is the most common form found in literature and is very suitable when only the normal speed $V \cdot n$ is known, and the actual velocity field V is not given explicitly, which is generally the case in computational physics.

Equations (3.21) and (3.22) are particular forms of the general Hamilton-Jacobi equation:

$$\varphi_t + H(\nabla\varphi) = 0 \quad (3.23)$$

where the Hamiltonian $H(\cdot)$ is a function that depends at most on the first order derivatives of the level set function. The Hamiltonian has a linear form in the case of (3.21) and is given by:

$$H_1(\nabla\varphi) = V \cdot \nabla\varphi \quad (3.24)$$

whereas in the case of (3.22) the Hamiltonian is non-linear:

$$H_2(\nabla\varphi) = V \cdot n |\nabla\varphi| \quad (3.25)$$

Methods for solving Hamilton-Jacobi type equations have been widely studied and several numerical methods exist for problems described by (3.24) and (3.25). If the speed function is given by local curvature, i.e. $V \cdot n = k = \nabla \cdot \left(\frac{\nabla\varphi}{|\nabla\varphi|} \right)$, then the level set equation (3.21) is not a Hamilton-Jacobi equation, since the front speed depends on second order derivatives of φ and requires a different set of numerical schemes to solve (Osher and Fedkiw, 2003).

Currently, the level set method has been successfully used to simulate the patterns in dendritic crystal growth. Sethian and Strain (1992) were the first to present a method for solving the standard model of solidification. Their solution combined boundary integral approach with the level set method to calculate the normal velocity in the interface. Their method was able to accurately track the evolution of complicated shapes that exhibit complex behavior typical to dendritic growth. Later, in Chen et. al (1997) an improvement is made by developing a simpler way of finding the normal velocity, without any restrictions imposed by a boundary integral formulation. A similar approach was done by Fried (2004) by using a finite element method for the Stefan problem, where the boundary is expressed as a zero level set. Leung et. al. (2004) propose a new approach to the level set formulation of the Stefan problem. They use an adaptive

mesh refinement, to modify the density of grid points close to the interface. This gives the advantage of less computational complexity and higher resolution in tracking the interface. A similar technique is used in (Chen et. al., 2009), where the spatial ratio between neighboring cells is adjusted.

3.4. Numerical studies of crystal growth in 2D

This section discusses of the difficulties that arise with numerical discretization and provides simulations of crystal growth models presented in the previous section.

3.4.1. Time and Space Discretization

In the following sections the level set function φ is considered to be defined in a subdomain $D \subset \mathbb{R}^2$, such that the domain D can be represented on a Cartesian grid. Consider the grid with the following properties: the points on the grid are uniformly spaced in an interval $0 \leq x \leq L_1$, $0 \leq y \leq L_2$, such that

$$\begin{aligned} x_i &= (i - 1)\Delta x, \quad i = 1, 2, \dots, N \\ y_j &= (j - 1)\Delta y, \quad j = 1, 2, \dots, M \end{aligned}$$

where N is the total number of nodes in the x direction and M is the total number of nodes in the y direction. Given L_1 , L_2 and N , M the spacing between 2 grid points is given by:

$$\Delta x = \frac{L_1}{N-1} \quad \text{and} \quad \Delta y = \frac{L_2}{M-1}$$

For simplicity it is assumed that $\Delta x = \Delta y = h$ and the number of points in each direction is taken to be equal.

In a similar fashion the discrete time t is considered uniformly spaced in the interval $0 \leq t \leq T$:

$$t_k = (k - 1)\Delta t, \quad k = 1, 2, \dots, P$$

where P is the number of time steps and Δt is the size of a time step: $\Delta t = \frac{T}{P-1}$.

On this discrete mesh the following approximations to the derivatives of the level set function can be defined:

- first order accurate forward and backward difference:

$$\varphi_x^+ \approx \frac{\varphi_{i+1} - \varphi_i}{\Delta x}, \quad \varphi_y^+ \approx \frac{\varphi_{j+1} - \varphi_j}{\Delta x} \quad (3.26)$$

$$\varphi_x^- \approx \frac{\varphi_i - \varphi_{i-1}}{\Delta x}, \quad \varphi_y^- \approx \frac{\varphi_j - \varphi_{j-1}}{\Delta x} \quad (3.27)$$

- second order accurate central difference:

$$\varphi_x \approx \frac{\varphi_{i+1} - \varphi_{i-1}}{2\Delta x}, \quad \varphi_y \approx \frac{\varphi_{j+1} - \varphi_{j-1}}{2\Delta y} \quad (3.28)$$

- second order accurate finite difference for the second derivative :

$$\varphi_{xx} \approx \frac{\varphi_{i+1} - 2\varphi_i + \varphi_{i-1}}{\Delta x^2}, \varphi_{yy} \approx \frac{\varphi_{j+1} - 2\varphi_j + \varphi_{j-1}}{\Delta y^2} \quad (3.29a)$$

$$\varphi_{xy} \approx \frac{\varphi_{i+1,j+1} - \varphi_{i+1,j-1} - \varphi_{i-1,j+1} + \varphi_{i-1,j-1}}{4\Delta x\Delta y} \quad (3.29b)$$

3.4.2. Simulation of geometric models of dendritic crystal growth

In this section issues related to numerical discretization and simulation of individual components of the geometric models are analyzed and simulations presented. Also a short analysis of difficulties in simulating geometric models is presented.

3.4.2.1. Motion of level sets in the normal direction

This section discusses the motion of an interface under an internally produced velocity field for constant motion in the normal direction. Let us recall the general level set equation:

$$\varphi_t + V|\nabla\varphi| = 0 \quad (3.30)$$

Motion on an interface in the normal direction is given by equation (3.30) where $V = a$, where a is chosen as a constant term. The sign of the speed a determines the direction in which the interface is moving at a given time: if $a > 0$ the interface moves in the direction of the normal and if $a < 0$ the interface will move opposite the normal direction. In the trivial case where $a = 0$, equation (3.30) is reduced to $\varphi_t = 0$, which means the interface is stationary at all times. Equation (3.30) is a Hamilton-Jacoby equation, with the Hamiltonian of the form $H(\varphi_x, \varphi_y)$ and there are a variety of numerical solvers available, as detailed in (Osher and Fedkiw, 2003). In this project, Godunov's method is used to compute spatial derivatives in equation (3.30). This method ensures that the flow of information is correctly discretized in time, i.e. how the gradient $\nabla\varphi = (\varphi_x, \varphi_y)$ is computed depending on the sign of the speed. The method for equation (3.30) is as follows, for both positive and negative values of the speed term:

- if $a\varphi_x^+ > 0$ and $a\varphi_x^- > 0$, $\varphi_x = \varphi_x^-$
- if $a\varphi_x^+ < 0$ and $a\varphi_x^- < 0$, $\varphi_x = \varphi_x^+$
- if $a\varphi_x^+ \geq 0$ and $a\varphi_x^- \leq 0$, $\varphi_x = 0$
- if $a\varphi_x^+ \leq 0$ and $a\varphi_x^- \geq 0$, $\varphi_x = \begin{cases} \varphi_x^+, |a\varphi_x^+| > |a\varphi_x^-| \\ \varphi_x^-, |a\varphi_x^-| > |a\varphi_x^+| \end{cases}$

The same process applies to obtain the derivative in any spatial dimension. After choosing the correct

values for the derivatives the gradient magnitude can be found simply by $|\nabla\varphi| = \sqrt{\varphi_x^2 + \varphi_y^2}$. The front is then advanced in time using a third order accurate TVD-RK (total variation diminishing Runge Kutta) method (Shu and Osher, 1989):

- first a forward Euler step is taken to advance the solution to time $t + \Delta t$:

$$\frac{\varphi^{t+1} - \varphi^t}{\Delta t} + V^t |\nabla\varphi^t| = 0$$

- φ^{t+1} is then used in another Euler step to advance the solution to time $t + 2\Delta t$:

$$\frac{\varphi^{t+2} - \varphi^{t+1}}{\Delta t} + V^{t+1} |\nabla\varphi^{t+1}| = 0$$

- this is followed by an averaging : $\varphi^{t+\frac{1}{2}} = \frac{3}{4}\varphi^t + \frac{1}{4}\varphi^{t+2}$, which produces a solution for time $t + \frac{1}{2}\Delta t$

In the case where the speed term is not constant, the term $V^{t+\frac{1}{2}}$ is computed from the relevant governing laws of the process (e.g. if the front evolves with speed proportional to local curvature, $V^{t+\frac{1}{2}}$ will be computed from the curvature of $\varphi^{t+\frac{1}{2}}$)

- the fourth stage is another forward Euler step , to advance the solution to time $t + \frac{3}{2}\Delta t$:

$$\frac{\varphi^{t+\frac{3}{2}} - \varphi^{t+\frac{1}{2}}}{\Delta t} + V^{t+\frac{1}{2}} |\nabla\varphi^{t+\frac{1}{2}}| = 0$$

- the final solution at time $t + \Delta t$ is then found by a second averaging step :

$$\varphi^{t+1} = \frac{1}{3}\varphi^t + \frac{2}{3}\varphi^{t+\frac{3}{2}}$$

Higher order integration schemes exist in literature, but require more computational effort and there are no major advantages in practical calculations.

To ensure stability the time step used in the numerical integration method must satisfy the CFL condition (Courant–Friedrichs–Lewy condition), which states that the time step must be taken smaller than a certain value, otherwise the simulation will produce incorrect results. This condition affirms that the speed $\Delta x/\Delta t$ of the numerical waves must be at least as fast as the absolute value of the physical wave speed $|V|$. The CFL step restriction for motion in the normal direction is:

$$\Delta t < \frac{1}{\max\left(\frac{|a\varphi_x|}{\Delta x} + \frac{|a\varphi_y|}{\Delta y}\right)} \quad (3.31)$$

Figure 3.4. illustrates the motion of an initial shape under positive normal velocity $V_n = 0.5$. The domain is taken to be $D = [-2,2] \times [-2,2]$, with a number of grid nodes $N = 200$ in each spatial direction, giving a spatial step size of $\Delta x = \Delta y = 0.201$. The time step restriction given by (3.31) is $\Delta t = 0.0284$ s. Time plots of the evolution are taken at every $\Delta t_{plot} = 0.125$ s until the final simulation time $T = 1$ s. Figure 3.5. illustrates motion under the same spatial and temporal conditions, but with negative speed $V_n = -0.5$.

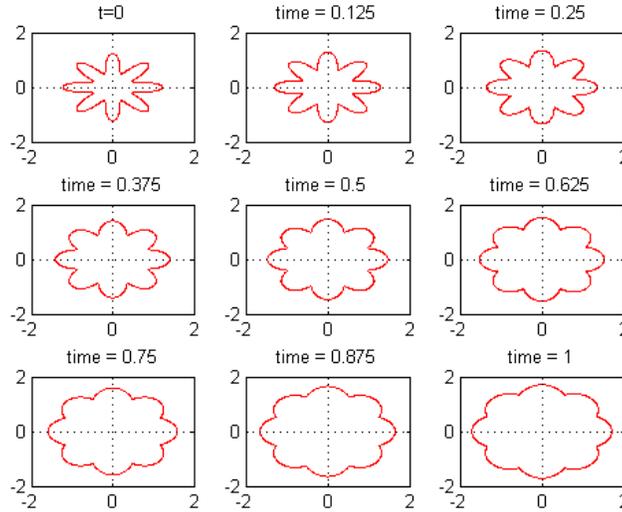


Fig. 3.4. Evolution of a star shaped curve under positive normal velocity

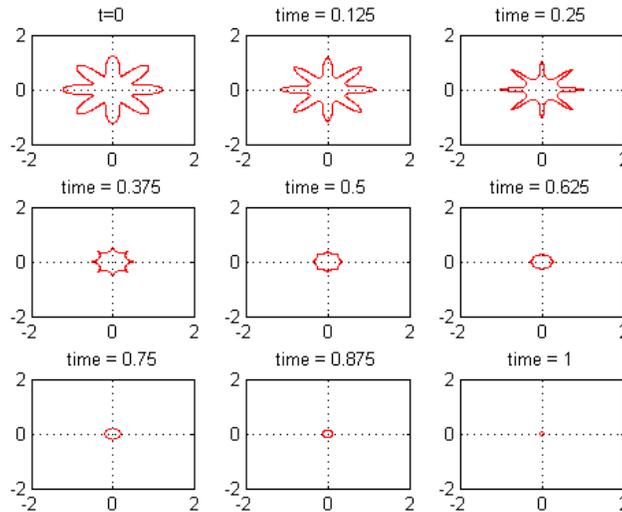


Fig. 3.5. Evolution of a star shaped curve under negative normal velocity

3.4.2.2. Motion of level sets under curvature flow

The local curvature is given by the divergence of the normal vector and can be written in terms of the level set function as:

$$k = \nabla \cdot \left(\frac{\nabla \varphi}{|\nabla \varphi|} \right) = \frac{\varphi_x^2 \varphi_{yy} - 2\varphi_x \varphi_y \varphi_{xy} + \varphi_y^2 \varphi_{xx}}{|\nabla \varphi|^3} \quad (3.32)$$

By replacing the normal speed with $V_n = -bk^q$, where q is the power of the curvature term and b is chosen as a positive constant, the equation that characterizes motion with speed proportional to curvature is obtained as:

$$\varphi_t = bk^q \cdot |\nabla \varphi| \quad (3.33)$$

Equation (3.33) can be solved by using the TVD-RK method previously described, using a CFL time step restriction given by:

$$\Delta t < \frac{1}{\frac{2b}{((\Delta x)^2)^q} + \frac{2b}{((\Delta y)^2)^q}} \quad (3.34)$$

Motion of level sets under curvature flow is presented in *Fig.3.6*. The domain is taken to be $D = [-2,2] \times [-2,2]$, with a number of grid nodes $N = 200$ in each spatial direction, giving a spatial step size of $\Delta x = \Delta y = 0.201$. The time step restriction of $\Delta t = 5.0504 \cdot 10^{-4}$ s is a few orders in magnitude smaller than the case of motion with constant normal speed. Time plots of the evolution are taken at every $\Delta t_{plot} = 0.125$ s until the final simulation time $T = 1$ s. The weighting term of the curvature is taken $b = 0.2$ and the power term $q = 1$. Motion under curvature shrinks any curve into circles, which then continue to shrink into a single point and vanish.

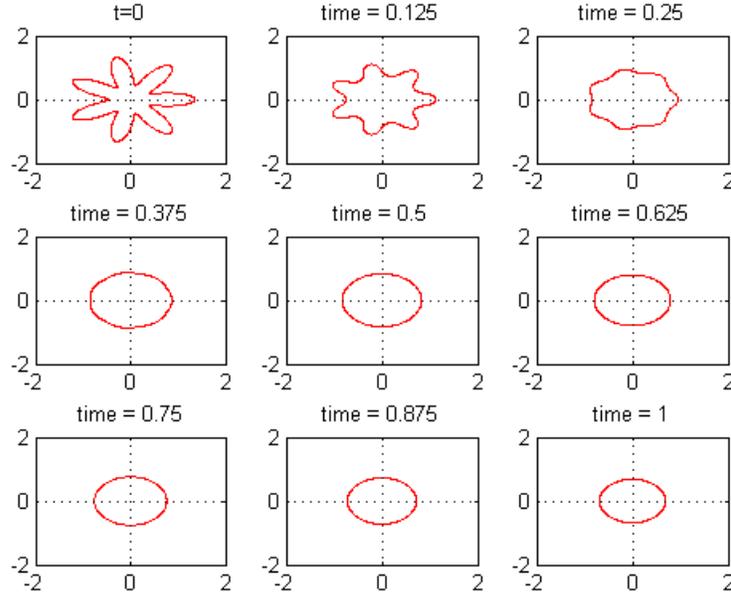


Fig. 3.6. Evolution of a star shaped curve under curvature flow

3.4.2.3. Motion of level sets under curvature flow combined with advection in the normal direction

This type of motion is characterized by normal speed $V_n = a - bk$, which gives the following level set equation:

$$\varphi_t + a|\nabla\varphi| = bk \cdot |\nabla\varphi| \quad (3.35)$$

and can be solved by discretizing separately the normal term and the curvature term, as discussed in previous sections. Once both terms are discretized, a 3rd order TVD-RK method can be used to advance the front forward in time. The CFL condition required to ensure stability of equation (3.35) is given by:

$$\Delta t < \frac{1}{\max\left(\frac{|a\varphi_x|}{\Delta x} + \frac{|a\varphi_y|}{\Delta y}\right) + \frac{2b}{(\Delta x)^2} + \frac{2b}{(\Delta y)^2}} \quad (3.36)$$

This type of motion can be viewed as a linearization of geometrical crystal growth model (3.3) and is illustrated in Fig.3.7. This can be regarded as a simple crystal growing under two competing forces: the normal velocity term, which sustains growth and tends to expand the crystal and the curvature force that smoothes out the irregularities on the crystal shape, playing the role of a surface tension type-force. The linearized model of crystal growth (3.35) cannot predict dendritic crystal growth: over time, the curvature term will continue to flatten any perturbations on the interface, until it eventually becomes a circle, which then can only be expanded by the normal term, hence complex shapes present in dendritic growth cannot form.

The domain is taken to be $D = [-2,2] \times [-2,2]$, with a number of grid nodes $N = 200$ in each spatial direction, giving a spatial step size of $\Delta x = \Delta y = 0.201$. The time step restriction is $\Delta t = 0.0046$ s. Time plots of the evolution are taken at every $\Delta t_{plot} = 0.125$ s until the final simulation time $T = 1$ s. The normal velocity is $V_n = 0.25$ and the curvature weight is $b = 0.02$.

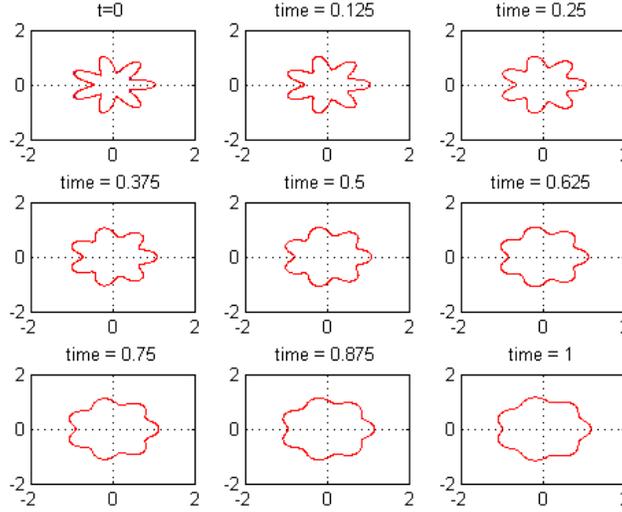


Fig. 3.7. Evolution of a star shaped under speed equal to the curvature combined with a normal speed term. Perturbations on the curve are diminished in time, and the shape tends to turn circular.

3.4.2.4. Motion of level sets under the second derivative of curvature

The level-set equation that characterizes the motion of a closed curve under this normal velocity is given by (Chopp and Sethian, 1999):

$$\varphi_t + c \cdot k_{ss} |\nabla \varphi| = 0 \quad (3.37)$$

where c is a weighting factor and the term k_{ss} denotes the second order derivative of the curvature with respect to local arc-length, and can be expressed in terms of the level set function φ as:

$$k_{ss} = \nabla \left[\nabla \left[\nabla \cdot \frac{\nabla \varphi}{|\nabla \varphi|} \right] \cdot \frac{(\varphi_y, -\varphi_x)}{|\nabla \varphi|} \right] \cdot \frac{(\varphi_y, -\varphi_x)}{|\nabla \varphi|} \quad (3.38)$$

$$= \frac{k_{xx} \varphi_y^2 - 2k_{xy} \varphi_x \varphi_y + k_{yy} \varphi_x^2}{\varphi_x^2 + \varphi_y^2} - \frac{(k_x \varphi_x + k_y \varphi_y) k}{\sqrt{\varphi_x^2 + \varphi_y^2}} \quad (3.39)$$

where k is the curvature given by (3.32) and the subscripts refer to derivatives with respect to the two spatial directions. It is obvious from (3.38) that the k_{ss} term involves computing derivatives of the level set function up the fourth order, which can lead to a noisy numerical simulation. In order to reduce errors

in numerical estimates, (Chopp and Sethian, 1999) propose using fourth order accurate approximations to the derivatives of φ , instead of (3.28)-(3.29a, b):

$$\varphi_x \approx \frac{1}{12\Delta x} (-\varphi_{i+2,j} + 8\varphi_{i+1,j} - 8\varphi_{i-1,j} + \varphi_{i-2,j}) \quad (3.40a)$$

$$\varphi_{xx} \approx \frac{1}{12(\Delta x)^2} (-\varphi_{i+2,j} + 16\varphi_{i+1,j} - 30\varphi_{i,j} + 16\varphi_{i-1,j} - \varphi_{i-2,j}) \quad (3.40b)$$

$$\begin{aligned} \varphi_{xy} \approx \frac{1}{48(\Delta x)^2} & (-\varphi_{i+2,j+2} + 16\varphi_{i+1,j+1} + \varphi_{i-2,j+2} - 16\varphi_{i-1,j+1} + \varphi_{i+2,j-2} - 16\varphi_{i+1,j-1} \\ & - \varphi_{i-2,j-2} + 16\varphi_{i-1,j-1}) \end{aligned} \quad (3.40c)$$

Given (3.40a, b, c), the derivatives φ_y , φ_{yy} , k_x , k_{xx} , k_y , k_{yy} and k_{xy} are approximated using the same numerical stencils. After all terms are discretized accordingly, the front can be advanced by using the 3rd order TVD-RK scheme previously presented. Under this type of curvature flow, closed curves tend to shrink into circles, and then remain stationary, due to the fact that the second order derivative of the curvature of a circle is equal to zero, which means that the level sets remain static.

This motion is demonstrated in *Fig. 3.8*, *Fig. 3.9*. The domain is taken to be $D = [-2,2] \times [-2,2]$, with a number of grid points $N = 50$ in each spatial direction, giving a spatial step size of $\Delta x = \Delta y = 0.0816$. The time step restriction is $\Delta t = 5 \cdot 10^{-6}$ s. Time plots of the evolution are taken at every $\Delta t_{plot} = 0.125$ s until the final simulation time $T = 0.1$ s. The curvature weighting term is $c = 0.06$. A larger weight would mean a smaller time restriction in order to ensure stability, which would greatly increase simulation time.

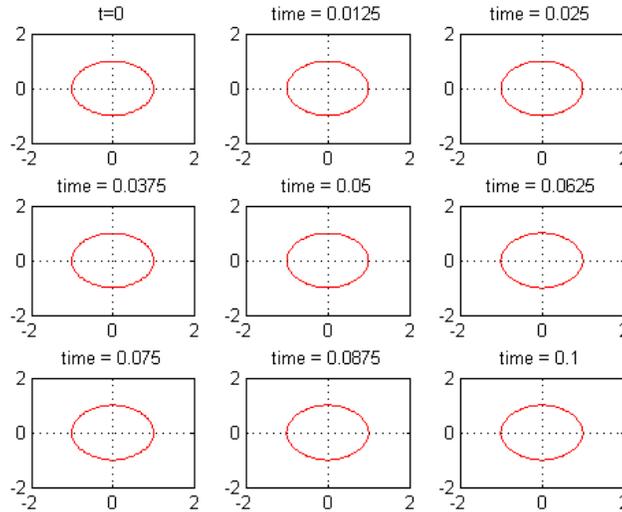


Fig. 3.8. A circle under curvature flow given by k_{ss} , remains stationary (after 20 000 iterations)

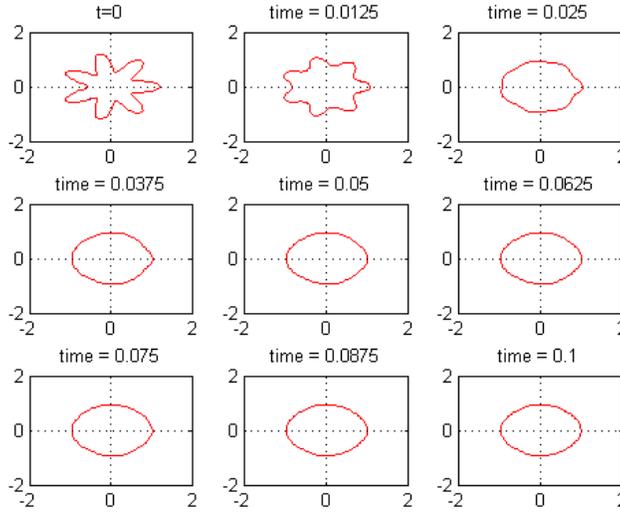


Fig. 3.9. A star shape curve evolving under curvature flow given by k_{ss} , transforms approximately into a circle after 20 000 iterations

3.4.2.5. Numerical difficulties of geometric models of dendritic crystal growth

From the previous sections it can be seen that geometric models of crystal growth have the general form:

$$\vec{N} \cdot \frac{d\vec{x}}{dt} = \sum_{i=0}^n a_i k^i + \gamma \frac{\partial^2 k}{\partial s^2}$$

where the coefficients a_i and γ take real positive or negative values and n is the order of the curvature polynomial. This general model can be re-written in a level set representation as:

$$\varphi_t + \left(a_0 + \sum_{i=1}^n a_i k^i + \gamma k_{ss} \right) \cdot |\nabla \varphi| = 0$$

$$\varphi_t + a_0 \cdot |\nabla \varphi| = - \left(\sum_{i=1}^n a_i k^i + \gamma k_{ss} \right) \cdot |\nabla \varphi|$$

Motion of level sets under curvature flow or a general flow involving curvature and its derivatives is numerically stable only if the weighting term is taken to be positive. The case where this constant has a negative value, i.e. $a_i < 0$ or $\gamma < 0$, allows for small perturbations caused by numerical approximations to grow, and lead to instability (Osher and Fedkiw, 2003). There was no study found on a regularization method to stabilize motion of level sets under this kind of flow.

Unfortunately geometric models presented have polynomial terms which are weighted by negative terms, which cause the simulation of such components to be numerically unstable.

3.4.3. Simulation of the modified Stefan problem

Numerical simulations of non-local models of growth require discretization of the set of equations (3.10)-(3.16). The velocity of the interface is given by the Stefan condition (3.15) which can then be used to evolve the front using a level set formulation. The method developed by (Chen. et. al., 1997) was implemented in this project in order to simulate dendritic growth. Solving the modified Stefan problem is usually a three step process:

a) firstly the initial temperature distribution is defined in a computational domain, an initial crystal seed is set, and the level set function attached to the seed is constructed. With this initial information the normal velocity can be determined from the Stefan condition (3.15). This speed is only valid for points on the interface and needs to be extrapolated to the whole domain, to properly move all level sets.

b) after the speed function is found, the front can be advanced forward in time with the level set method by using a third order TVD-RK scheme. This motion usually does not preserve the signed distance property of the level set function and, after each time step the level set function needs to be reinitialized as a signed distance function.

c) the last step involves updating the temperature in the solid and liquid subdomains and applying the Dirichlet boundary condition (3.13a) on the interface.

3.4.3.1. Velocity extension

The normal velocity is defined only on the interface, but can be easily extended to the whole domain using the advection equation:

$$V_{\tau_{ext}} + S(\varphi\varphi_x)V_x = 0 \quad (3.41)$$

where $S(\cdot)$ represents the sign function, u is the temperature, $V_x = \frac{\partial u}{\partial x}$ on the interface, τ_{ext} is a fictitious time. In (Chen. et. al., 1997) the authors propose using four different terms in approximating the extension of the velocity: the two Cartesian coordinates $x - y$ and the 45° rotated Cartesian coordinates $\xi - \eta$, in order to minimize the effects of grid orientation. Discretization of the four speed terms are given by:

$$u_{i,j}^1 = -S_{i,j}(\varphi_x) \left((u_{i+2,j} - u_{i+1,j}) - (u_{i-1,j} - u_{i-2,j}) \right) / h \quad (3.42a)$$

$$u_{i,j}^2 = -S_{i,j}(\varphi_y) \left((u_{i,j+2} - u_{i,j+1}) - (u_{i,j-1} - u_{i,j-2}) \right) / h \quad (3.42b)$$

$$u_{i,j}^3 = -S_{i,j}(\varphi_\eta) \left((u_{i+2,j+2} - u_{i+1,j+1}) - (u_{i-1,j-1} - u_{i-2,j-1}) \right) / \sqrt{2}h \quad (3.42c)$$

$$u_{i,j}^4 = -S_{i,j}(\varphi_\xi) \left((u_{i+2,j-2} - u_{i+1,j-1}) - (u_{i-1,j+1} - u_{i-2,j+2}) \right) / \sqrt{2}h \quad (3.42d)$$

where u^1, u^2, u^3, u^4 are the temperature jumps across the interface and $\varphi_x, \varphi_y, \varphi_\eta, \varphi_\xi$ are the derivatives

of the level set function taken in the $x - y$ and $\xi - \eta$ directions, and h is the grid spacing. This discretization is followed by a smoothing step to remove any sharp jumps that may have formed, which is required to maintain accuracy when calculating the velocity field over the whole domain:

$$\begin{cases} \text{if } S_{i,j}(\varphi\varphi_x) > 0, & u_{i,j}^1 = u_{i,j}^1 - \frac{1}{2}(u_{i,j}^1 - u_{i-1,j}^1) \\ \text{if } S_{i,j}(\varphi\varphi_x) < 0, & u_{i,j}^1 = u_{i,j}^1 + \frac{1}{2}(u_{i+1,j}^1 - u_{i,j}^1) \end{cases} \quad (3.43)$$

The equations related to the other spatial directions are processed in a similar fashion as (3.43).

Finally the velocity extension is calculated from values of u^1, u^2, u^3, u^4 and the level set function φ using:

$$V_{i,j} = \frac{1}{2} \left(u_{i,j}^1 \left(\frac{\varphi_x}{|\nabla\varphi|} \right)_{i,j} + u_{i,j}^2 \left(\frac{\varphi_y}{|\nabla\varphi|} \right)_{i,j} + u_{i,j}^3 \left(\frac{\varphi_\eta}{|\nabla\varphi|} \right)_{i,j} + u_{i,j}^4 \left(\frac{\varphi_\xi}{|\nabla\varphi|} \right)_{i,j} \right) \quad (3.44)$$

This speed is defined at each grid point in the discrete mesh and can be used in equation (3.21). The spatial derivatives of φ from (3.44) are approximated by second order accurate schemes and the gradient $|\nabla\varphi|$ is computed using central difference approximations. After this, the level set equation (3.21) can be solved forward in time using the third order Runge-Kutta scheme.

3.4.3.2. Re-initialization of the level set function

After a solution of (3.21) is computed, the resultant level set function can cease to be a signed distance function ($|\nabla\varphi| \neq 1$), after only one iteration. This means that the level sets are no longer equally spaced between each other. However, conserving the signed distance property of the level set function is a key element in order to capture the interface accurately. To maintain this desired property, the solution is to reinitialize φ as a SDF (signed distance function) after each iteration. A direct method of doing this would be to find the zero level contour and recalculate the distances from it to all grid nodes. Although this approach would solve the problem, it can become computationally expensive, as interface complexity and mesh point density grow. To avoid this, (Sussman et. al., 1994) propose solving the following equation to steady-state to obtain the re-initialization effect:

$$\begin{aligned} \varphi_t &= S(\varphi_0)(1 - |\nabla\varphi|) \\ \varphi(\mathbf{x}, 0) &= \varphi_0(\mathbf{x}) \end{aligned} \quad (3.45)$$

where $\varphi_0(\mathbf{x})$ is the current level set function and $S(\cdot)$ is the sign function. Equation (3.45) is discretized by:

$$\varphi_{i,j}^{t+1} = \varphi_{i,j}^t - \Delta t \cdot S_\varepsilon(\varphi_{i,j}^0) G(\varphi_{i,j}^t) \quad (3.46)$$

With:

$$S_\varepsilon(\varphi)_{i,j} = \frac{\varphi_{i,j}}{\sqrt{\varphi_{i,j}^2 + \varepsilon^2}}, \varepsilon = 2h$$

$$\begin{cases} a = \varphi_x^- & b = \varphi_x^+ \\ c = \varphi_y^- & d = \varphi_y^+ \end{cases}$$

$$G(\varphi)_{i,j} = \begin{cases} \sqrt{\max((a^+)^2, (b^-)^2) + \max((c^+)^2, (d^-)^2)} - 1, & \text{if } \varphi_{i,j}^0 > 0 \\ \sqrt{\max((a^-)^2, (b^+)^2) + \max((c^-)^2, (d^+)^2)} - 1, & \text{if } \varphi_{i,j}^0 < 0 \\ 0 & \text{otherwise} \end{cases}$$

The superscript $(n)^+$ and $(n)^-$ denote $\max(n, 0)$ and $\min(n, 0)$ respectively and S_ε represents the sign function smoothed by order parameter ε . Usually equation (3.46) has to be iterated 3-4 times each time step to ensure that the data in φ is close to a SDF. Even though this method provides a better solution to initialization, one disadvantage is that during the process the zero level set can move, resulting in loss of area over time. Chang et. al. (1996) proposed the following procedure to solve this problem:

$$\varphi_t + (A_0 + A(t))(-P + k)|\nabla\varphi| = 0 \quad (3.47)$$

$$\varphi(\mathbf{x}, 0) = \varphi_0(\mathbf{x})$$

where A_0 is the total mass of the initial level set function $\varphi(\mathbf{x}, 0)$, $A(t)$ is the mass associated with the current level set function φ_t , P is a positive constant taken for stabilization purposes and k is the curvature of the interface.

Another approach to solve the re-initialization without the effects of mass loss is done in (Russo and Smereka, 2000) where an improvement over the algorithm proposed by (Sussman et. al., 1994) is presented. The authors suggest using different expressions for the sign function S_ε and for matrix G .

3.4.3.3. Temperature Update

Let us recall the heat equation in the domain together with the Gibbs-Thomson condition on the interface:

$$\frac{\partial u}{\partial t}(x, y, t) = D_{S/L} \cdot \nabla(\nabla u(x, y, t)) \quad (3.48)$$

$$u(x, y, t) = -\sigma_c k - \sigma_v V_\Gamma \quad (3.49)$$

Let us assume for simplicity that the thermal diffusivities in the solid and liquid region are both equal to 1. Equation (3.48) can be discretized by applying explicit or implicit methods. Although explicit methods are simpler to implement, they require a very small time step restriction for stability. Among the implicit methods the following methods are evoked: implicit Euler method and Crank-Nicolson method.

The implicit Euler method discretization is given by:

$$u^{t+1} - \Delta t \cdot \nabla(\nabla u^{t+1}) = u^t \quad (3.50)$$

where the term ∇u^{t+1} is discretized using central differences. Equation (3.50) can be rewritten on a discrete rectangular mesh as:

$$\left(1 + \frac{4\Delta t}{h^2}\right) u_{i,j}^{t+1} - \frac{\Delta t}{h^2} \cdot (u_{i+1,j}^{t+1} + u_{i-1,j}^{t+1} + u_{i,j+1}^{t+1} + u_{i,j-1}^{t+1}) = u^t \quad (3.50a)$$

In this form, it is easier to see that (3.50a) can be written as a linear system of equations

$$\mathbf{A} \cdot u^{t+1} = \mathbf{b} \cdot u^t$$

with unknowns $u_{i,j}^{t+1}$. The resulting matrix \mathbf{A} is symmetric and the system can be solved very fast by using linear solver methods (LUP decomposition, Cholesky decomposition, PCG method). This method allows choosing time steps proportional to the square of the grid spacing, greatly reducing computational effort as compared to explicit methods. The implicit Euler method is unconditionally stable.

The Crank-Nicolson integration method can be used to attain second-order accuracy in both time and space:

$$u^{t+1} - \frac{\Delta t}{2} \cdot \nabla(\nabla u^{t+1}) = u^t + \frac{\Delta t}{2} \cdot \nabla(\nabla u^t) \quad (3.51)$$

Here the Laplacian operators are discretized using only central differences. On a discrete rectangular mesh it has the form:

$$\begin{aligned} \left(1 + 2\frac{\Delta t}{h^2}\right) u_{i,j}^{t+1} - \frac{\Delta t}{h^2} \cdot (u_{i+1,j}^{t+1} + u_{i-1,j}^{t+1} + u_{i,j+1}^{t+1} + u_{i,j-1}^{t+1}) = \\ = \left(1 - 2\frac{\Delta t}{h^2}\right) u_{i,j}^t + \frac{\Delta t}{h^2} \cdot (u_{i+1,j}^t + u_{i-1,j}^t + u_{i,j+1}^t + u_{i,j-1}^t) \end{aligned} \quad (3.51a)$$

As in the case of the implicit Euler method, equation (3.51) can be written in the form of a linear system:

$$\mathbf{A} \cdot u^{t+1} = \mathbf{B} \cdot u^t$$

which can be solved in the same manner. The Crank-Nicolson is unconditionally stable but the time step needs to be chosen with care, due to the fact that decaying oscillations may appear in the solution. It is recommended that a small time is used, similar to the implicit Euler method.

Any of the two methods can be used to update the temperature in the domain, but for grid points that near the interface special care is needed. If the interface passes between two grid points, the stencils of (3.50a) or (3.51a) need to be modified. For example, let us say the interface is between points x_i and x_{i+1} , and the equation for the node x_i is of interest. The solution is not defined across the interface and the values of u_{i+1}^{t+1} need to be valid. This is achieved by producing ghost values (Gibou and Fedkiw, 2005) using extrapolation, from values left of the interface. For example in the case of constant extrapolation, one would take $u_{i+1}^G = \tilde{T}$, where $\tilde{T}(x) = u_l$ represents the extrapolation function and is given by (3.49). Linear extrapolation is given by:

$$\left\{ \begin{array}{l} \tilde{T}(x) = c \cdot x + d \\ \tilde{T}(0) = u_i \\ \tilde{T}(\theta\Delta x) = u_l, \text{ with } \theta = \frac{x_l - x_i}{\Delta x} \end{array} \right.$$

where x_l is the exact location of the interface between two grid points. Cases where the interface crosses between grid points in other spatial directions can be handled in a similar fashion. Constant and linear extrapolations are the simplest to implement but other approximations can be used (quadratic or cubic extrapolation). The stencils can be modified on every point of the domain if needed, by simply modifying the corresponding coefficients in the linear system.

3.4.3.4. Simulation results

In this section simulations for the modified Stefan problem are presented. All simulations have a number of properties in common:

- The temperature is updated using a modified Crank-Nicolson method, using a time step of $\Delta t = 10^{-3}$ seconds, on discrete rectangular domain with $N = 109$ grid points in each spatial direction, yielding a spatial step size of $h = 0.037$.
- For simplicity, the thermal diffusivities are considered to be equal to 1 in both liquid and solid domains.
- The interface temperature is chosen to be $u_l = 0^\circ$ Celsius, which means that as the crystal grows, the solid region will have the same temperature $u_s = 0^\circ$ Celsius. The degree of undercooling is $\Delta u = -0.5^\circ$ Celsius. Also the external boundaries of the rectangular domain are kept at constant temperature equal to the undercooling ($u_{boundary} = \Delta u$ in $\Omega_L(t)$).

The numerical results are shown in *Figs.3.10.-.3.12.*

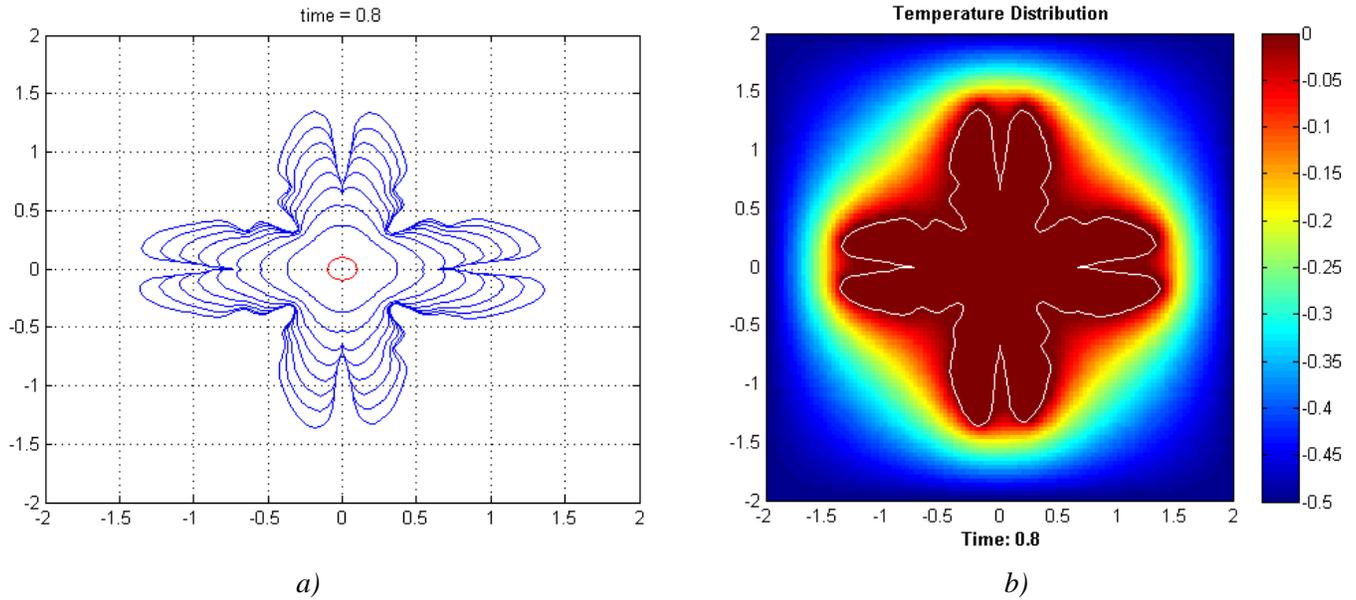


Fig. 3.10. a) Evolution of a circular seed with radius $r = 0.1$ over a time interval $\Delta T = 0.8$ s. The time levels are shown in increments of 0.1 seconds. b) The temperature distribution at final time ΔT

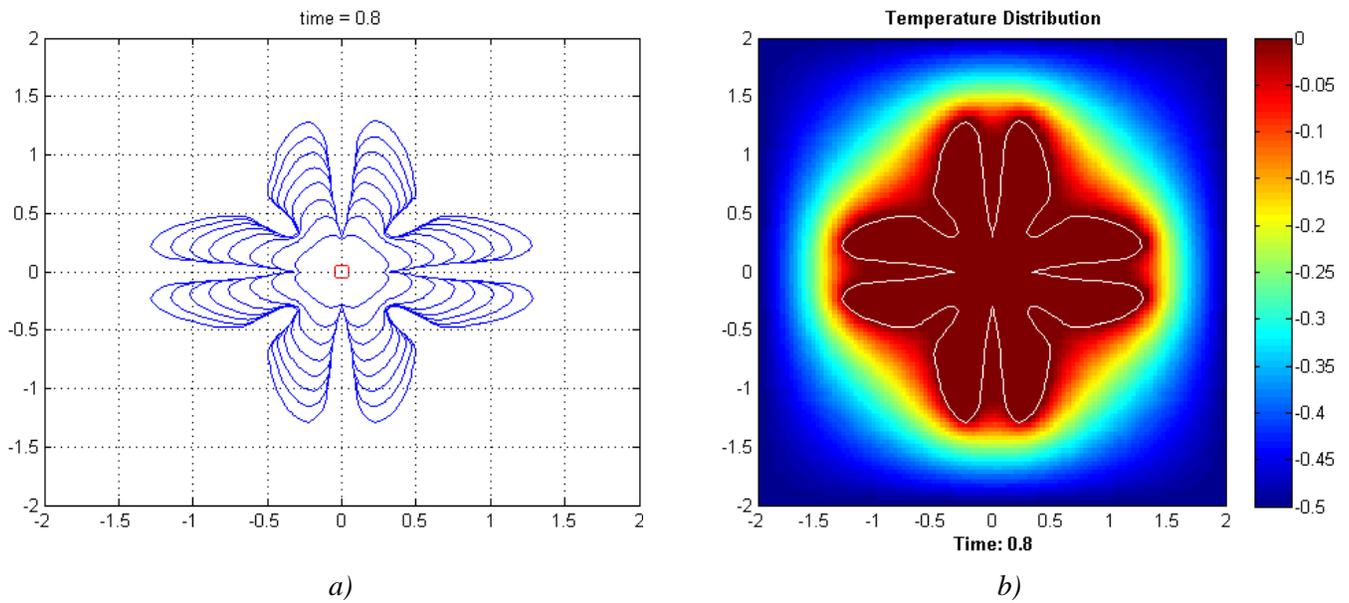


Fig. 3.11. a) Evolution of a square seed with $l = 0.1$ over a time interval $\Delta T = 0.8$ s. The time levels are shown in increments of 0.1 seconds. b) The temperature distribution at final time ΔT .

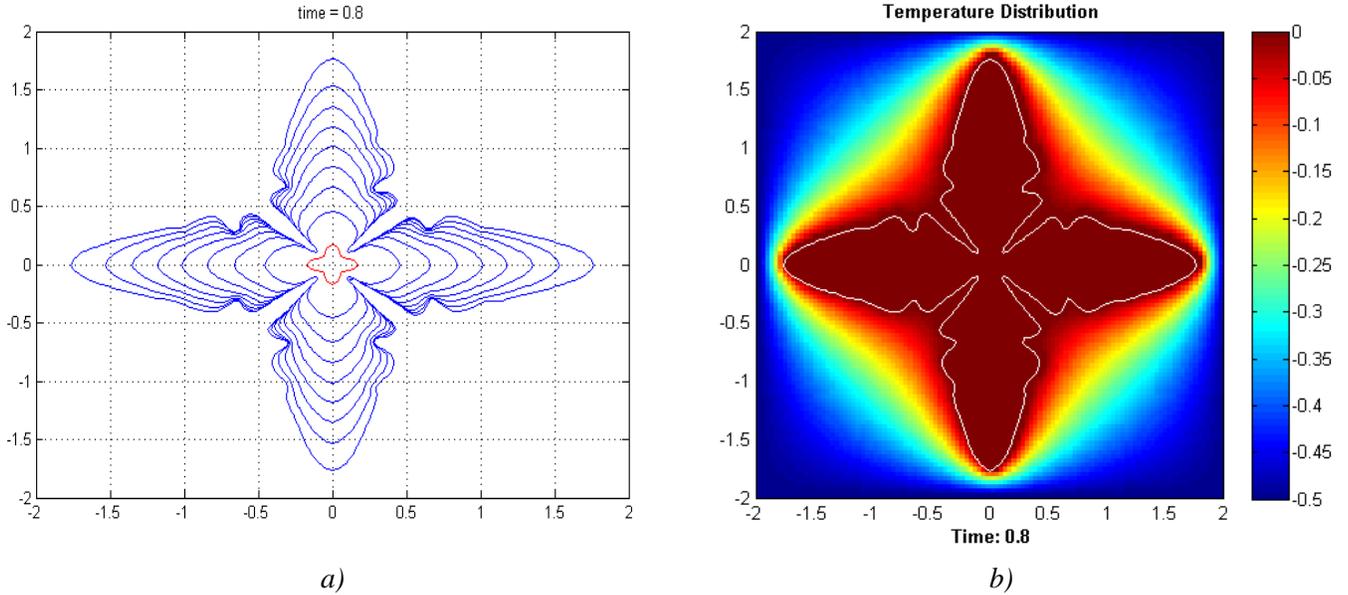


Fig. 3.12. *a)* Evolution of a four sided star shape seed over a time interval $\Delta T = 0.8$ s. The time levels are shown in increments of 0.1 seconds. *b)* The temperature distribution at final time ΔT .

Although, for numerical reasons these simulations were carried out using a simplified solidification model and different initial seeds, the model maintains its essential properties; it exhibits key characteristics of real dendritic growth, such as side branching. The average simulation time for the non-local model in the form of the modified Stefan problem for each particular case is quite high, each simulation taking 7-8 hours using a HPC (high performance computing) cluster.

3.4.4. CML simulation of dendritic crystal growth

3.4.4.1. Model description

Coupled map lattice models provide an alternative means of modeling dendritic growth. As stated in Section 2.3, each CML model consists of a number of elements: the discrete lattice structure, the neighborhood taken into consideration and the transition rule which governs the evolution of the model. In the following, a modified version of the model proposed by Sakaguchi and Ohtaki (1999) is presented. The interface in the model is assumed to be sharp, in the sense that points on the mesh are categorized into occupied and unoccupied cells.

The evolution model under consideration is the two-phase Stefan problem presented in the previous sections, i.e. equations (3.10)-(3.13), where the computational domain is taken to be a square lattice. The

discretization using a CML formulation requires several steps. The CML model is coupled with the heat equation, and has two distinct steps:

- The first step requires solving the diffusion equation over the whole domain:

$$u_{i,j}^{t+1} = u_{i,j}^t + D\{u_{i+1,j}^t + u_{i-1,j}^t + u_{i,j+1}^t + u_{i,j-1}^t - 4u_{i,j}^t\} \quad (3.52)$$

Here D represents the diffusion coefficient, u the temperature and the superscript the time. Equation (3.52) represents the standard five point stencil and is highly biased along the coordinate axes. For this reason a modified form is used, where the representation given by (3.52) is averaged with one given by rotating the axes by 45° , resulting in a nine-point stencil, called the ‘‘isotropic Laplacian’’, which reduces artificial anisotropy:

$$u_{i,j}^{t+1} = u_{i,j}^t D \left[\frac{1}{6} (u_{i-1,j}^t + u_{i,j+1}^t + u_{i+1,j}^t + u_{i,j-1}^t) + \frac{1}{12} (u_{i-1,j-1}^t + u_{i+1,j+1}^t + u_{i+1,j-1}^t + u_{i-1,j-1}^t) - u_{i,j}^t \right] \quad (3.53)$$

- Each node in the mesh has two sets of information embedded: information related to the temperature field and information related to the crystal surface. The second is expressed using a global variable $x_{i,j}$, which can change its state only at unoccupied nearest neighbors of already occupied cells. After this growth step, the temperature is then modified to take into account the effects of the growth pattern. The second step (the transition rule) is given by:

$$\begin{cases} x_{i,j}^{t+2} = x_{i,j}^t + c_1(T_{i,j}^m - u_{i,j}^{t+1}) \\ u_{i,j}^{t+2} = u_{i,j}^{t+1} + c_2(T_{i,j}^m - u_{i,j}^{t+1}) \end{cases} \quad (3.54)$$

Equation (3.54) has effect only on cells that satisfy $u_{i,j} < T_{i,j}^m$, which is considered the melting temperature. If this condition is not satisfied, then $u_{i,j}^{t+2} = u_{i,j}^{t+1}$ and $x_{i,j}^{t+2} = x_{i,j}^t$. The rate of growth is proportional to the difference between the melting temperature $T_{i,j}^m$ and the current temperature $u_{i,j}^{t+1}$. The parameters c_1 and c_2 correspond to a measure of released latent heat with $L = \frac{c_2}{c_1}$. As L increases (the amount of latent heat produced rises) and the diffusion process acts as a restrictive factor for the growth.

3.4.4.2. Model simulations

Simulations were carried out using von Neumann and Moore neighborhoods (*Fig. 2.3.*). The following parameters were used for all simulations:

- the computational domain is formed out of square cells with $N = 300$ points in each spatial direction,
- the melting temperature is $T_{i,j}^m = 0^\circ$ Celsius,

- the initial temperature in the domain is taken $T_0 = -1.5^\circ$ Celsius,
- the diffusivity constant is $D = 0.2$ and the latent heat parameters $c_1 = 0.285, c_2 = 0.7$.
- to ensure that the heat diffusion process is computed properly a time update step of $\Delta t = 10^{-3} s$ was used.
- as an initial condition for growth, a single solid seed is placed at the center of the domain.

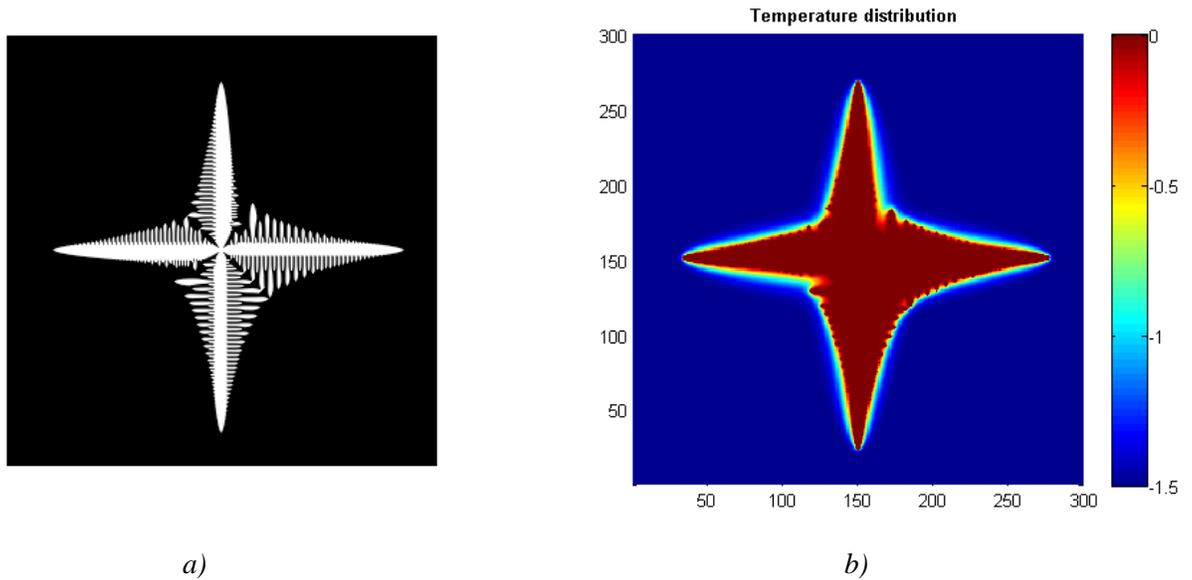


Fig. 3.13. a) Crystal growing in undercooled melt after $\Delta T = 1.5$ seconds using the normal von Neumann capture rule b) Temperature distribution at final time ΔT

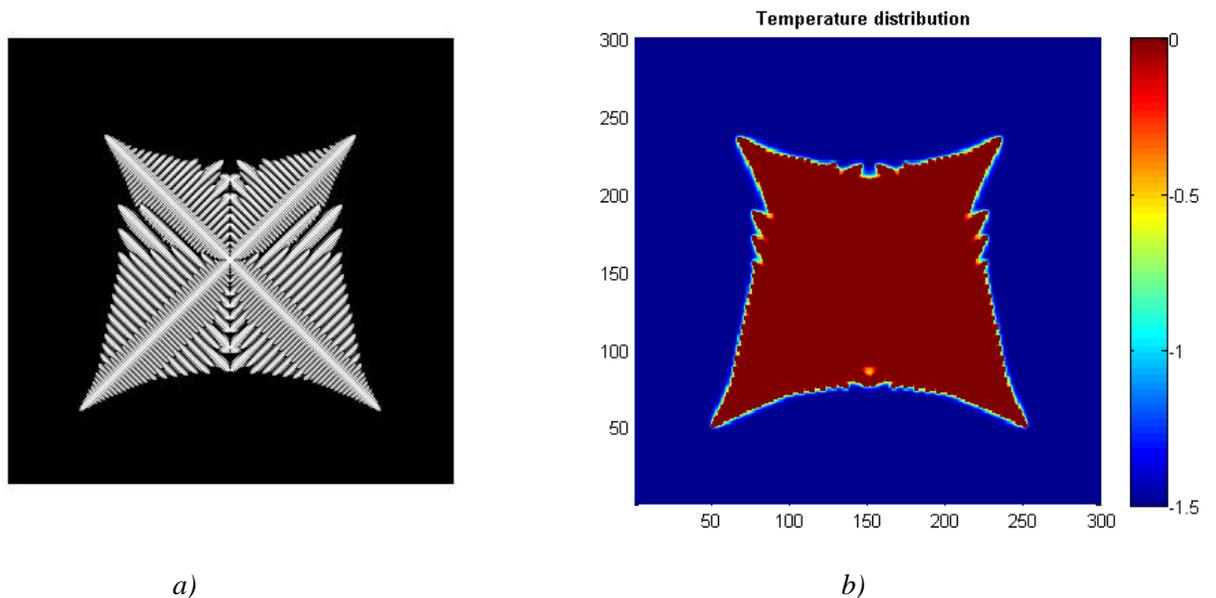


Fig. 3.14. a) Crystal growing in undercooled melt after $\Delta T = 1$ second using the normal Moore capture rule b) Temperature distribution at final time ΔT

As expected, the simulations results for different neighborhoods are different due to the fact that under

the von Neumann rule the crystal presents a growth that tends to favor the axial direction, whereas the Moore rule favors growth patterns in diagonal directions. The simulation results show that CML models possess the ability to reproduce general characteristics of dendritic crystal growth.

3.5. Conclusions

Dendritic crystal structures are found in numerous structures, and have a significant influence over the behavior, and subsequently properties of the system of interest. Various approaches have been developed to study dendritic growth, but the underlying laws that characterize this growth type have yet to be fully understood. Geometrical models of interface evolution provide an excellent candidate to describe the highly nonlinear spatio-temporal dynamics found in dendritic growth, in terms of local properties of the crystal interface. Global parameters, such as the temperature, which are known to have a noteworthy influence in the growth process, can be incorporated into the geometrical laws, extending the capability of developed models. This temperature influence on growth is further studied in upcoming chapters, where the control of the evolving interface is considered in two and three dimensional space. Although more accurate models for the growth process are available, solving the associated equations requires significantly higher computational effort in comparison to geometrical representations.

4. Models of wildfire propagation

4.1. Introduction

Wild land fires are present throughout most geographical regions (wilderness, rural areas etc.) and occur in a highly varied conditions in terms of live/dead vegetation, atmospheric conditions, terrain structure, humidity or other influencing factors. From a systems engineering point of view, wild land fires involve very complex phenomena, typically chemical oxidation reactions, which are usually exothermic in nature, releasing energy in the form of heat, and the transfer of this energy to the nearby unburnt fuel resource, followed by the ignition and combustion processes (Saito, 2001, Ward, 2001). Frequently, this results in extensive damage to both fauna and flora, and, if the fire is within the urban-wildlife border, loss of human lives. Due to the dynamic behavior of wild land fires and the relatively large number of influencing parameters that dictate the fire intensity and rate of spread, the prediction, control, containment and prevention of such processes is not an easy task to accomplish. Having highlighted some of the features of fire spread models, it is straightforward to include wildfire evolution processes within the spatio-temporal class of models.

This chapter provides a survey of the wildfire evolution models, together with analysis and numerical simulations of wind-aided fire growth models. The chapter is organized as follows. Section 4.2 provides a comprehensive review of fire propagation models, with emphasis on wind-aided growth models described by geometrical laws of interface evolution. The third part, Section 4.3 offers a detailed numerical analysis of wind-aided fire spread models, using a series of idealized conditions – topographical and environmental.

4.2. Wind-aided fire evolution models

In this chapter, geometrical models that describe the wind-aided evolution of a single fire front interface generated by a single ignition event are studied. The fire is assumed to evolve in a plane, ignoring fire height, which is usually adequate in observing the burned area perimeter spread over time.

A simple classification of forest fires can be made based on the biofuel type and the fire location relative to the ground (Finney and Ryan, 1995) into surface, crown, spotting and acceleration fires:

- Surface fires – are found in regions where the biofuel does not exceed the height of 2 m, with the usual burning material being shrubs, small trees, grass, leaves etc. It is reasonable

to include in this section the category of ground fires, which consumes fuel found between the mineral soil and dead vegetation area. In this work, the main focus is set on this type of fire spread, since it's the most commonly occurring type of wildfire.

- Crown fires – are usually found in areas with vegetation containing both surface fuel and a canopy layer, and usually develop from surface fires. Depending on the severity of the fire, it may be classed as active (if the surface and canopy vegetation burn alongside) or passive crown fire (if only some of the canopy layer vegetation is affected).
- Spotting fires – are known to occur in regions rich in resin content, and are caused by the fact that burnable materials from the vegetation is transported in the form of embers over some distance away from the initial fire, causing new conflagrations; this makes fire suppression and prediction in such cases a nearly impossible task.
- Acceleration fires – are found in regions with high slopes (valleys, canyons etc.), and develop rapidly due to heat being transported to unburnt fuel in an accelerated manner due to convective currents.

Fire spread modeling approaches can be classified into physical, quasi-physical, empirical, quasi-empirical. Physical fire evolution models are models that describe the fire evolution using a theoretical approach (Fujii et al., 1980, Hwang and Xie, 1984, Weber, 1989). Such models take into account both the laws of physics and chemical reactions that take place during a fire, such as fluid mechanics, combustion, gas matter reactions, heat transfer etc. In comparison with physical models, the quasi-physical models only take into account the physics aspect of the fire evolution problem, whilst overlooking the chemical processes that are present (Wolff et al. 1991, Albini, 1985, Butler et al., 2004 Vaz et al., 2004). Although these models are founded on rigorous theoretical approaches, their validation is problematic, mainly due to the difficulties in the evaluation of individual components that contribute to the fire spread mechanism (Pastor et al., 2003). This shortcoming led to the development of empirical models, which are derived from statistical correlations from data extracted gathered in field or laboratory experiments (Burrows et al., 1991, Alexander et al., 1990, Marsden-Smedley and Catchpole, 1995, Fernandes et al., 2002). Although easier to validate, these models are often only valid in the specific scenarios in which the fire was observed.

This section focuses on quasi-empirical models. As stated from their name, these models are obtained following experimental observations that are linked and analyzed with a physical context that characterizes the fire evolution (Rothermel and Anderson, 1966, Sullivan, 2007a and 2007b). Therefore quasi-empirical models are well suited for characterizing a wide range of fire evolution cases, with the observed data usually driving the model structure (parameters). This property makes the quasi-empirical modelling approach an excellent candidate from a systems engineering perspective in terms of dynamics

analysis and prediction, numerical simulation and development of suppression techniques. Let $\Gamma(s, t)$ be a positively oriented curve (closed and simple planar curve) representing a moving boundary. The time varying curve is defined as a continuous map $\bar{x}_\Gamma: [0, S(t)) \times [0, T] \rightarrow \mathbb{R}^2$, where $\bar{x}_\Gamma(s, t) = (x_1^\Gamma(s, t), x_2^\Gamma(s, t))$ are the Cartesian coordinates of a point on the curve $\Gamma(s, t)$ at a given time, $s \in [0, S(t))$ is the arc length and it is assumed that the fire front is a closed curve at all times, i.e. the condition $\bar{x}_\Gamma(0, t) = \bar{x}_\Gamma(S(t), t)$ holds. The fire interface is defined as the curve $\Gamma(s, t)$ that surrounds the burnt area. Furthermore, the parameterization is chosen such that the unburnt biofuel lies on the right of the interface, in the direction of increasing s (Fig. 4.1).

The time evolution of the fire interface can then be described by the equation

$$\begin{cases} \frac{\partial}{\partial t} \bar{x}_\Gamma(s, t) = F(\bar{u}_\Gamma\{\bar{x}_\Gamma(s, t)\}) \cdot \vec{N} \\ \bar{x}_\Gamma(s, 0) = \Gamma_0 \end{cases} \quad (4.1)$$

where \vec{N} is the outward unit normal, $F(\cdot)$ is a general functional that describes the rate of spread of the fire and $\bar{u}_\Gamma\{\cdot\}$ represents a set of state variables that characterize each point $\bar{x}_\Gamma(s, t)$ along the curve, i.e. the collection of parameters that influence the growth.

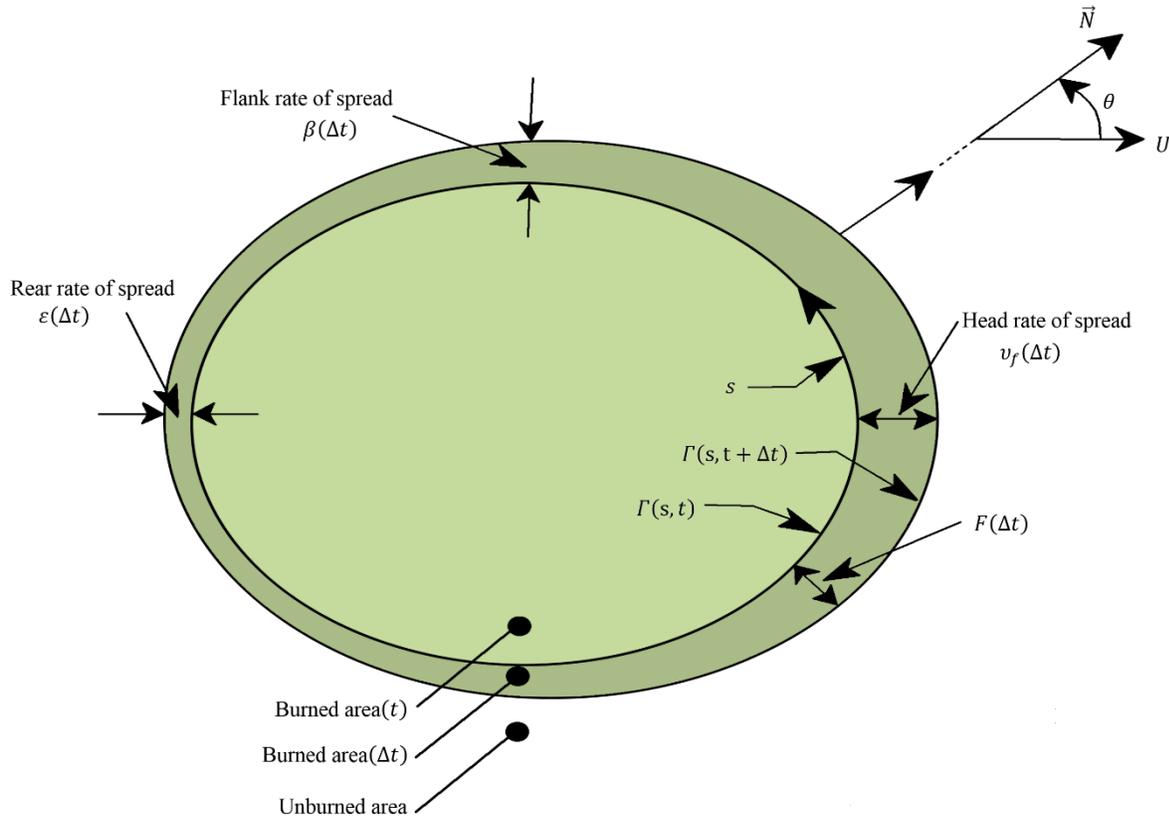


Fig. 4.1. Time variation of contour $\Gamma(s, t)$ in time Δt

One of the first semi-empirical models was proposed by Frandsen (1971) who modelled the fire evolution in the $x - z$ plane by considering a volume element ΔV that is moving at a constant depth in the z plane, from $x = -\infty$ to $x = 0$, where the fire interface is located. He proposed that the rate of spread of the fire front is given by:

$$F = \frac{I_{xig} + \int_{-\infty}^0 \left(\frac{\partial I_z}{\partial z} \right)_{z_c} dx}{\rho_{be} Q_{ig}} \quad (4.2)$$

where I_{xig} represents the horizontal heat flux absorbed at ignition time, ρ_{be} is the bulk density of the available fuel, Q_{ig} denotes the net heat absorbed by the fuel as it nears ignition and the integral term $\left(\frac{\partial I_z}{\partial z} \right)_{z_c}$ is a measure of the gradient of the vertical intensity of the heat flux evaluated at depth z_c of the fuel bed. Equation (4.2) formulates the rate of spread of the fire using the ratio between the heat flux transmitted into the fuel bed (the numerator term) and the heat required for ignition by the unburnt fuel (the denominator term), but doesn't take into account directly any topographical or atmospheric conditions, limiting its use.

Following the model proposed in (4.2), Rothermel (1972) extended Frandsen's approach, introducing a series of new concepts to include the effects of wind, terrain topography and detailed fuel properties into the mechanism of heat propagation, giving the rate of spread:

$$F = \frac{\Gamma' \cdot w_n \cdot h \cdot \left(1 - 2.59 \frac{M_f}{M_x} + 5.11 \left(\frac{M_f}{M_x} \right)^2 - 3.52 \left(\frac{M_f}{M_x} \right)^3 \right) \cdot n_s \cdot \xi \cdot (1 + \Phi_w + \Phi_s)}{\rho_{be} \cdot \varepsilon \cdot (250 + 1.116 M_f)} \quad (4.3)$$

which takes into account numerous parameters of the fire interface dynamics. The rate of spread F of the model (4.3) depends on the following set of variables:

- The optimum reaction velocity Γ' , which indicates the completeness and rate of fuel consumption
- The net fuel loading w_n
- The biofuel heat content h
- A scaling factor determined by the ratio of fuel moisture, M_f and the moisture of extinction M_x
- The mineral damping coefficient n_s
- The propagating flux ratio ξ
- Wind (Φ_w) and slope (Φ_s) contribution factors
- The amount of fuel involved in the ignition process, ρ_{be}
- The ratio between ρ_{be} and the actual amount of available fuel ρ_b , denoted by ε .

A more convenient expression of Rothermel's model can be obtained by introducing a measure of

the heat released per area unit of the fire front I_R , which is proportional to the rate of change of the biofuel from solid state to gas state:

$$F = \frac{I_R \xi (1 + \Phi_w + \Phi_s)}{\rho_{be} \varepsilon Q_{ig}} \quad (4.3a)$$

Here $I_R = -h \frac{dw}{dt}$ represents the reaction intensity, with $\frac{dw}{dt}$ expressing a measure of the mass loss rate per unit area in the fire front, and h has the same meaning as in (4.3) and Q_{ig} is the heat of pre-ignition.

Although written in compact form (4.3a), each term involves a relatively large number of parameters; ultimately eleven parameters are needed to fully describe the evolution of fire interface.

Rothermel's model was proven to be practical, most of its dynamics being consistent with observed wildfire evolution in nature, making it one of the most widely used simulation models and the basis for most of the current fire evolution models (Pastor et. al, 2003). Although the model (4.3) resolves some of the weaknesses of the Frandsen model, it still retains a significant series of shortcomings – in order to accurately describe fire dynamics the number of inputs to the model needs to be relatively large, with most parameters being difficult to estimate or ascertain (e.g. fuel moisture, fuel density, fuel mineral percentage etc.); in addition, the rate of spread is valid only at the head of the interface and cannot predict the evolution at the flanks or at the rear of the fire, meaning the speed functional F does not properly describe the fire dynamics.

Fendel and Wolff (2001) proposed another quasi-empirical approach to model the functional F , based on Rothermel's model. They proposed a speed functional of the form:

$$F = F(U, \theta, m; \dots) \quad (4.4)$$

where now the fire rate of change is dependent on the wind velocity magnitude U , the angle θ between the fire interface normal \vec{N} and the wind direction, measured counterclockwise, the mass loading m of the biofuel and additional parameters, independent of U, θ, m , such as the curvature k of the fire front, as suggested by Osher and Sethian (1998). Although curvature effects have been observed to influence the head of the fire under strong warm winds, these effects are limited to early stages of fire growth and do not hold, in general, at large spatio-temporal scales.

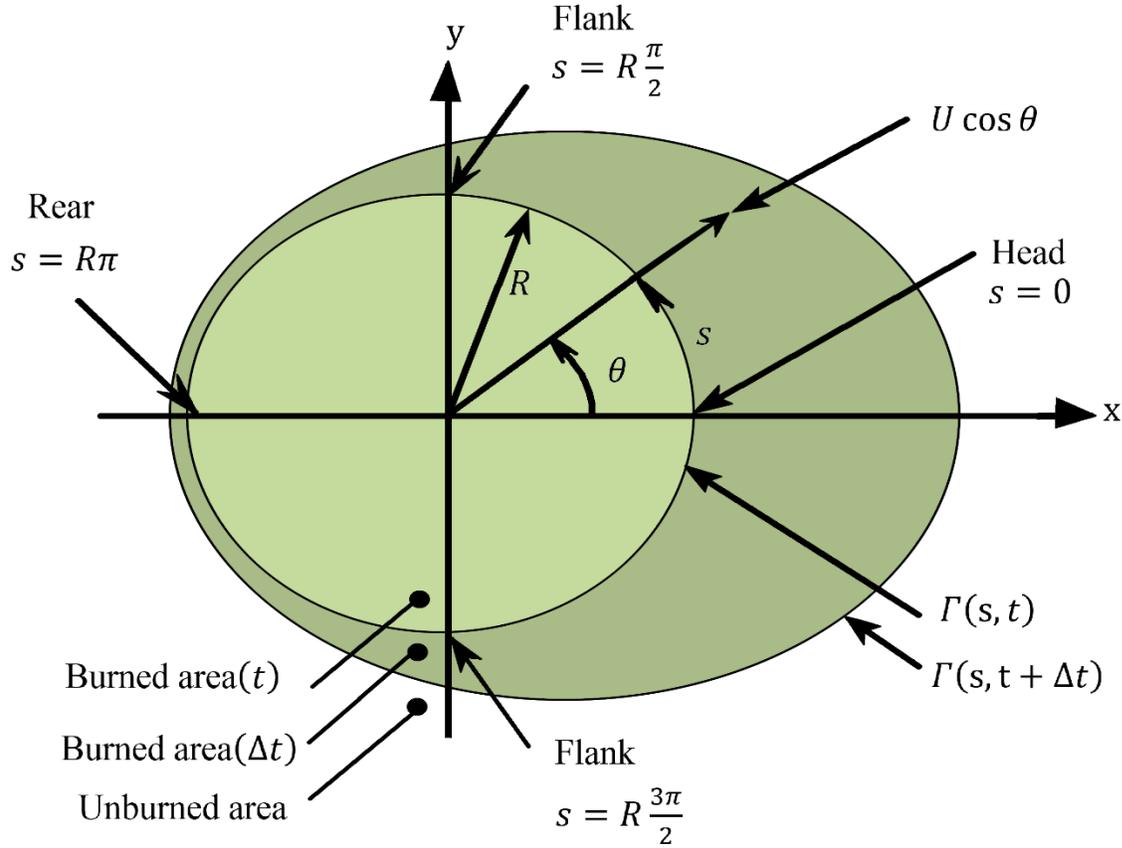


Fig. 4.2. Fire front evolution as a function of wind speed U and wind direction θ . Initial wildfire is considered to be a circle of radius R

The model (4.4) characterizes the time dynamics of the fire interface as a whole by assuming that the main driver of fire evolution is the wind and by taking into account the appropriate parameters distributed at the front, rear and sides of the fire (Fig 4.2). Given the three specific possible wind directions ($\theta = 0$ – downwind, $\theta = \pi$ – upwind and $\theta = \{\frac{\pi}{2}, \frac{3\pi}{2}\}$ at the fire flanks), the speed functional F is taken to be:

$$\begin{cases} F(U, \theta) = v_f(U \cos(\theta)) + \beta(U \sin^\lambda(\theta)), & |\theta| \leq \frac{\pi}{2} \\ F(U, \theta) = \beta(U \sin^\lambda(\theta)) + \varepsilon(U \cos^2(\theta)), & |\theta| > \frac{\pi}{2} \end{cases} \quad (4.5)$$

with $v_f(\cdot)$, $\beta(\cdot)$ and $\varepsilon(\cdot)$ denote the velocities at the front, flanks and rear of the fire interface and the term $\lambda \approx 2 - 4$ is a tuning parameter. The three rates of spread that characterize the model must be consistent with physical observations, in the sense that the rate of spread at $\theta = 0$ should be the dominant term in F , since the conductive and convective heat transfer determined by wind speed is the principal mechanism of fire spread. Also the rate of spread at the rear ($\theta = \pi$) should have the smallest contribution since the preheating effects of the unburnt fuel due to current fire are diminished by the convective cooling effects of the upwind. In general, the following holds:

$$v_f(\cdot) > \beta(\cdot) > \varepsilon(\cdot) \quad (4.6)$$

As a simple choice, with $\lambda = 2$, the model (4.5) becomes:

$$\begin{cases} F(U, \theta) = \{\varepsilon_0 \cos^2 \theta + c_1 \sqrt{U \cos(\theta)}\} + \{\varepsilon_0 \sin^2 \theta + aU \sin^2 \theta \cdot e^{-bU \sin^2 \theta}\}, & |\theta| \leq \frac{\pi}{2} \\ F(U, \theta) = \{\varepsilon_0 \sin^2 \theta + aU \sin^2 \theta \cdot e^{-bU \sin^2 \theta}\} + \{\varepsilon_0 \cos^2 \theta \cdot e^{-\varepsilon_1 U \cos^2 \theta}\}, & |\theta| > \frac{\pi}{2} \end{cases} \quad (4.7)$$

The specific form of $F(U, \theta)$ in (4.7) is obtained using:

$$v_f(U) = \varepsilon_0 + c_1 \sqrt{U} \quad (4.8)$$

$$\beta(U) = \varepsilon_0 + aU \cdot e^{-bU} \quad (4.9)$$

$$\varepsilon(U) = \varepsilon_0 \cdot e^{-\varepsilon_1 U} \quad (4.10)$$

as the wind-speed functions. The model parameters $\Theta = [\varepsilon_0, \varepsilon_1, a, b, c_1]$ do not depend on U or θ but they incorporate properties of the biofuel and fuel bed over which the fire is spreading. One of the major advantages of this model is that it can be easily simulated numerically, and can be readily applied to scenarios where the wind direction is time varying or the distribution of vegetation is spatially varying.

Another recent approach to determining the rate of spread is given by Fernandes (2001). He proposes an exponentially fitted rate of spread, obtained from experimental prescribed fires using different biofuels:

$$F = a \cdot e^{bU} \cdot e^{-cM_d} \cdot h^d \quad (4.11)$$

where U is the wind speed measured at 2m height, M_d represents the percentage of moisture content found in (dead) biofuel, h is a measure of the vegetation height and the terms a, b, c, d were estimated to best fit the observed data. The model takes into account simple biofuel properties and is in good agreement with the fire dynamics determined by wind speed, however no correlation is made between the terrain topography and atmospheric conditions.

An improvement of model (4.4) was proposed by Mallet et. al. (2009) by minimizing the set of parameters needed to describe the spatio-temporal dynamics, while still maintaining the principal features that determine the shape of the fire interface:

$$\begin{cases} F(U, \theta) = \varepsilon_0 + m\sqrt{U} \cdot (\cos\theta)^{\frac{3}{2}}, & |\theta| \leq \frac{\pi}{2} \\ F(U, \theta) = \varepsilon_0[\alpha + (1 - \alpha)|\sin\theta|], & |\theta| > \frac{\pi}{2} \end{cases} \quad (4.12)$$

The reduced parameter set $\Theta = [\varepsilon_0, \alpha, m]$ is given by ε_0 which represents the velocity at the flanks, the ratio $\alpha \in [0,1]$ between the flank and rear velocities and m which is a measure of biofuel density. Besides the parameter reduction, one major difference between model (4.4) and (4.12) lies in the fact that only the head of the fire front is dependent on the wind speed and orientation while the rate of change at the flanks and rear has a smaller impact on the overall expansion of the fire interface. Since (4.12)

contains only three independent parameters it is significantly easier to tune either directly from experimental prescribed fires or through numerical simulation and analysis.

The simplistic model approach allows to incorporate the terrain topography (or other relevant parameters) in a straightforward manner, by adjusting the speed functional F by:

$$F_{slope}(U, \theta, \mu) = F(U, \theta) \cdot e^{\mu} \quad (4.13)$$

where the exponential term e^{μ} is an estimate of the terrain slope. From physical observations (Luke and McArthur, 1978) a simple choice of the exponent in (4.13) can be made by associating positive μ values with positive terrain incline (i.e. the fire front is moving uphill and has an accelerated rate of spread), negative μ values to negative terrain incline (i.e. the fire front is advancing downhill and has a damped rate of spread) and $\mu = 0$ in the case where the terrain is flat.

The main goal of fire evolution models is to predict the highly nonlinear behavior of the spatial spread of the fire interface, under varying scenarios that describe the physical landscape together with environmental atmospheric conditions. Therefore fast and accurate simulations are needed in cases where real time prediction of the evolving interface might aid in deciding the best suppression strategy. Given such constraints, the PDE models presented could hold considerably more amounts of information about the dynamics than would be necessary, given their continuous time-space characteristics. In such cases CA models might be better suited to compute the physical dynamics using discrete sets of data that characterize a series of physical quantities.

Several approaches have been proposed to model and predict forest fires. Most commonly classical CA models with rectangular elements (Karafyllidis and Thanailakis, 1997, Quartieri et. al, 2010) or hexagonal cells (D'Ambrosio et. al, 2006) are studied, but other approaches such as CA structures coupled with Fuzzy cognitive maps (Carvalho et. al., 2006) have been explored. The states of the CA are usually set to be growth dependent factors, such as wind speed, wind direction or terrain topography increasing the capabilities of the CA models. More recently, probabilistic CA models have been proposed (Pak and Hayakawa, 2011, Almeida and Macau, 2011) which provide a fire spread rate that depends on the probabilities assigned to individual cells within the CA structure (the probability of a cell to become burnt, the probability of a burned cell to influence a neighboring unburnt cell, the proportion of vegetation the cell contains etc.).

While the underlying model structure is common (CA modelling approach) the transition rule is often quite different, although it tries to capture the same interface dynamics and is based on similar physical assumptions. In the work of (Karafyllidis and Thanailakis, 2006) the binary CA model is characterized by:

$$\begin{aligned}
S_{i,j}^{t+1} = & S_{i,j}^t + (nH_{i-1,j}S_{i-1,j}^t + wH_{i,j-1}S_{i,j-1}^t + eH_{i,j+1}S_{i,j+1}^t + sH_{i+1,j}S_{i+1,j}^t) \\
& + 0.83(nwH_{i-1,j-1}S_{i-1,j-1}^t + neH_{i-1,j+1}S_{i-1,j+1}^t + swH_{i+1,j-1}S_{i+1,j-1}^t \\
& + seH_{i+1,j+1}S_{i+1,j+1}^t)
\end{aligned} \tag{4.14}$$

The CA states in (4.14) are proportional to the wind velocity magnitude and direction taken in north, south, east, west directions $(n, s, e, w, nw, ne, sw, se)$ and the weights $H_{m,n}$ that linearly approximate the height difference between cells on the CA lattice. Although the states of model (4.14) are two-dimensional, numerical results have been found to be in good agreement with observed forest fires. Furthermore, the addition of wind and terrain effects does not significantly increase the computational time, making the model a good candidate for real time simulations.

A higher dimensional CA is considered in (Quartieri et. al, 2010) where the state dimension is taken to be 4 and the transition rule is given by the number of neighboring cells that are currently burning (N_b):

$$S_{i,j}^t = \begin{cases} 0 - \text{unburnt fuel cell} \\ 1 - \text{not flammable cell} \\ 2 - \text{burning fuel cell} \\ 3 - \text{burned fuel cell} \end{cases} \tag{4.15}$$

$$S_{i,j}^{t+1} = f(S_{i,j}^t, N_b)$$

The term N_b incorporates preconditions of ignition, such that if sufficient neighboring cells are in state $S_{k,l}^t = 2$, then at the next time step the influenced cell will change its state from inactive to active. This effect is similar to the heat transfer generated and passed to the unburnt biofuel by the evolving fire interface as in Rothermel's approach. Similar to (4.15), the statistical approach of (Almeida and Macau, 2011) uses a series of probabilities to determine the state evolution of a cell within the CA lattice:

$$\begin{cases} S_{i,j}^{t+1} = f(S_{i,j}^t, \mathcal{P}_D, \mathcal{P}_B, \mathcal{P}_I) \\ S_{i,j}^t \in [S_0, S_1, S_2, S_3] \end{cases} \tag{4.16}$$

Where $\mathcal{P}_D, \mathcal{P}_B, \mathcal{P}_I$ model the probability of a cell to contain vegetation, the probability of a cell to end its combustion process and the probability of a cell to begin its ignition process with the allowed state values are similar to (4.15).

Although CA models provide a more simplistic approach, in both theoretical and numerical terms, they can be used to obtain relevant descriptions of wildfire dynamics. Although simpler to implement numerically, as stated in the previous chapters, CA models are usually affected by anisotropy at small lattice resolutions, and provide inaccurate results. This results in an additional term that needs to be taken into account, which leads to increasing the system complexity. One major advantage still remains the diversity in constructing the transition rule, which provides CA with a great flexibility in adjusting the time dynamics of the fire front. However most parameters that influence forest fires are continuous in

nature, and significant information is lost through the discretization effects, typical to CA models, described in the second chapter of this thesis.

4.3. Numerical studies of wildfire propagation

This section is focused on the numerical discretization and simulation of geometric models that describe the evolution of the wildfire fronts, taking into account time and space varying parameters such as fuel density, wind speed and wind direction. As a consequence, the rate of spread of the interface, i.e. the speed functional (4.1) has to be calculated at every point on the fire interface and then used to evolve the front in time.

Various numerical methods have been used to simulate wildfire dynamics. The main approaches used to this means are cellular automation (Muzy et. al., 2005), ellipse reconstruction (Anderson et. al, 1982, Finney et. al, 1994), the finite element method (Linn et. al, 2000 and 2002), marker particle method (Filippi et. al, 2010), volume of fluid method (Hirt and Nichols, 1981) and the level set method (Mallet et. al., 2009).

The volume of fluid method, is one of the most powerful methods for tracking a free evolving interface. The front Γ is represented on a grid where cells are assigned real values from the interval $[0,1]$, such that the value 0 is allocated to cells outside Γ , 1 is allocated to cells that are entirely located inside the contour defined by Γ and estimated 'volumes' between 0 and 1 are associated with grid cells through which the interface passes. After this embedding is complete, the front is advected using a speed functional, and the volume percentages are then updated for all grid cells, giving the new position of the interface. Although widely used in computational physics the method has a series of disadvantages over the previously mentioned the level set method (Osher and Fedkiw, 2003), such as difficulties in handling multiple velocity fields or handling topological changes of the interface (merging/braking). Therefore the level set method will be the method of choice to simulate the geometrical models of wildfire evolution.

More specifically, the representation of the rate of spread of the wildfire interface given by (4.13) will be used in the numerical analysis presented here.

4.3.1. Simulation of geometric models of wildfire evolution

The motion of the fire front interface in the normal direction is given by the model (4.1)

$$\begin{cases} \frac{\partial}{\partial t} \bar{x}_\Gamma(s, t) = F(\bar{u}_\Gamma\{\bar{x}_\Gamma(s, t)\}) \cdot \vec{N} \\ \bar{x}_\Gamma(s, 0) = \Gamma_0 \end{cases}$$

with a specific choice of the speed functional F illustrated in (4.13)

$$\begin{cases} F(U, \theta, \mu) = \left(\varepsilon_0 + m\sqrt{U} \cdot (\cos\theta)^{\frac{3}{2}} \right) \cdot e^\mu, & |\theta| \leq \frac{\pi}{2} \\ F(U, \theta, \mu) = (\varepsilon_0[\alpha + (1 - \alpha)|\sin\theta|]) \cdot e^\mu, & |\theta| > \frac{\pi}{2} \end{cases}$$

The representation of interface motion in the normal direction, described in the previous chapter, was shown to be governed by the well-known level set equation:

$$\varphi_t + V|\nabla\varphi| = 0 \quad (4.17)$$

In the following sections let us consider the level set function φ is defined in a generic bounded subdomain $D \subset \mathbb{R}^2$, $D: [-l, l] \times [-w, w]$, $\{l, w\} \in \mathbb{R}$ represented on a uniformly spaced Cartesian grid. The evolving fire interface is denoted by $\Gamma(t)$ and let $\varphi: D \subset \mathbb{R}^2 \times [0, T] \rightarrow \mathbb{R}$ be its associated level set function, constructed using (3.18). By combining (4.13) with (4.17) the level set representation of the evolving fire interface is obtained as:

$$\begin{cases} \varphi_t + e^\mu \cdot \left(\varepsilon_0 + m\sqrt{U} \cdot (\cos\theta)^{\frac{3}{2}} \right) \cdot |\nabla\varphi| = 0, & |\theta| \leq \frac{\pi}{2} \\ \varphi_t + e^\mu \cdot (\varepsilon_0[\alpha + (1 - \alpha)|\sin\theta|]) \cdot |\nabla\varphi| = 0, & |\theta| > \frac{\pi}{2} \end{cases} \quad (4.18)$$

with the parameters $U \in \mathbb{R}^+$, $\theta \in [-\pi, \pi]$ and $\mu \in \mathbb{R}$.

In the level set equations (4.18) the Hamiltonian given by $H(\varphi_x, \varphi_y) = F(U, \theta, \mu) \cdot |\nabla\varphi|$ needs to be numerically approximated such that the discretized Hamiltonian $\hat{H}(\varphi_x^+, \varphi_x^-; \varphi_y^+, \varphi_y^-)$ is consistent, in the sense that $\hat{H}(\varphi_x^+, \varphi_x^-; \varphi_y^+, \varphi_y^-) = H(\varphi_x, \varphi_y)$. This can be achieved by using various numerical approximation methods (Lax-Friedrichs, Roe-Fix, Sethian-Osher, Godunov). However, due to its generality and robustness, the Godunov scheme is the most suited to approximate $\hat{H}(\varphi_x^+, \varphi_x^-; \varphi_y^+, \varphi_y^-)$ (Osher and Fedkiw, 2003). The information contained in $\hat{H}(\varphi_x^+, \varphi_x^-; \varphi_y^+, \varphi_y^-)$ is then used to advance the front in time using a third order accurate TVD-RK method to obtain the approximation of the level set function φ at time $t + \Delta t$. Special care needs to be taken in ensuring stability of the numerical integration resulting in the following CFL time step restriction needs to be satisfied:

$$\Delta t \left(\frac{|H_1|}{\Delta x} + \frac{|H_2|}{\Delta y} \right) < 1 \quad (4.19)$$

where $H_1 = \frac{F(U, \theta, \mu)\varphi_x}{|\nabla\varphi|}$, $H_2 = \frac{F(U, \theta, \mu)\varphi_y}{|\nabla\varphi|}$ and the partial derivatives of the level set function, φ_x, φ_y are computed using a second order central difference stencil.

Given that the front shape is dependent on the zero level set of the level set function φ , special care needs to be taken after the time integration of (4.18), to ensure the interface is tracked correctly. One of the most common causes for inaccuracy is loss of the signed distance property (i.e. $|\nabla\varphi| \neq 1$) in time.

To resolve this issue, the numerical re-initialization method described in the previous chapter is used to re-distance the level sets of the degraded level set function φ , so that $|\nabla\varphi| = 1$ holds.

Although (4.18) describes the level set representation of the wind aided rate of spread of a wildfire, the numerical implementation is not straightforward. The wind direction angle θ , representing the angle measured in the anti-clockwise direction between the positive Ox axis and the interface normal (Fig.2) is not directly measurable.

Recalling that the interface normal can be written in terms of the level set function as

$$\vec{N} = \frac{\nabla\varphi}{|\nabla\varphi|} = \frac{\varphi_x}{\sqrt{\varphi_x^2 + \varphi_y^2}} \vec{i} + \frac{\varphi_y}{\sqrt{\varphi_x^2 + \varphi_y^2}} \vec{j} \quad (4.20)$$

the inner product between the Ox component of the wind speed, $U_x \vec{i}$ and the interface normal \vec{N} can be evaluated through:

$$\langle U_x \vec{i}, \vec{N} \rangle = \|U_x\| \cdot \|\vec{N}\| \cos \theta \quad (4.21)$$

giving

$$\cos \theta = \frac{\langle U_x \vec{i}, \vec{N} \rangle}{\|U_x\| \cdot \|\vec{N}\|} \quad (4.22)$$

By replacing the respective values in (4.22) it follows that:

$$\cos \theta = \frac{\varphi_x}{\sqrt{\varphi_x^2 + \varphi_y^2}} \quad (4.23)$$

which provides a direct relationship between the level set function φ and the cosine of θ .

Evaluating magnitude of the cross product between $U_x \vec{i}$ and \vec{N} results in:

$$\|U_x \vec{i} \times \vec{N}\| = \|U_x\| \cdot \|\vec{N}\| \sin \theta \quad (4.24)$$

$$\sin \theta = \frac{\|U_x \vec{i} \times \vec{N}\|}{\|U_x\| \cdot \|\vec{N}\|} \quad (4.25)$$

which after replacing terms in (4.24) and rearranging gives

$$\sin \theta = \frac{\varphi_y}{\sqrt{\varphi_x^2 + \varphi_y^2}} \quad (4.26)$$

Substituting (4.22) and (4.26) into (4.18), results in a direct correlation between the time evolution of level set function associated with the wildfire interface, φ_t in terms of its first order partial derivatives, φ_x and φ_y and the speed functional $F(U, \theta, \mu)$ as :

$$\varphi_t = \begin{cases} -e^\mu \cdot \left(\varepsilon_0 + m\sqrt{U} \cdot \left(\frac{\varphi_x}{\sqrt{\varphi_x^2 + \varphi_y^2}} \right)^{\frac{3}{2}} \right) \cdot |\nabla\varphi|, & |\theta| \leq \frac{\pi}{2} \\ -e^\mu \cdot \left(\varepsilon_0 \left[\alpha + (1 - \alpha) \left| \frac{\varphi_y}{\sqrt{\varphi_x^2 + \varphi_y^2}} \right| \right] \right) \cdot |\nabla\varphi|, & |\theta| > \frac{\pi}{2} \end{cases} \quad (4.27)$$

which can now be discretized and implemented numerically, provided the input parameter set given by $\Theta = [\varepsilon_0, \alpha, m, \mu]$ and the initial shape of the fire front, Γ_0 are known.

Within the following subsections, the same notations provided in the previous chapter are used for the numerical approximation of discrete partial/full derivatives of the level set function. In most simulations the spatial domain D is taken to be the square centered at point $O(0,0)$ on a Cartesian grid, with $D: [-40,40] \times [-40,40]$ and $N = 800$ points in each spatial direction, giving a grid spatial resolution of $\Delta x = \Delta y = 0.1$. The initial shape of the fire front is taken to be a circle of radius $R \in \mathbb{R}^+$, centered at the point $C_0(x_0, y_0)$ of the computational Cartesian grid (Fig. 4.3). As the interface is advanced in time, the level set function degrades from a signed distance function and after every $N_{it} = 5$ iterations, the re-initialization algorithm is used to address this issue.

4.3.1.1. Basic wildfire evolution

First a basic evolution is addressed, under constant wind speed, wind speed direction and biofuel distribution. To this aspect, the parameter vector $\Theta = [\varepsilon_0, \alpha, m]$ as $\varepsilon_0 = 0.2$, $\alpha = 0.5$ and $m = 0.5$ with the constant wind speed $U = 10$. The wind speed direction is taken to be constant, with the wind blowing in the S-N direction. The initial fire front is set to be a circle of radius $R = 5$, centered at $C_0(0, -10)$. The interface then is evolved using the fixed set parameters for the time period $T = [0,3]$ and snapshots of the interface are taken at constant time intervals $\Delta t_{plot} = 0.25$ (Fig. 4.4). As regions inside the fire perimeter are considered to have depleted the biofuel resource (and therefore cannot sustain burning), the total area enclosed by the wildfire grows with time, with the growth being stronger in the North direction due to wind effects (Fig. 4.5 , Fig 4.6).

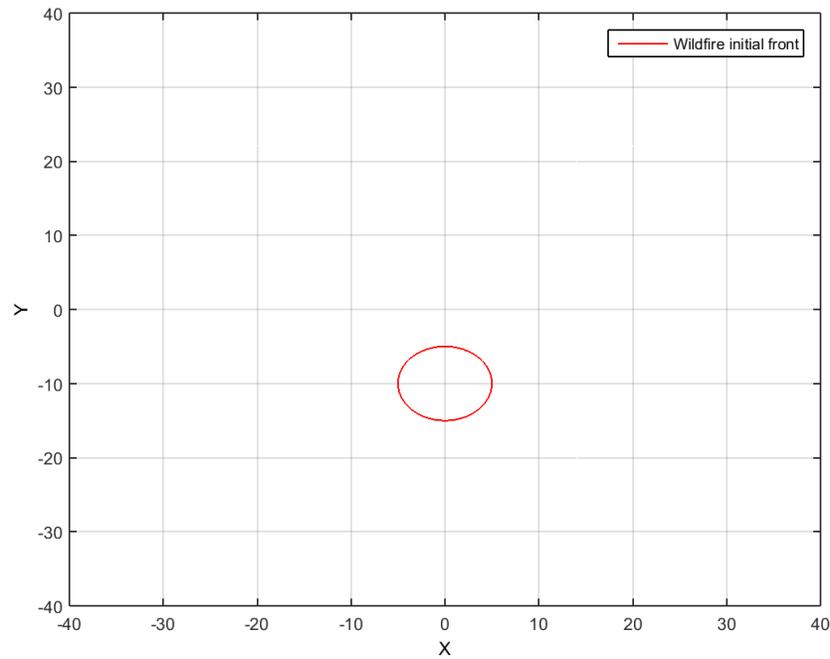


Fig. 4.3. Fire front evolution initial boundary and computational domain

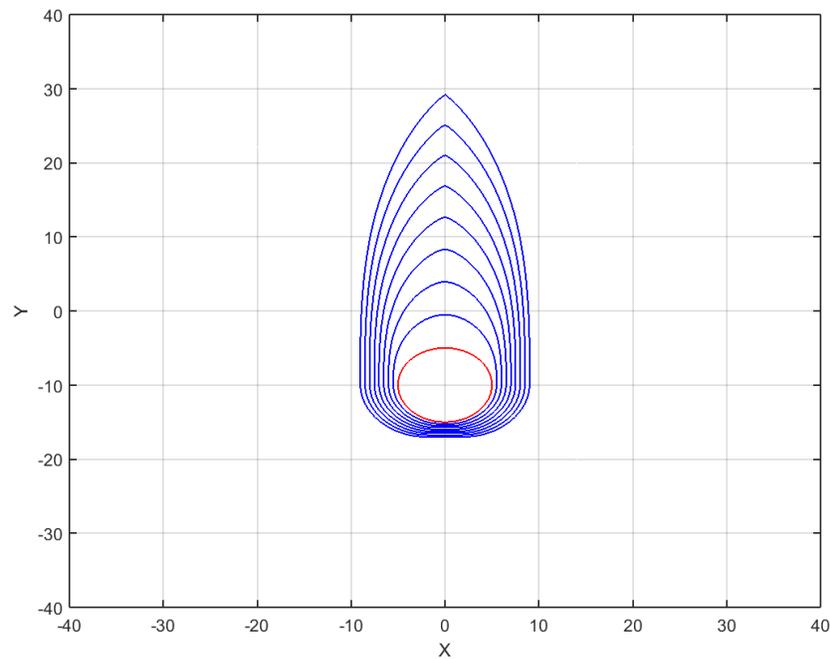


Fig. 4.4. Fire front evolution of a circular initial fire front (red), under $\Theta = [0.2, 0.5, 0.5]$ with constant wind speed $U = 10$ blowing in the S-N direction. The time level sets are shown in increments of

$$\Delta t_{plot} = 0.25$$

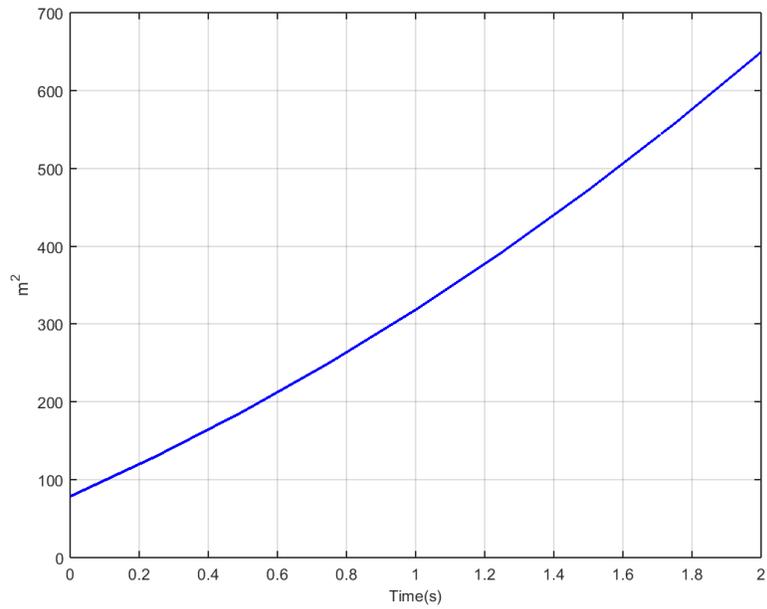


Fig. 4.5. Time evolution of the area enclosed by the fire front under constant wind speed

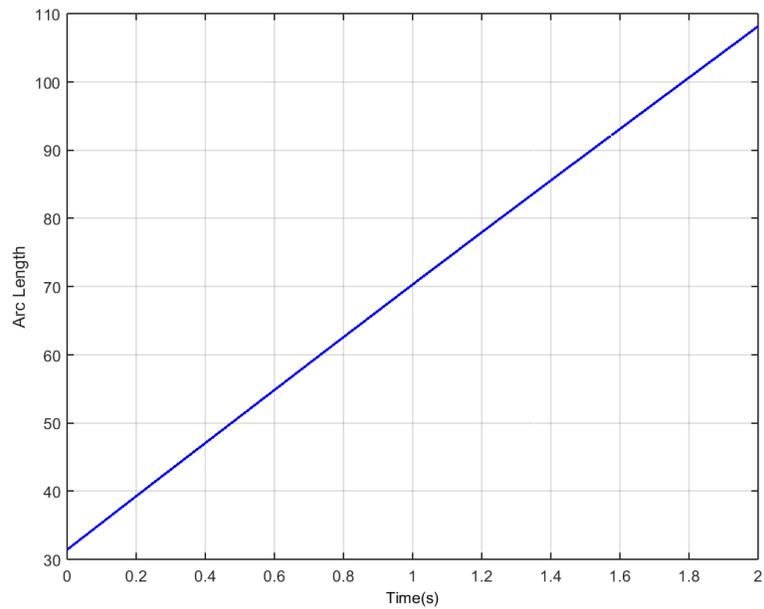


Fig. 4.6. Time evolution of fire front arc length under constant wind speed

4.3.1.2. Evolution under wind speed magnitude and wind direction variation

A more representative scenario can be set by varying the wind speed magnitude U and the wind direction θ . Usually both are time varying and significantly influence the rate of spread of the ‘head’ of

the fire front. Though the amount described by U can be directly incorporated into (4.27), introducing arbitrary angular increments $\Delta\theta$ to θ , independent of other parameters requires the following conversion:

$$r = \sqrt{\varphi_x^2 + \varphi_y^2} \quad (4.28)$$

$$\theta' = \arctan\left(\frac{\varphi_y}{\varphi_x}\right) \quad (4.29)$$

$$\theta^{t+\Delta t} = \theta' + \Delta\theta \quad (4.30)$$

$$\begin{cases} \varphi_x = r \cos(\theta^{t+\Delta t}) \\ \varphi_y = r \sin(\theta^{t+\Delta t}) \end{cases} \quad (4.31)$$

which is the well-known direct and inverse coordinate transformation from Cartesian coordinates to polar coordinates. The effects of varying wind speed are shown in Fig. 4.7 – Fig 4.9. The wind direction is set to be S-N, with the wind speed taking discrete, increasing values in the interval $[10,90]$ in increments of $\Delta U = 10$ at every $\Delta t_{wind} = 0.25$.

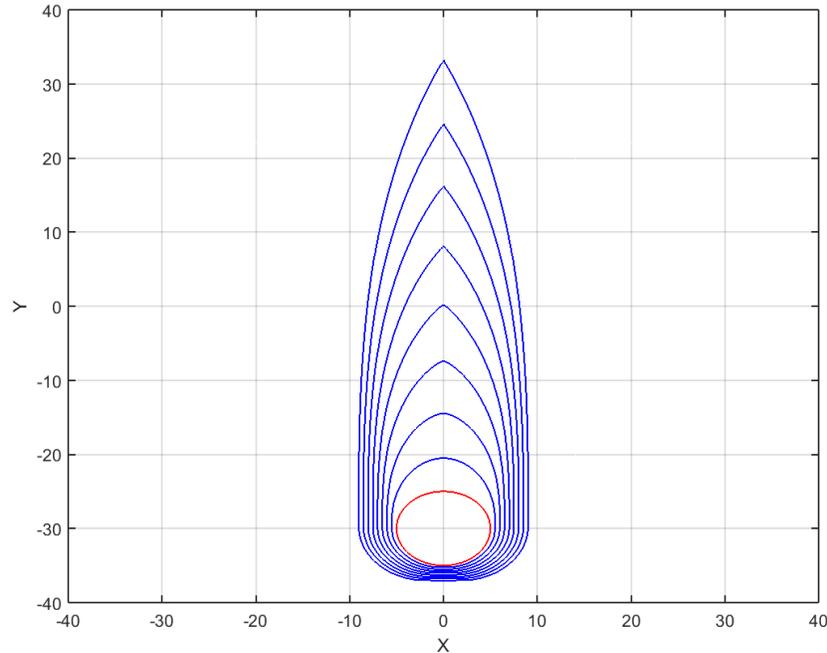


Fig. 4.7. Time evolution of fire front under changing wind speed

Observed wind speed and wind direction during a prescribed fire are usually not constant throughout the experiment, which is consistent with observations of real wildfire scenarios. This behavior is exemplified in Fig. 4.10 – Fig. 4.12, where in addition to the gradual increase in wind speed magnitude, the wind direction progressively changes from a south-north direction to an east-

west direction, which can be clearly seen from the change in the fire front head. As expected, under more aggressive wind conditions, the fire front spreads faster, covering a broader area.

4.3.1.3. Fire islands and multiple fire fronts

Spotting effects in wildfires are commonly observed, although prediction and prevention of such events is an extremely difficult task. Spotting fires that start near the original surface fire front usually develop into similar surface fires. As the fires grow, they can merge, creating a single wildfire interface. Given that initially these fire fronts are not linked directly, islands of unburnt biofuel can be formed within the wildfire perimeter, but in time they are consumed by the fire. Simulations shown in Fig.4.13 illustrates the evolution of multiple fire fronts that evolve together. The initial fire interfaces are set to be circles of radii $R_1 = 8, R_{2,3,4} = 5$ centered at $C_1(-15, -20), C_2(0, -30), C_3(0, -5)$ and $C_4(10, -15)$. The results shown in Fig. 4.14 illustrate the behavior of multiple fire fronts that have unburnt islands. The initial fronts are taken to be circles with radii $R_1 = 5, R_2 = 10, R_3 = 5$ with centres at $C_1(-20, -20), C_2(0, -20), C_3(0, -5)$. A circular region of unburnt fuel with $R_{ub} = 4$ and $C_{ub}(4, -20)$ is placed inside the second fire. Similar initial conditions to previous simulations are used (fuel loading, flank velocity, constant wind speed), but with several independent fire fronts evolving simultaneously. After a period of time, the fronts merge together, demonstrating the capability of the level set method to deal with multiple fire fronts in a straight forward manner.

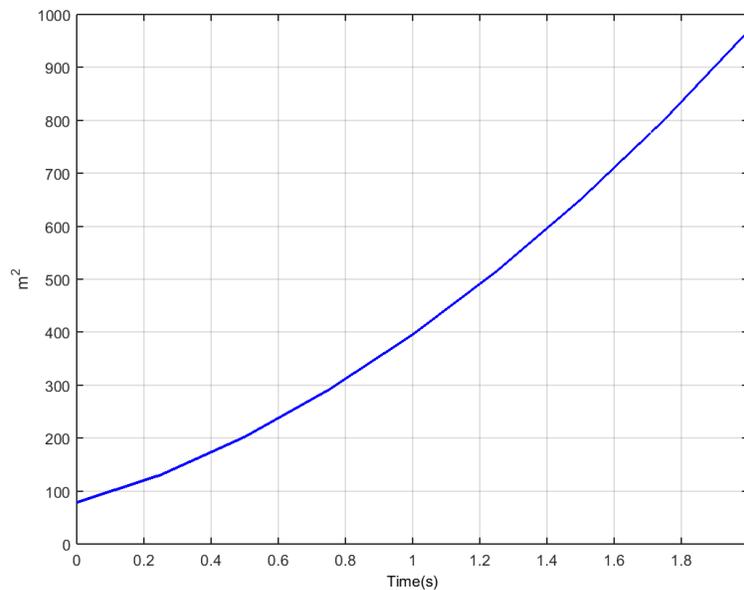


Fig. 4.8. Time evolution of the area enclosed by the fire front under changing wind speed

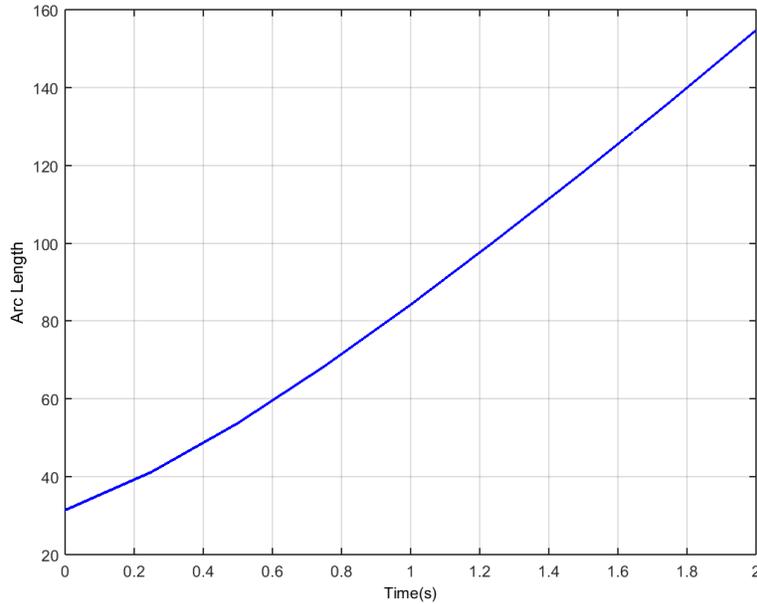


Fig. 4.9. Time evolution of the fire front arc length under changing wind speed

4.3.1.4. Effects of fuel density on wildfire propagation

Fuel concentration is one of the key influencing parameters in the spread of wildfires, and often the distribution is not constant within an observed region. To illustrate this effect a series regions of constant biofuel density were considered (Fig. 4.15a), with variations from low fuel density ($m = 0.1$), medium density ($m = 0.5$) and high fuel density ($m = 1$) defined. The initial fire front configuration is given by a series of spatially distributed wildfires superimposed over a terrain with the previously defined fuel distribution (Fig. 4.15b). The initial fronts are taken to be circles with radii $R_1 = 5, R_2 = 10, R_3 = 5$ with centres at $C_1(-20, -20), C_2(0, -20), C_3(10, -5)$. The multiple fire fronts are shown to evolve at different rates, under the influence of constant wind speed. As the wildfire passes through each fuel density region, it can be seen that the fuel loading greatly impacts the rate of spread (Fig. 4.16).

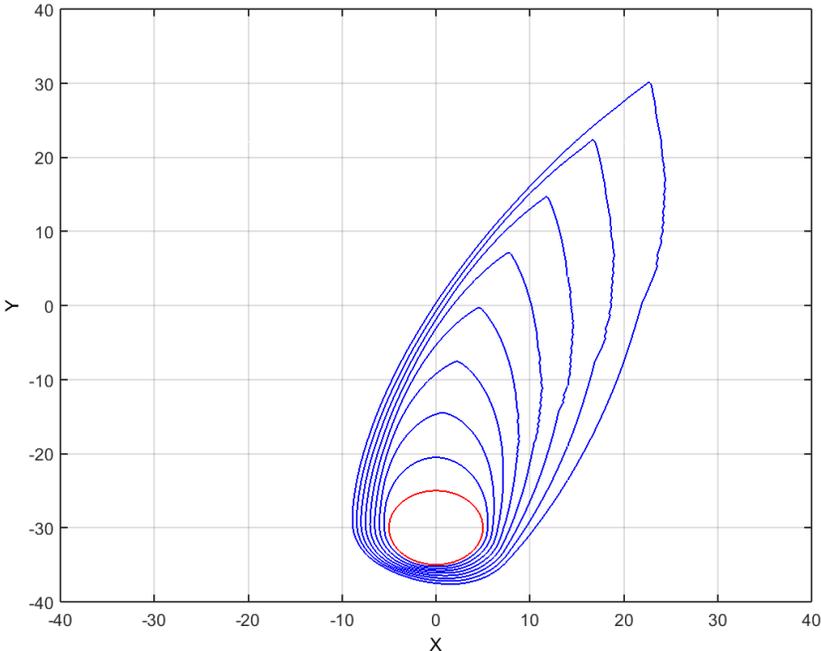


Fig. 4.10. Time evolution of fire front under changing wind speed and wind direction

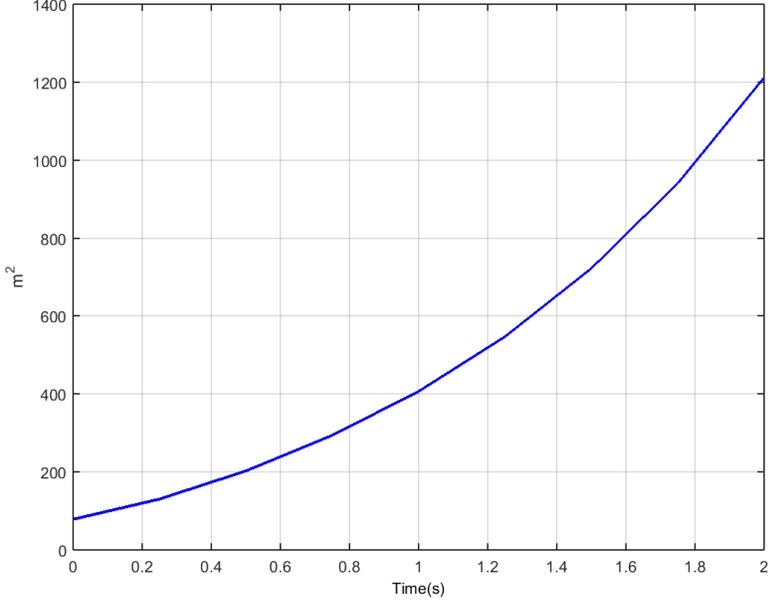


Fig. 4.11. Time evolution of the fire front area under changing wind speed and wind direction

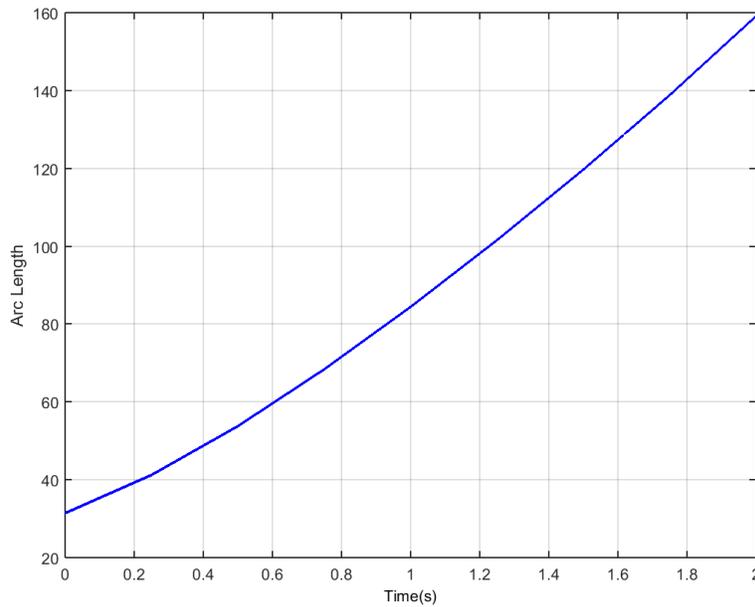


Fig. 4.12. Time evolution of the fire front arc length under changing wind speed and wind direction

4.3.1.5. Terrain elevation effects on wildfire propagation

Terrain topography usually influences the behavior wind aided wildfires by altering the rate of spread under uphill, downhill or flat terrain conditions. Usually, the fire front propagates at a higher rate under higher positive terrain gradient with backwind and is known to be slowed down under negative gradient under the same conditions. An idealized terrain consisting of 3 different terrain inclinations were considered, representing an undulating terrain: a flat ground region followed by a hill area (both up and downhill, with no plateau region at the peak) followed by a flat surface (Fig. 4.17). The initial fronts are taken to be circles with radii $R_1 = 5$ and $R_2 = 3$ with centres at $C_1 = (-10, -25)$ and $C_2 = (-10, -20)$. Under constant fuel loading, and wind speed magnitude and direction (S-N), the wildfire is evolving at higher rates in regions where the slope is taken to be positive (the distance between time consecutive level sets is greater) and has a damped evolution in regions with a negative slope (Fig. 4.18). It is important to notice that although the wind component amplifies or inhibits the wildfire evolution mainly at the head of the fire, the rate of spread at the flanks can still have noticeable effects.

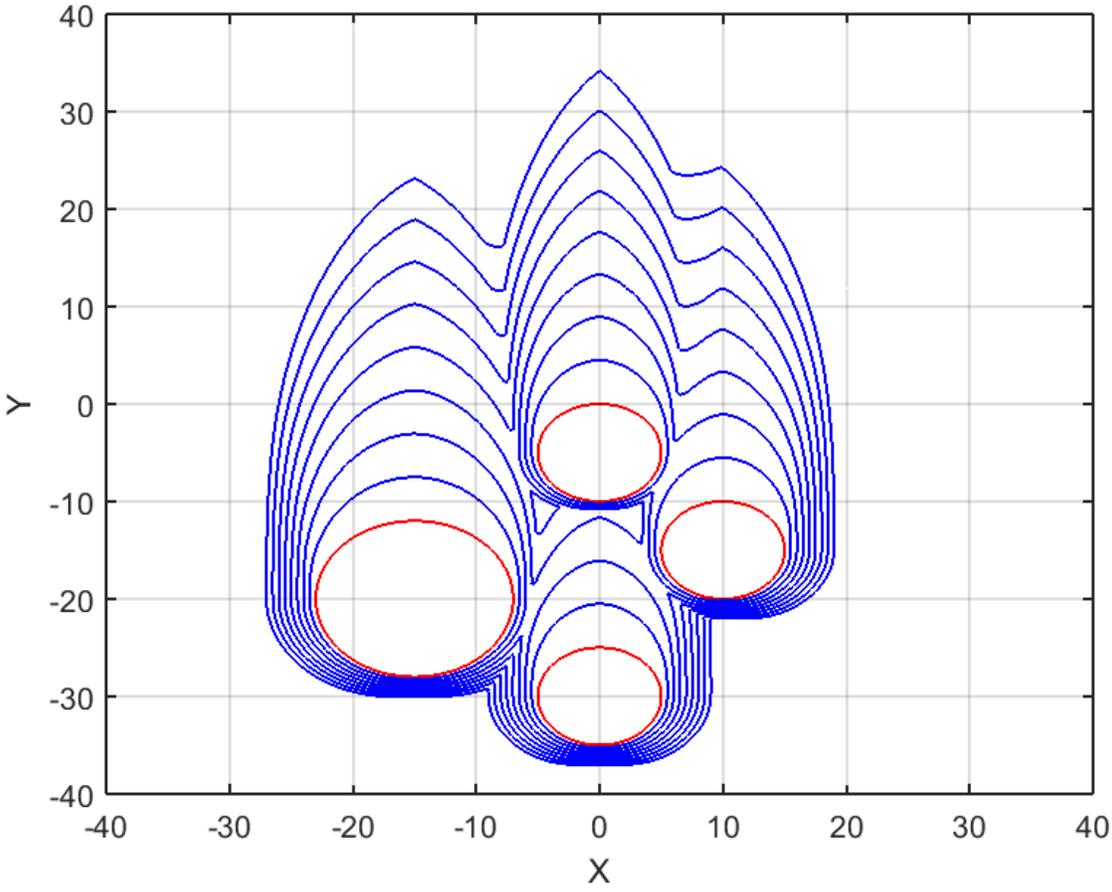


Fig. 4.13. Time evolution of multiple independent fire fronts

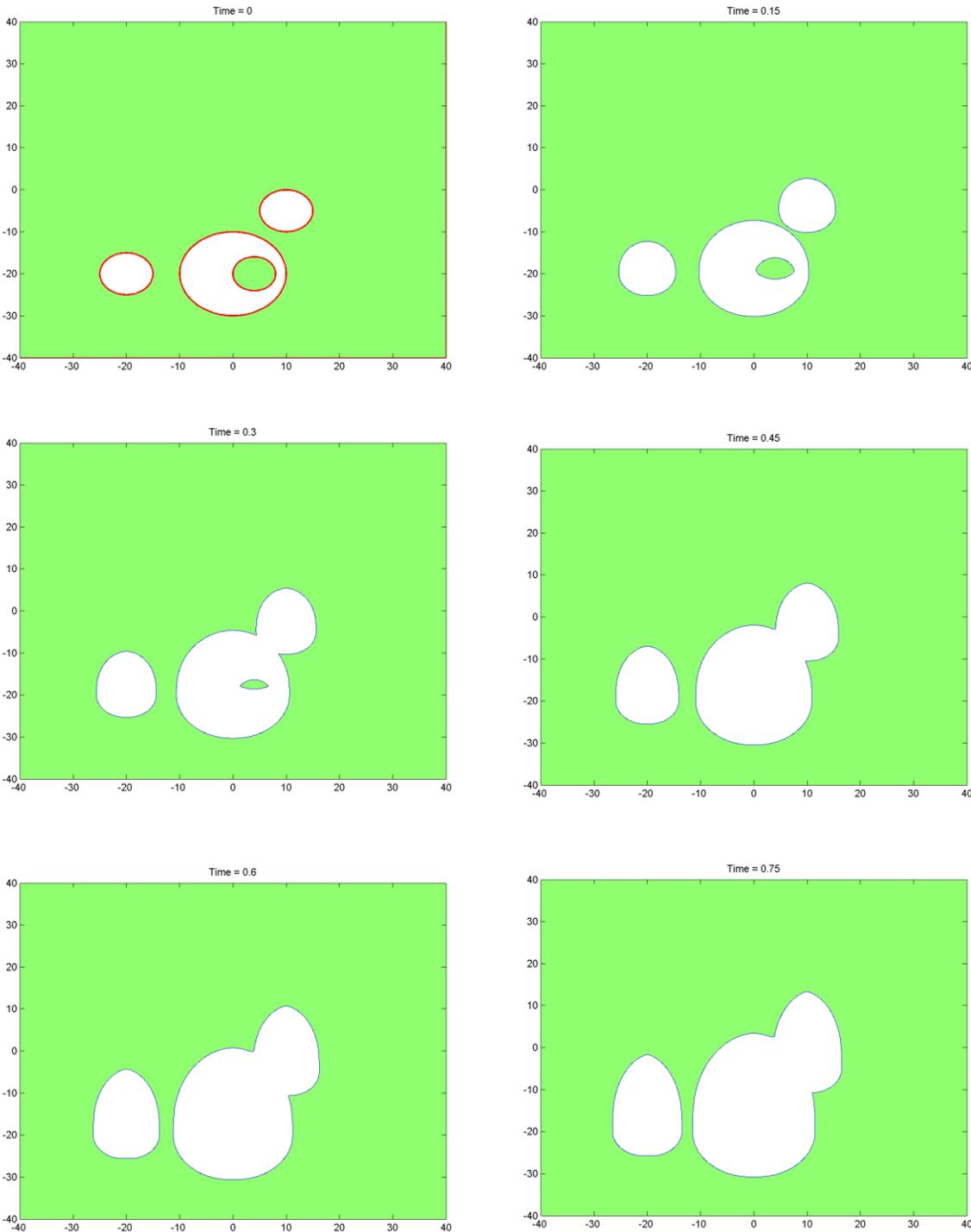


Fig. 4.14. Time evolution of multiple fire fronts with unburnt biofuel islands

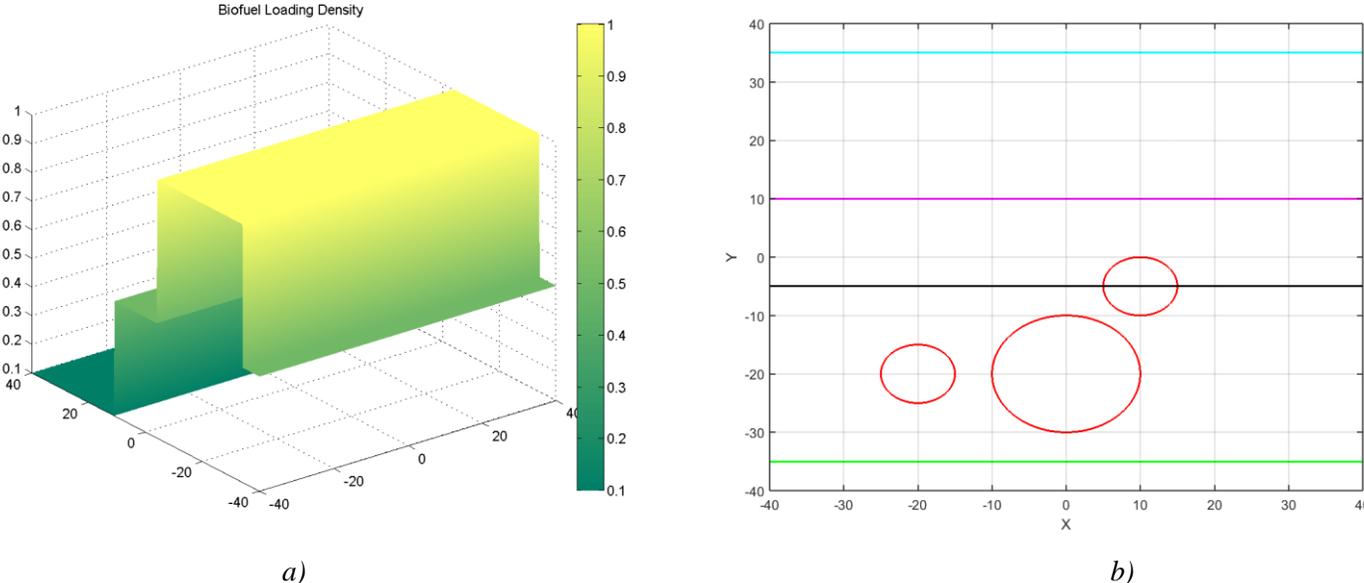


Fig. 4.15. a) Biofuel density b) Initial wildfires under varying fuel density terrain – green, black, magenta, cyan regions are defined according to a)

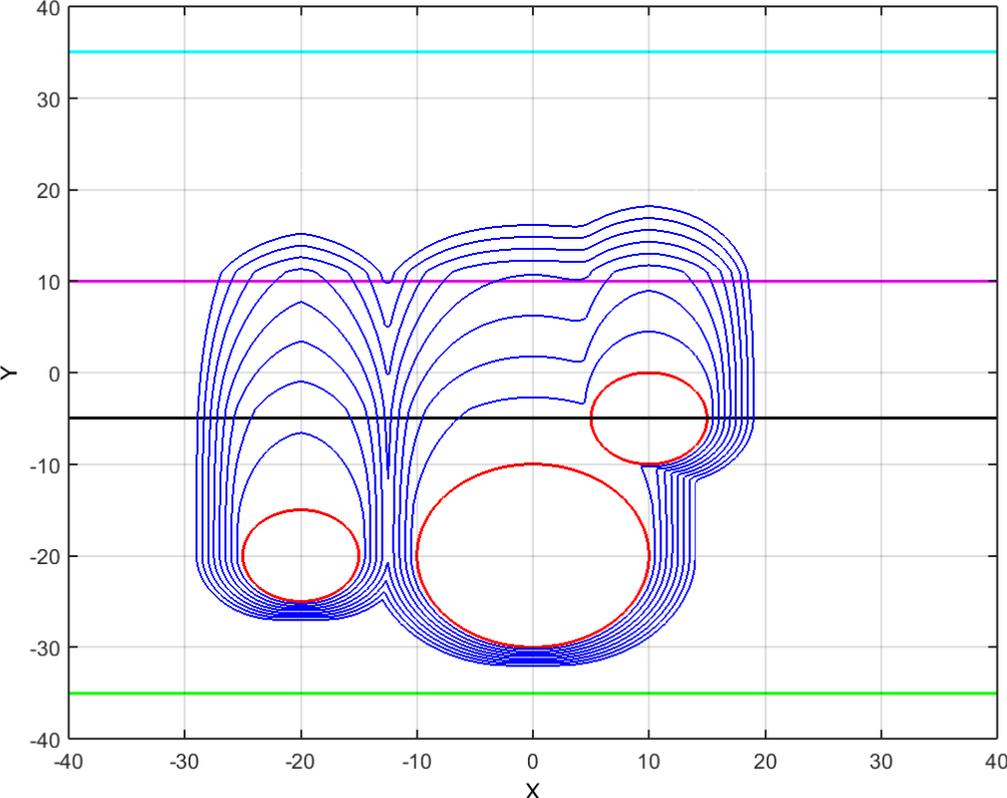


Fig. 4.16. Wildfire evolution under varying fuel density

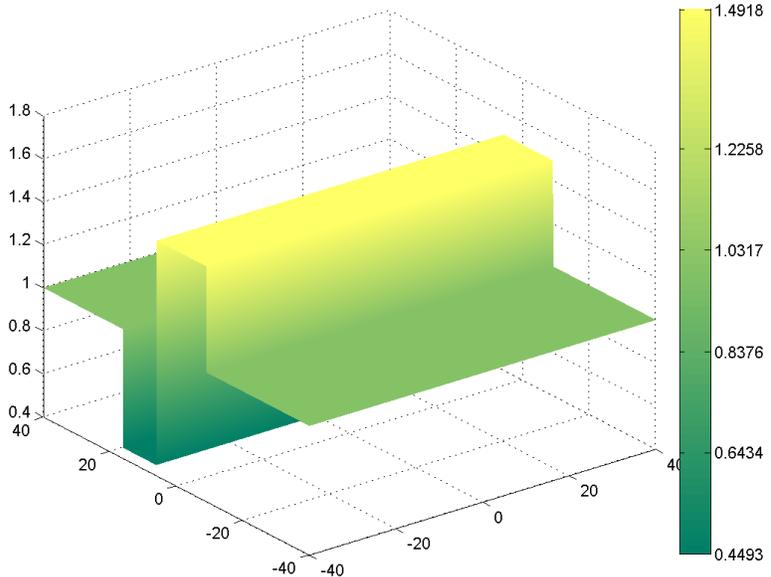


Fig. 4.17. Terrain slope map (e^μ), $\mu_{down} = -0.7, \mu_{up} = 0.2$

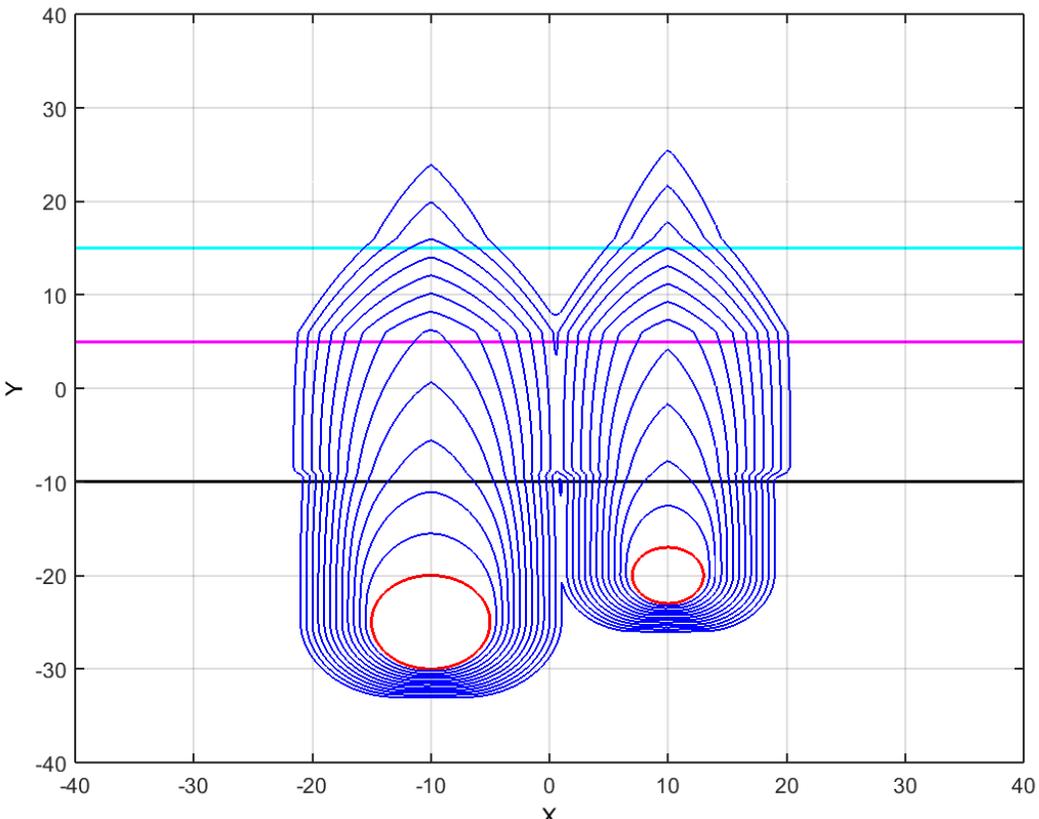


Fig. 4.18. Wildfire evolution under varying terrain topography

4.4. Conclusions

This chapter explored modelling approaches to wind aided wildfire evolution, with emphasis on geometrical quasi-empirical models, which combine both experimental data and fundamental governing laws to determine the spatio-temporal dynamics of the fire front. A numerical analysis of the models was undertaken in a two dimensional space, using a level set formulation, to explore the behavioral dynamics of a wildfire in a series of idealized scenarios, under constant or discrete time-varying parameters (wind speed magnitude, wind direction, biofuel loading, terrain topography). Additional improvements to the model can be made by incorporating a series of real-world values: satellite imaging data and digital terrain elevation data can be used to refine the model and obtain more accurate results; these real-world parameters can greatly aid in determining an effective fire suppression strategy, an approach that is explored in more detail in the subsequent chapters.

5. A new approach for controlling crystal growth using a level set formulation

5.1. Introduction

Dendritic crystal growth is mainly driven by the temperature gradient evolution over time. Thus, the speed at which the boundary of the crystal evolves is mainly affected by how the temperature around it changes (Glicksman and Lupulescu, 2004).

The problem of controlling crystal growth can be viewed as the problem of determining the time-dependent temperature distribution over the domain of interest such that the boundary of the crystal evolves in a desired fashion.

A solution to this problem could be found by solving an inverse design problem over the spatial domain of interest. Given a model of crystal interface evolution and the desired crystal front position over time, the goal is to compute the temperature distribution inside the domain and the heat flux at its boundaries that would ensure that the crystal boundary follows the desired ‘trajectory’ (Zabaras et. al. 1992, Kang and Zabaras, 1995). These problems are ill-posed and require some form of spatial regularization method to ensure stability. Since these problems are open loop approaches, they cannot deal with inherent perturbations or measurement noise that would be present during control.

Another strategy involves controlling the changes in heat flux on the boundary of the spatial domain so that the heat distribution inside the domain follows the desired reference distribution or in such way that a specific functional is minimized. The control of the free boundary of an evolving crystal represented by a sharp interface model (the Stefan problem) has been approached for the first time by (Hinze and Ziegenbalg, 2007) using an optimal control approach, according to the authors. They represented the free interface using a graph, and provided examples for one and two dimensional cases. Other approaches have been developed using simplified one dimensional problems (Banks et. al., 2008), where the laser control of the boundary is considered. In (Bernauer and Herzog, 2011) the time evolution of the crystal boundary was defined as a motion planning problem (optimal control), with the interface now being represented using a level set formulation.

Although over the past few years the areas of modeling and numerical simulations of dendritic crystal growth have become an important area of research, there are no articles that point towards control of geometric dendritic growth models presented in the previous sections. In this chapter a new method for controlling the shape of an evolving interface, which is based on a geometric model of interface

evolution and exploits the level set formulation is introduced and demonstrated through numerical simulations. The chapter is organized as follows: Section 5.2 briefly describes the geometrical representations of dendritic growth together with a basic control problem, where the temperature is considered to be the manipulated variable. In Section 5.3, a novel approach for controlling crystal growth, using a level set formulation, is presented. The next part, Section 5.4 provides details of the numerical implementation of the proposed control algorithm in two dimensions, followed by Section 5.5, where the extension to a three dimensional space is presented. Finally, Section 5.6 presents detailed numerical results for the proposed control law, using both artificial interface reference shapes and complex interface profiles extracted from experimental data.

5.2. Level set formulation of dendritic crystal growth

Let us consider the time evolution of a solid-liquid interface, in a classic case of dendritic solidification. Let $\Gamma(t)$ be a positively oriented curve (closed and simple planar curve) representing the moving boundary. The curve is defined as a continuous map $\tilde{x}^F: [0, S) \times [0, T] \rightarrow \mathbb{R}^2$, where $\tilde{x}(s, t) = (x_1^F(s, t), x_2^F(s, t))$ are the Cartesian coordinates of a point on the curve $\Gamma(t)$ at a given time, $s \in [0, S)$ is the arc length, and the periodic boundary condition $\tilde{x}(0, t) = \tilde{x}(S, t)$ holds. The curve is parameterised so that the interior is on the left in the direction of increasing s . The unit tangent to the curve is $\hat{t} = \tilde{x}_s / |\tilde{x}_s|$ and the unit normal, pointing inside the curve, is given by $\hat{n} = \tilde{x}_s^\perp / |\tilde{x}_s^\perp|$. Here $|\tilde{x}| = (\tilde{x} \cdot \tilde{x})^{1/2}$ denotes the norm of \tilde{x} , $\tilde{x} \cdot \tilde{y}$ denotes the inner product of \tilde{x} and \tilde{y} , $(\alpha_1, \alpha_2)^\perp = (-\alpha_1, \alpha_2)$ and $\tilde{x}_s = \partial \tilde{x} / \partial s$. The velocity of the interface $\Gamma(t)$ can be written as

$$\frac{d\tilde{x}}{dt} = V_n \hat{n} + V_\tau \hat{t} \quad (5.1)$$

where V_n and V_τ are the projections of the velocity in the normal and tangential direction, respectively. The propagation speed of interface is assumed to be orthogonal to the tangent \hat{t} (i.e. $V_\tau = 0$) and typically depends on the geometrical properties of the interface, such as the curvature $k(s, t) = -\hat{n} \cdot \frac{\partial^2 \tilde{x}}{\partial t^2}$ and its derivatives, and on the orientation angle θ (defined by $\cos \theta = \hat{n} \cdot \hat{y}$)

$$\hat{n} \cdot \frac{d\tilde{x}}{dt} = V_n(k, \theta) \quad (5.2)$$

Now considering the class of local geometric models introduced in previous chapters that have been used to model dendritic crystal growth:

$$\left\{ \begin{array}{l} \hat{n} \frac{d\tilde{x}}{dt} = \left(F(k) + \gamma \frac{\partial^2 k}{\partial s^2} \right) [1 + \varepsilon \cos(m\theta)] \\ \tilde{x}(s, 0) = \Gamma(0) = \Gamma_0 \\ F(k) = \sum_{i=0}^p a_i k^i \end{array} \right. \quad (5.3)$$

where \hat{n} is the unit normal of $\Gamma(t)$ at \tilde{x} at time t , $a_i \in \mathbb{R}$ are physical parameters that are related to the physical growth process, and the term $\gamma \frac{\partial^2 k}{\partial s^2}$, with $\gamma \in \mathbb{R}^+$ is a measure of surface tension. In the case of dendritic crystal growth, it was shown that $p=3$, $a_1 = 1$, a_2 is related to the degree of undercooling and a_3 represents a measure of the minimum bubble size for nucleation. The extra term proportional to ε was introduced to include the effects of repeated side-branching characteristic to dendritic crystal growth. This effect is reflected in enhanced growth in the direction given by the angle θ made by the curve normal and the axis along preferred m -fold symmetry.

The level set method provides a framework to capture the interface dynamics by using a higher dimensional function φ to parameterize a bounded domain in which the interface is evolving, and furthermore the interface corresponds to the zero level set of the function φ . Let $\Omega_\Gamma(t) \subset \mathbb{R}^2$ denote the interior of $\Gamma(t)$. The level set function $\varphi_\Gamma: \mathbb{R}^2 \times [0, T] \rightarrow \mathbb{R}$ is then

$$\varphi_\Gamma(x, t) = \begin{cases} -d(x, t), & x \in \Omega_\Gamma \\ 0, & x \in \Gamma \\ +d(x, t), & x \in \mathbb{R}^2 \setminus \Omega_\Gamma \end{cases} \quad (5.4)$$

where $d(x, t)$ is a distance function defined by $d(x, t) = \min_{s \in [0, S]} \|x - x_\Gamma(s, t)\|$. If $\Gamma(t)$ represents time varying interface in a domain $\Omega \subset \mathbb{R}^2$, and $\varphi_\Gamma(x, t)$ is its attached level set function, then the zero level set of the function $\varphi_\Gamma(x, t)$ has to evolve in the same way as the interface, and $\Gamma(t) = \{x \in \mathbb{R}^2 | \varphi_\Gamma(x, t) = 0\}$ holds for any time $0 \leq t \leq T_{max}$. Using (5.5) the model (5.4) can be rewritten as:

$$\frac{\partial \varphi_\Gamma}{\partial t} + \left(a_0 + \sum_{i=1}^p a_i k^i + \gamma \frac{\partial^2 k}{\partial s^2} \right) |\nabla \varphi_\Gamma| = 0 \quad (5.5)$$

where the speed function considered as a weighted curvature polynomial:

$$F = \sum_{i=0}^p a_i k^i + \gamma \frac{\partial^2 k}{\partial s^2} \quad (5.6)$$

with $\gamma > 0$ and $a_{i>0} \in \mathbb{R}^+$. The advection term a_0 determines the direction of evolution for $\Gamma(t)$. Negative values will move the points $x \in \Gamma(t)$ in the direction opposite to the curve normal (solidification), and positive values will move the points $x \in \Gamma(t)$ towards the inside of the curve, in the normal direction (melting).

The main factor driving dendritic crystal growth is the undercooling imposed to the molten phase. It is therefore natural to choose the temperature as a control parameter and develop a control strategy using it as the manipulated variable and the output as the solid-liquid interface. Let us consider the temperature on a bounded domain $\Omega \subset \mathbb{R}^2$, defined as $u: \Omega \rightarrow \mathbb{R}$. The governing equation inside the domain is given by the time dependent heat equation:

$$\begin{cases} \frac{\partial}{\partial t} u(x, y, t) = D \cdot \nabla(\nabla u(x, y, t)) + u_{ext} \\ u(x, y, 0) = u_0 \end{cases} \quad (5.7)$$

where D is the thermal diffusivity, u_0 is the initial temperature inside Ω at time $t = 0$ and $u_{ext}: \Omega \rightarrow \mathbb{R}$ represents an external heat source. The idea is to manipulate the input variable u_{ext} such that $u(x, y, t)$ takes the desired profile inside Ω , such that the zero level set of φ_Γ reaches a desired profile, given the evolution law (5.5). This would mean that the undercooling changes in time, change which would prescribe the shape and speed of growth of the crystal. The input control function can be applied on the boundary of the domain $\partial\Omega$ (boundary control) or it can be prescribed on the whole surface of Ω (distributed control). In the next section, a distributed control problem is presented, using a level set formulation of the evolving crystal interface.

5.3. Control of evolving interfaces governed by geometric laws of evolution

Consider the general geometrical model of boundary evolution:

$$\hat{n} \frac{d\tilde{x}}{dt} = F(k(s, t), u(x, t)) = u(x, t) \cdot \left[a_0 + \sum_{i=1}^3 a_i k^i(s, t) + \gamma \frac{\partial^2 k(s, t)}{\partial s^2} \right] \quad (5.8)$$

where $u(x, t)$ represents the control function; which in our case, is the temperature. The domain in which the crystal grows is assumed to be a bounded, rectangular domain $\Omega \subset \mathbb{R}^2$, $\Omega: [-l, l] \times [-w, w]$. To describe the control method the following need to be defined:

- Let the desired crystal interface (reference) be denoted by a closed and non-intersecting contour, Γ_r
- The evolving crystal boundary, with the initial crystal seed $\Gamma_0 = \Gamma(t = 0)$, which needs to be evolved in a controlled manner is indicated by $\Gamma(t)$.
- Let $\varphi_r(x)$, $\varphi_r: \Omega \times [0, T] \rightarrow \mathbb{R}$ and $\varphi_\Gamma(x, t)$, $\varphi_\Gamma: \Omega \times [0, T] \rightarrow \mathbb{R}$ be the level set functions attached to with Γ_r and $\Gamma(t)$ respectively, with $T \geq 0$ representing the maximum time window.

The goal is to design a feedback control law:

$$u(x, t) = u(\varphi_r(x), \varphi_\Gamma(x, t), t) \quad (5.9)$$

such that $\varphi_\Gamma(x, t) \rightarrow \varphi_r(x)$ (i.e. $\Gamma(t) \rightarrow \Gamma_r$) as $t \rightarrow \infty$. Equation (5.9) assumes that the growth process can be viewed in real-time and the control $u(x, t)$ can be applied on Ω in real-time (i.e. unnoticeable time delay, regardless of its amplitude or spatial location).

To properly observe the growth process, it is assumed that optical data is used, and continuous snapshots are taken of the domain Ω in which the crystal is growing, at appropriate time steps. Let $I(x, t): \Omega \rightarrow \mathbb{R}$ be the measured image intensity function of a snapshot taken at time t , which is processed to determine the crystal interface $\Gamma(t)$ and its interior $\Omega_\Gamma(t)$. By introducing the functions:

$$\Phi_r(x) = \text{sign}(-\varphi_r(x, t)) \quad (5.10)$$

$$\Phi_\Gamma(x, t) = \text{sign}(-\varphi_\Gamma(x, t)) \quad (5.11)$$

a distributed control law can be formulated as:

$$u(x, t) = K_p \cdot e(x, t) + K_i \int_0^t e(x, \tau) d\tau + K_d \frac{d}{dt} e(x, t) \quad (5.12)$$

where K_p, K_i, K_d are the proportional, integral and derivative controller gains respectively, and the error function is given by:

$$e(x, t) = -\frac{1}{2} \Phi_r(x) \cdot [\Phi_r(x) - \Phi_\Gamma(x, t)] \cdot [\varphi_r(x) - \varphi_\Gamma(x, t)] \quad (5.13)$$

The term (5.13) represents the error function that is used in as the input to the closed loop control law. The error function is defined using level set representations of two simple non-intersecting 2D curves, namely the ‘set-point’ curve Γ_r and the controlled interface $\Gamma(t)$. The first part of the equation, encompassing the functions $\Phi_r(x), \Phi_\Gamma(x, t)$ contains the information related to the zero level contours of the set-point and evolving interface, explicitly providing the spatial locations where the control needs to be applied. The right hand element of (5.13), containing the term defined by the level set function $g(\varphi_r(x), \varphi_\Gamma(x, t)) = \varphi_r(x) - \varphi_\Gamma(x, t)$ provides a measure of the distances between the two curves of interest, providing information regarding the magnitude of the control action that needs to be applied to minimize $e(x, t)$. Therefore, equation (5.13) provides a measure of the interface shape difference between two arbitrary curves, regardless of boundary complexity or position within a specified domain, and can be directly used in the design of a feedback control law.

Then, the level set formulation for the time evolution of the closed loop system (5.8) is governed by:

$$\frac{\partial \varphi_\Gamma}{\partial t} + u(x, y, t) \cdot \left(a_0 + \gamma \frac{\partial^2 k(s, t)}{\partial s^2} \right) |\nabla \varphi_\Gamma| = \left(- \sum_{i=1}^3 a_i k(s, t)^i \right) \cdot |u(x, y, t)| \cdot |\nabla \varphi_\Gamma| \quad (5.14)$$

There are two distinct behaviors related to crystal growth, namely solidification and melting. A desired crystal shape can be achieved not only by controlling solidification, but also through controlled

melting of a crystal. By joining these two cases, any desired outline can be achieved, starting from an arbitrary seed shape.

5.4. Discretization

It is worth noting that (5.14) contains the curvature power terms $k(s, t)^i$, $i = \overline{1, 3}$ and the curvature Laplacian $k_{ss} = \frac{\partial^2 k(s, t)}{\partial s^2}$ which are expressed in terms of interface arc length. In order to simulate the closed loop system numerically using the level set approach, these curvature terms have to be approximated from the level set function φ_Γ .

As shown in Chapter 3, Section 3.4.2, $k(s, t)$ and k_{ss} can be expressed in terms of the level set function φ_Γ and its spatial derivatives as:

$$k = \nabla \frac{\nabla \varphi_\Gamma}{|\nabla \varphi_\Gamma|} = \frac{\varphi_x^2 \varphi_{yy} - 2\varphi_x \varphi_y \varphi_{xy} + \varphi_y^2 \varphi_{xx}}{|\nabla \varphi_\Gamma|^3} \quad (5.15)$$

$$k_{ss} = \nabla \left[\nabla \left[\nabla \cdot \frac{\nabla \varphi_\Gamma}{|\nabla \varphi_\Gamma|} \right] \cdot \frac{(\varphi_y, -\varphi_x)}{|\nabla \varphi_\Gamma|} \right] \cdot \frac{(\varphi_y, -\varphi_x)}{|\nabla \varphi_\Gamma|} = \frac{\varphi_y^2 k_{xx} - 2k_{xy} \varphi_x \varphi_y + \varphi_x^2 k_{yy}}{\varphi_x^2 + \varphi_y^2} - \frac{k(k_x \varphi_x + k_y \varphi_y)}{\sqrt{\varphi_x^2 + \varphi_y^2}} \quad (5.16)$$

where $(\cdot)_y$, $(\cdot)_x$, $(\cdot)_{xx}$, $(\cdot)_{yy}$, $(\cdot)_{xy}$ are the first order, second order and mixed directional derivatives of φ_Γ and k . Computing k_{ss} using (5.16) involves computing derivatives of the level set function up to the fourth order, which introduces numerical noise. To decrease the errors in numerical estimations, fourth order accurate approximations are used when calculating k_{ss} (Chopp and Sethian, 1999).

To advance the front in time a third order total variation-diminishing (TVD) Runge-Kutta (which has $\mathcal{O}(\Delta t^3)$ truncation error) scheme was used to solve (5.8). TVD RK schemes are guaranteed not to introduce any spurious fluctuations during the integration, which might result from higher order temporal discretization (Shu and Osher, 1989). Consider the following equation:

$$\frac{\partial \varphi}{\partial t} = \mathcal{L}(\varphi) \quad (5.17)$$

where \mathcal{L} denotes the spatial operator of (5.8). Then, the time discretization of (5.17) can be expressed by:

$$\begin{cases} \varphi^{(1)} = \varphi^{(0)} + \Delta t \hat{\mathcal{L}}(\varphi^{(0)}) \\ \varphi^{(2)} = \frac{3}{4} \varphi^{(0)} + \frac{1}{4} \varphi^{(1)} + \frac{1}{4} \Delta t \hat{\mathcal{L}}(\varphi^{(1)}) \\ \varphi^{(3)} = \frac{1}{4} \varphi^{(0)} + \frac{2}{3} \varphi^{(2)} + \frac{2}{3} \Delta t \hat{\mathcal{L}}(\varphi^{(2)}) \end{cases} \quad (5.18)$$

with $\varphi^{(0)} = \varphi_r(t)$, $\varphi_r(t + \Delta t) = \varphi^{(3)}$ and $\hat{\mathcal{L}}$ represents the discrete approximation of \mathcal{L} .

5.5. Extension to three dimensions

The previous sections considered controlling the crystal growth in two dimensions. This simplifying statement is adequate in many practical cases, where only planar interactions are of interest. However, since dendritic growth is a three dimensional process, in this section an extension of the control method to three dimensions is presented, using a level set approach. Consider the evolution of a crystal seed in a bounded domain $D \subset \mathbb{R}^3$. The evolving interface is represented by the zero level set of a three dimensional SDF φ , i.e. a surface that separates two domains of positive volumes. Due to the properties of the level set approach, an extension to higher dimensions is straightforward and is reflected explicitly in its geometric properties:

- The interface unit normal $\vec{n} = (n_x, n_y, n_z)$ is given by

$$\vec{n} = \frac{\left(\frac{\partial \varphi}{\partial x}, \frac{\partial \varphi}{\partial y}, \frac{\partial \varphi}{\partial z}\right)}{\left|\left(\frac{\partial \varphi}{\partial x}, \frac{\partial \varphi}{\partial y}, \frac{\partial \varphi}{\partial z}\right)\right|} = \frac{\nabla \varphi}{|\nabla \varphi|} \quad (5.19)$$

- The divergence of the unit normal (interface mean curvature k) is defined by:

$$k_1 = \varphi_x^2 \varphi_{yy} + \varphi_y^2 \varphi_{xx} + \varphi_x^2 \varphi_{zz} + \varphi_z^2 \varphi_{xx} + \varphi_y^2 \varphi_{zz} + \varphi_z^2 \varphi_{yy} \quad (5.20a)$$

$$k_2 = \varphi_x \varphi_y \varphi_{xy} + \varphi_x \varphi_z \varphi_{xz} + \varphi_y \varphi_z \varphi_{yz} \quad (5.20b)$$

$$k = \frac{1}{|\nabla \varphi|^3} (k_1 - 2k_2) \quad (5.20c)$$

It is obvious that increasing the spatial dimension, the amount of information that needs to be processed is significantly increased. Since the level set function φ needs to be a signed distance function, additional effort is needed in properly defining it. Similar to (5.19) and (5.20), other geometrical quantities can be defined in a similar manner from their corresponding definition.

Considering that φ is defined on a discrete uniform Cartesian grid $\{x_\alpha, y_\beta, z_\gamma\}$ with $\{x_\alpha, y_\beta, z_\gamma | 1 \leq \alpha \leq m, 1 \leq \beta \leq n, 1 \leq \gamma \leq p\}$, $m, n, p \in \mathbb{N}$, the discretization of the LSF can be done in a dimension by dimension manner for all spatial directions, using the formulae from previous chapters, for all spatial derivatives. It is worth mentioning that the re-initialization algorithm can be extended in an upfront manner, using all relevant elements that are added due to the dimensionality increase.

Having set the framework for interface dynamics representation in three dimensions, the control method (5.12) is readily applicable, with the difference that the reference and evolving interfaces are

now represented by surfaces. Also, the interface dynamics is observed in a 3D space, and therefore observations in this sense are considered to be available to close the feedback loop. To exemplify the motion of a surface in \mathbb{R}^3 , consider a bounded and uniformly spaced spatial domain of $D = [-2,2] \times [-2,2] \times [-2,2]$, with $N = 50$ grid nodes in each spatial direction, yielding a spatial step size $\Delta x = \Delta y = \Delta z = 0.081$. The initial surface is set to be a regular octahedron, which is set to evolve under a linear growth model, namely evolution dependent on local curvature and normal velocity (Fig 5.1). The normal velocity is set to be $V_n = -0.1$ and the curvature weight $b = 0.2$, i.e. the crystal is melting. Time plots of the evolution are taken at every $\Delta t_{plot} = 0.04$ s until the final simulation time $T = 0.02$ s (Fig 5.2).

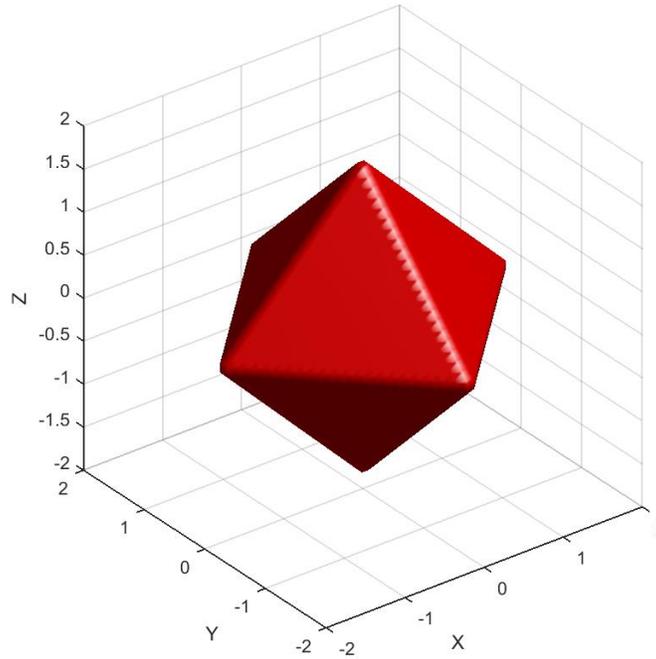


Fig. 5.1. Initial interface (regular octahedron)

5.6. Numerical simulations of the crystal growth control method

This section demonstrates the proposed control approaches through numerical simulation examples of applied control of the crystal interface. In this dissertation both synthetic and real reference shapes, derived from experimental data, are used. Three control strategies are investigated, namely the classical P, PI and PID distributed control laws. The synthetic data is generated in both 2D and 3D cases and used directly in the control method. To generate a reference shape from real world data,

experimental data obtained from observing growth patterns of NH_4Br crystals were analysed. Snapshots were recorded during the growth process, and stored in a PC. The resulting image intensity functions were then processed to extract the complex solid liquid interface of the crystal, which is then used as the reference shape in the control algorithm. Following the numerical simulations, a comparison of the several proposed control laws is performed using multiple performance indicators.

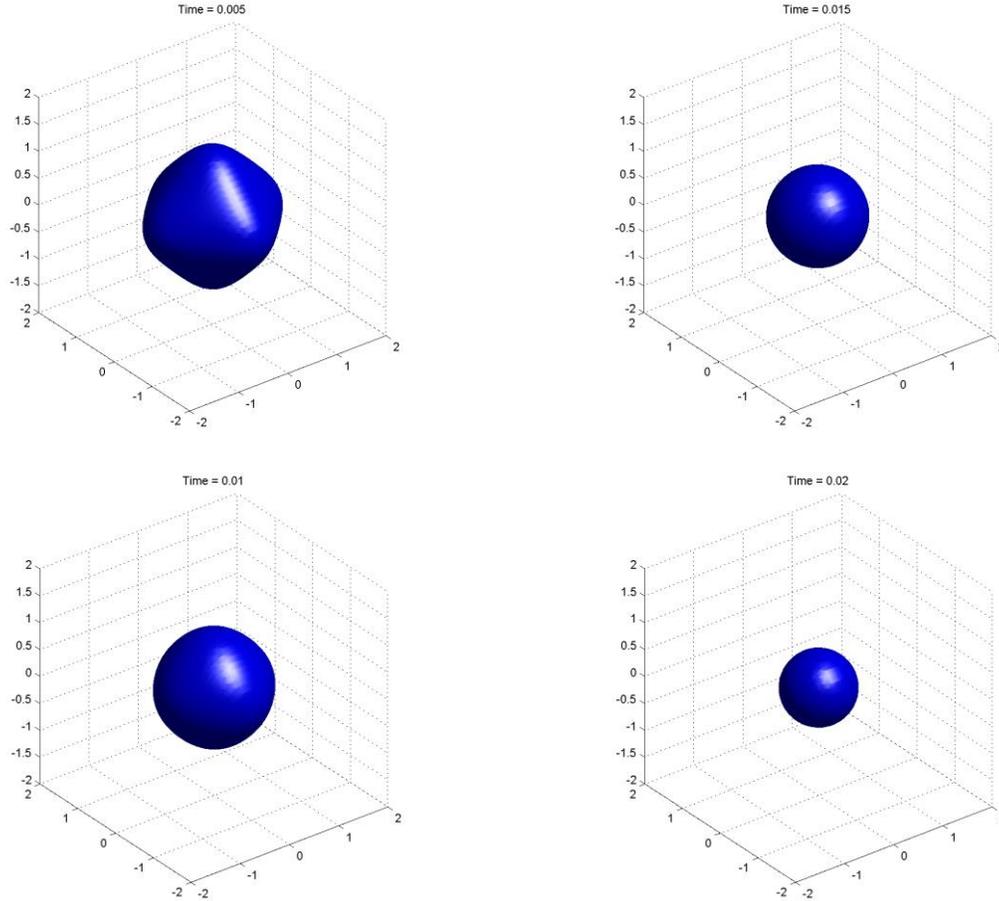


Fig. 5.2. Evolution of a regular octahedron under normal velocity given by curvature and advection in the normal direction.

5.6.1. Synthetic data simulations

To exemplify the proposed control method, several cases are presented, in both two and three dimensional spaces. Using a proportional control law and artificially generated reference sets, solidification-melting scenarios are simulated, where the control law is given by (5.10)-(5.13).

In two dimensions, an evolving crystal interface is described by a simple, non-intersecting closed curve. An artificially generated Jordan curve (Krantz, 2008) in the form of a 4-point star was chosen as

the reference shape (Γ_r), together with an initial circular seed (Γ_0) of radius $R = 1.4$ (Fig 5.3). The goal is to control the evolution of an initial crystal seed, of arbitrary area and arc length, in order to obtain a crystal that has the given reference shape. The parameters a_i , $i = \overline{0,4}$ of the model (5.8) were chosen as follows $a_0 = 1$, $a_1 = -0.01$, $a_2 = -10^{-5}$, $a_3 = -10^{-5}$, $a_4 = -10^{-5}$. The initial crystal shape is chosen such that the crystal interface has some sections inside and others outside of Γ_r . The two dimensional space in which the growth is observed is set as the square $\Omega: [-3, 3] \times [-3, 3]$, which is discretized on a uniform Cartesian grid with resolution of 300×300 points, giving a spatial step size of $\Delta x = \Delta y = 0.02$. The controller gains in (5.12) were chosen empirically as $K_p = 5$, $K_i = 1$ and $K_D = 10^{-3}$. To evaluate the performance of the proposed controller for the two dimensional case, the following normalized error functions E_{arc} and E_{area} were introduced, together with an overall performance error G_{error} :

$$E_{arc}(t) = \frac{Arclength(\Gamma_r) - Arclength(\Gamma(t))}{Arclength(\Gamma_r)} \quad (5.22)$$

$$E_{area}(t) = \frac{Area(\Omega_r) - Area(\Omega_\Gamma)}{Area(\Omega_r)} \quad (5.23)$$

$$G_{error}(t) = \text{sum}(|e(x, t)|) \quad (5.24)$$

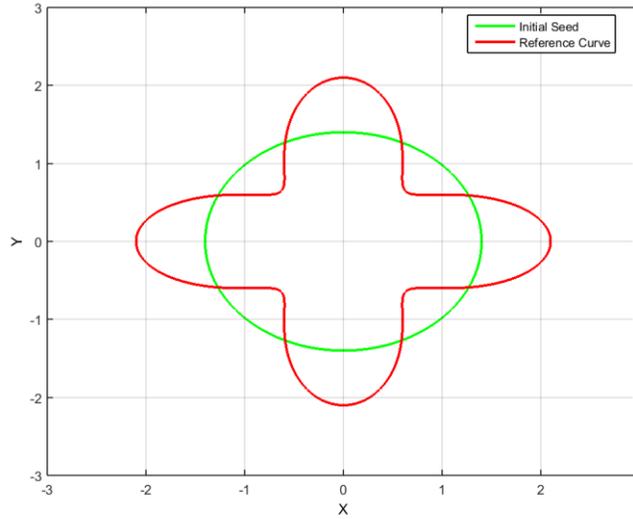


Fig. 5.3. Setup for the 2D control problem

In the three dimensional case, the error measures for the controller performance are chosen to be the surface area and volume errors, E_{sa} and E_v , defined by:

$$E_{sa} = \frac{SurfaceArea(\Gamma_r) - SurfaceArea(\Gamma(t))}{SurfaceArea(\Gamma_r)} \quad (5.25)$$

$$E_v = \frac{\text{Volume}(\Gamma_r) - \text{Volume}(\Gamma(t))}{\text{Volume}(\Gamma_r)} \quad (5.26)$$

where now Γ_r and $\Gamma(t)$ are three dimensional functions.

The controlled spatio-temporal system (5.14) was then simulated for $T_{sim} = 1.5$ seconds using the proportional controller (Fig. 5.5). The time evolution of $E_{arc}(t)$, $E_{area}(t)$ and $G_{error}(t)$ illustrated in Fig. 5.5 – Fig.5.7, exemplify the performance of the control approach.

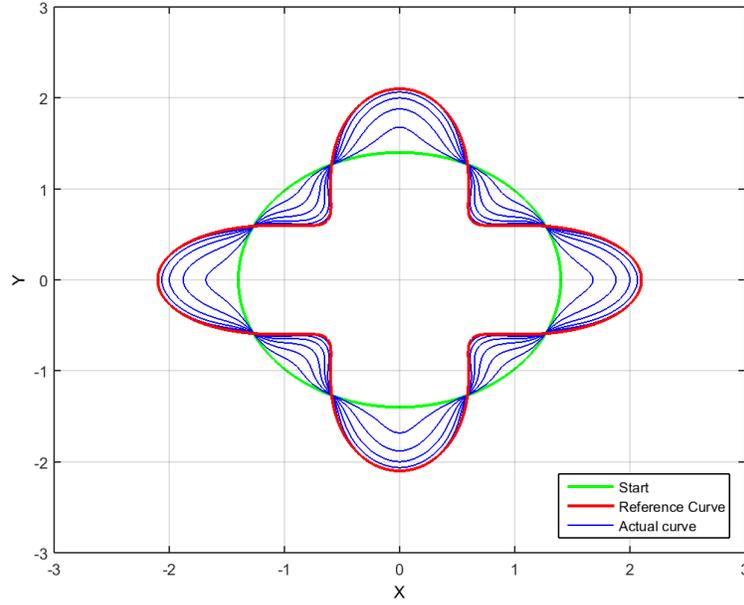


Fig. 5.4. Time evolution of the initial crystal seed. Time plots (blue) shown in increments of $\Delta t_{plot} = 0.1$ seconds

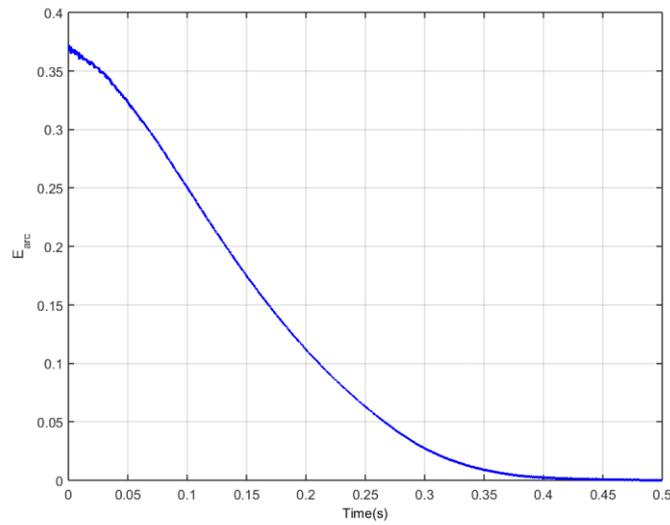


Fig. 5.5. Arc length error 2D synthetic seed

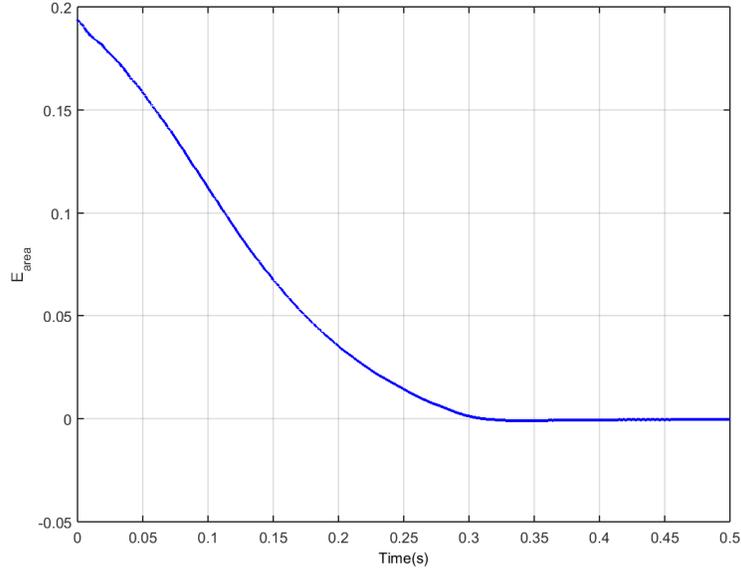


Fig. 5.6. Area error 2D synthetic seed

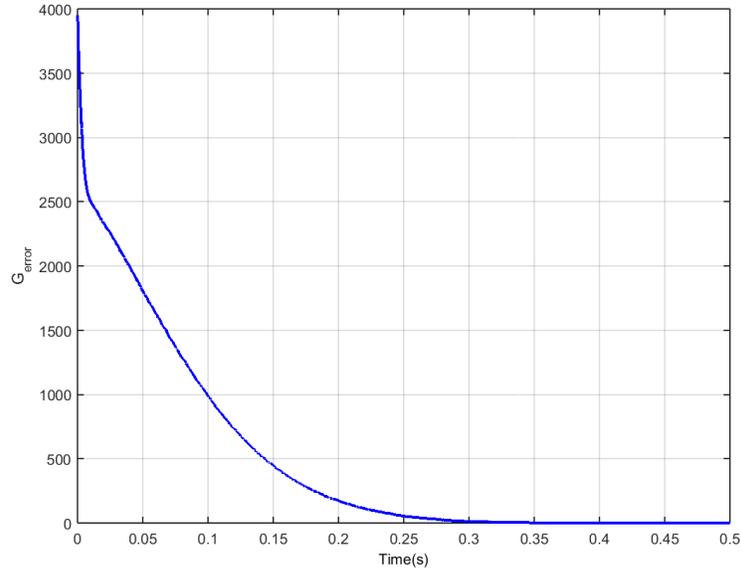


Fig. 5.7. Global error 2D synthetic seed

In three dimensions, the growth is observed in the cube given by $\Omega: [-2, 2] \times [-2, 2] \times [-2, 2]$, which is discretized on a Cartesian grid with $N = 100$ Points in each spatial direction. The initial volume is set to be a cube centered at $O(0,0,0)$ with edge length $L = 2$, and the reference volume is a sphere with center at $C(0.2,0.2,0.2)$ and radius $R = 1.5$ (Fig. 5.8).

Under these assumptions the interface is set to evolve under the geometric law given by

$$\frac{\partial \varphi_\Gamma}{\partial t} + u(x, y, z, t) \cdot (a_0) |\nabla \varphi_\Gamma| = (-a_1 k) \cdot |u(x, y, z, t)| \cdot |\nabla \varphi_\Gamma| \quad (5.27)$$

with $a_0 = 1$, $a_1 = -0.01$ and k given by (5.20c). The PID control law (5.12) is used, with an empirical choice for the parameters $K_P = 8$, $K_i = 5$ and $K_D = 1 \cdot 10^{-3}$. The closed loop system is then simulated for $T_{sim} = 0.4$ seconds (Fig. 5.9). In order to quantify the performance of the proposed 3D control approach, the enclosed volume and the surface area errors are calculated and normalized using the respective reference values (Fig. 5.10 and Fig. 5.11).

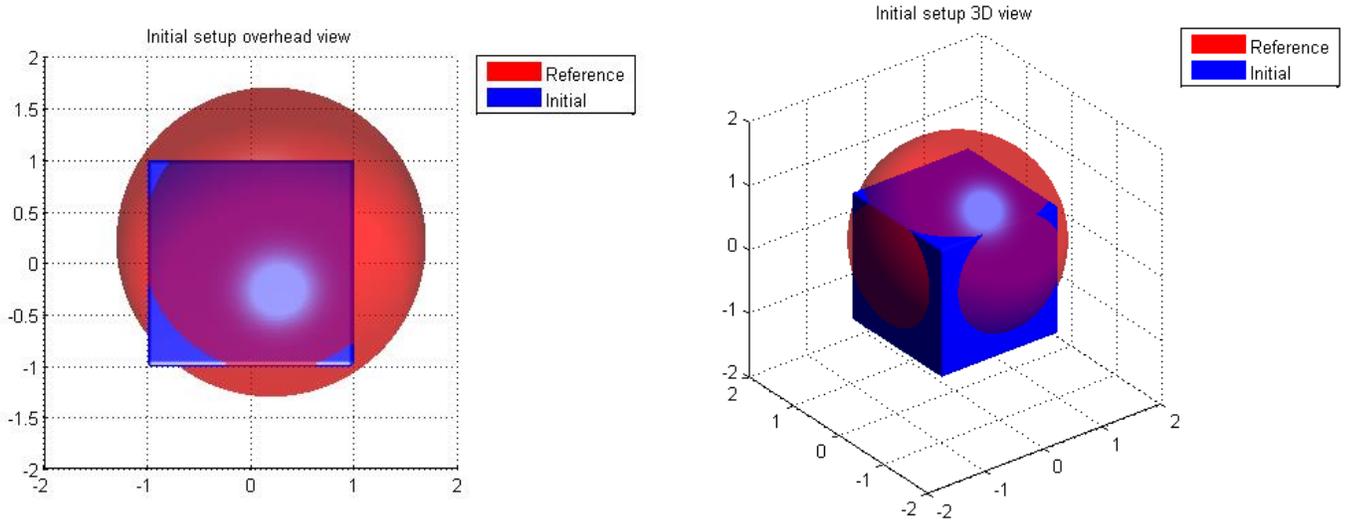


Fig. 5.8. Initial conditions for the 3D control problem

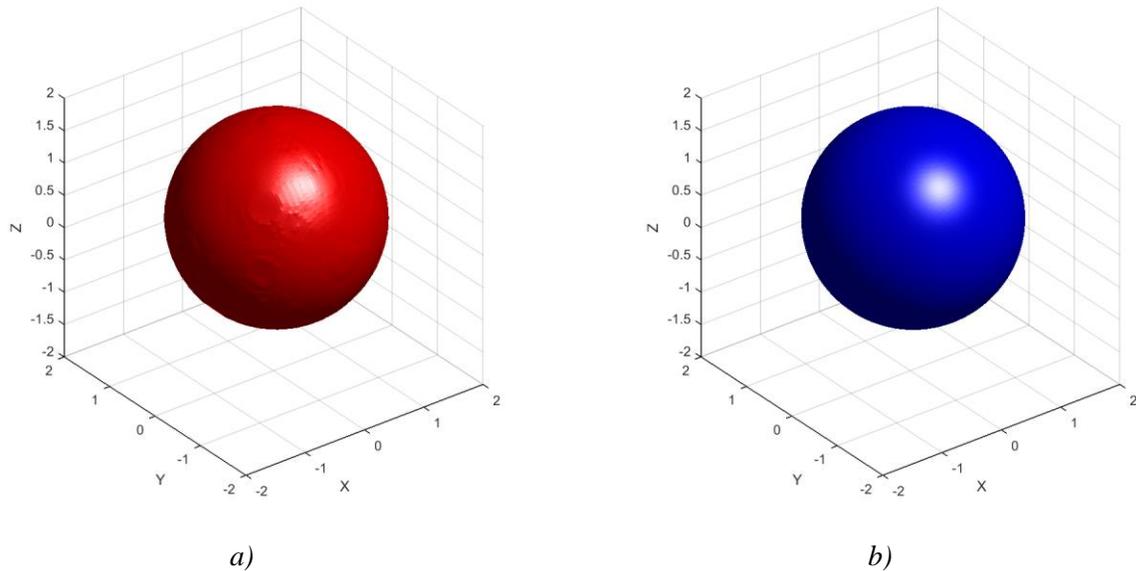


Fig. 5.9. 3D controlled growth: a) system evolution after time T_{sim} b) Reference shape

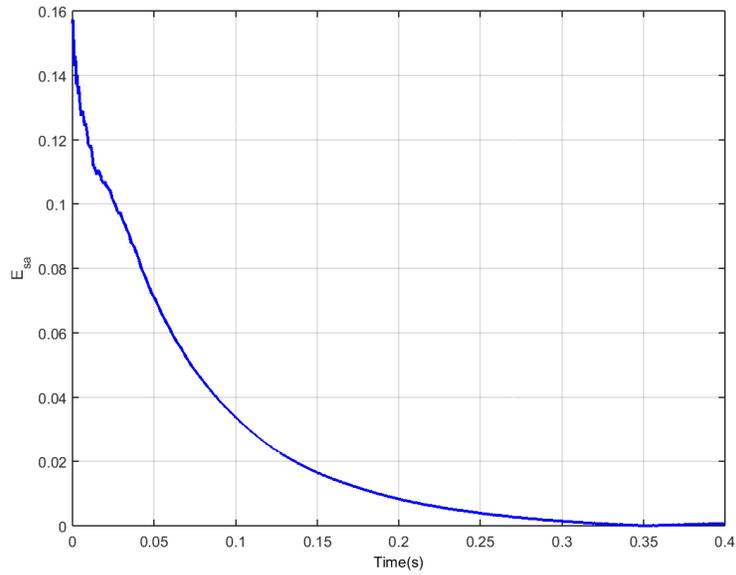


Fig. 5.10. Surface area error – 3D control

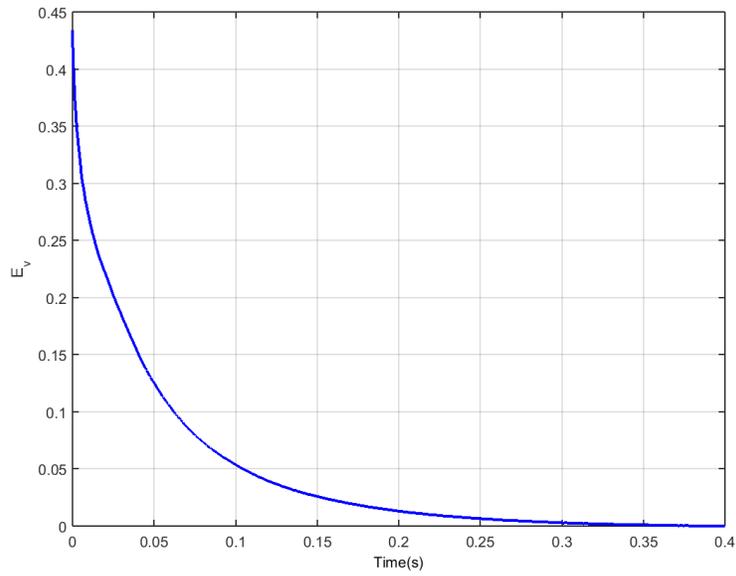


Fig. 5.11. Enclosed volume error – 3D control

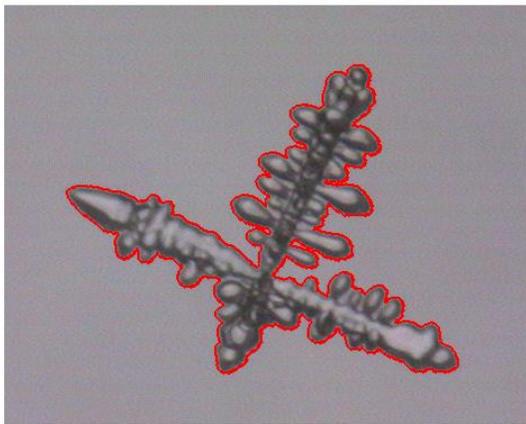
In both 2D and 3D cases, the error converges towards zero, and the initial seed reaches the desired interface shape, thus proving the effectiveness of the novel control approach.

5.6.2. Simulations using experimental data

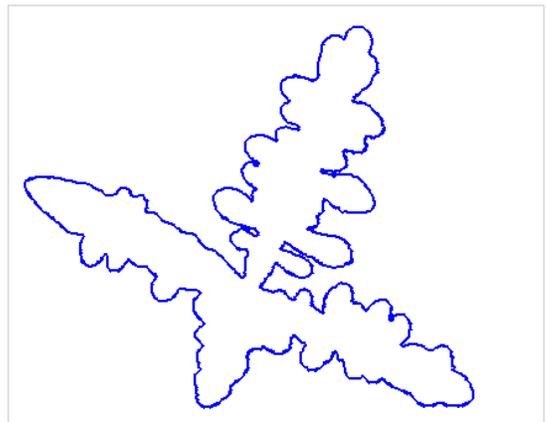
This section will explore the control of dendritic growth using observed growth patterns of NH_4Br crystals. In the previous section an application of the proposed control method was presented, using relatively simple interface shapes. However it is well known that dendritic patterns exhibit highly complex spatio-temporal patterns and the aim is to be able to control any initial seed into an arbitrarily complex crystal shape. For this purpose, experimental data of the NH_4Br growth process was captured at relevant time intervals in the form of RGB images, and a single frame was selected for analysis (Fig. 5.12). After the image in Fig. 5.12 is processed and an image segmentation algorithm is applied (Gonzalez et al., 2004), the underlying intricate boundary of the dendrite is extracted (Fig. 5.13a, Fig 5.13b), which is then used as the reference for the controlled system.



Fig. 5.12. Growth pattern of NH_4Br dendrite crystal



a)



b)

Fig. 5.13. a) Segmented image (central dendrite) b) Boundary of NH_4Br dendrite crystal

The discretization is changed with regards to the previous section, with the spatial domain now defined by $\Omega: [-3, 3] \times [-3, 3]$. The domain is discretized on a uniform Cartesian grid using a spatial step size of $\Delta x = \Delta y = 0.005$, resulting in a mesh of 1200×1200 nodes. The model parameters in (5.8) were chosen as $a_0 = 1$, $a_1 = -0.01$, $a_2 = -10^{-5}$, $a_3 = -10^{-5}$, $a_4 = -10^{-5}$. The PID parameters were chosen empirically to be $K_p = 5$, $K_i = 8$, $K_D = 0$. Having defined the reference curve, the initial seed (Γ_0) is defined by a circle of radius $R = 0.5$ centred at the point $C(0,0)$. (Fig 5.14). The system is then simulated for $T_{sim} = 0.7$ seconds and time snapshots are taken every $\Delta t_{plot} = 0.1$ seconds (Fig 5.16) and the performance is tracked through the three performance measures defined in the previous section (Fig. 5.15 – Fig. 5.18).

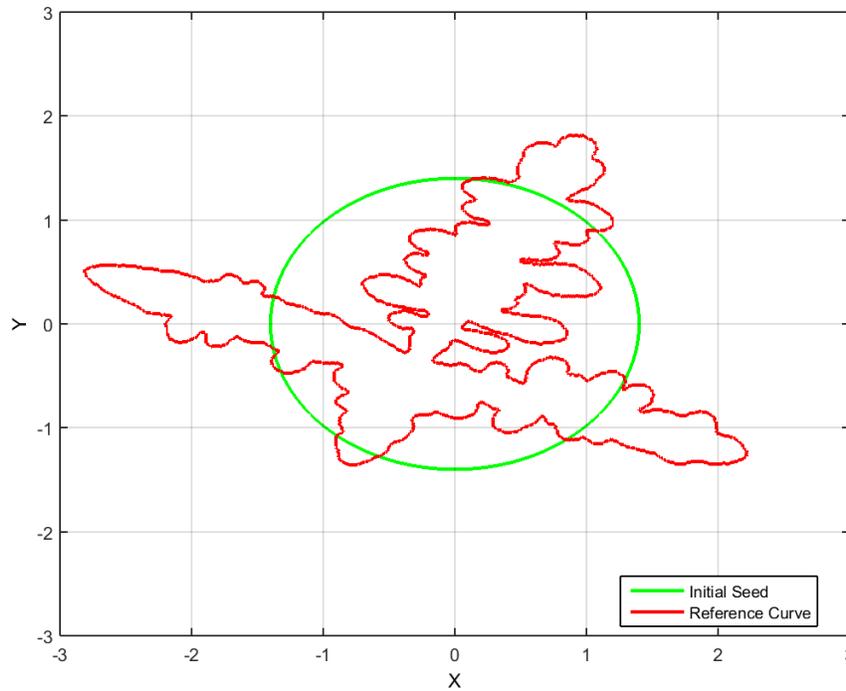


Fig. 5.14. Initial control setup using NH_4Br dendrite crystal reference

The accuracy of numerical discretization is a key factor in the control method, and is linked with the complexity of the target interface. For simple interfaces the arc length and area errors are very close to zero, meaning that the final shape has reached the desired reference. The effect of numerical discretization is more noticeable when the complexity of the boundary is increased. For the dendritic reference shape the reference properties are described by

$$Arclength(\Gamma_r) = 28.4855$$

$$Area(\Omega_r) = 4.933$$

The final values for E_{arc} and E_{area} in the controlled system are non-zero, even though relatively fine grid resolution was used. Nonetheless, the final area and the arc length were tracked to within 0.3% and respectively 2.3% of their reference values. By increasing the resolution of the representation (increasing the number of grid points, lowering the spatial step size), the crystal boundary is better represented, and the approximation errors will converge towards zero.

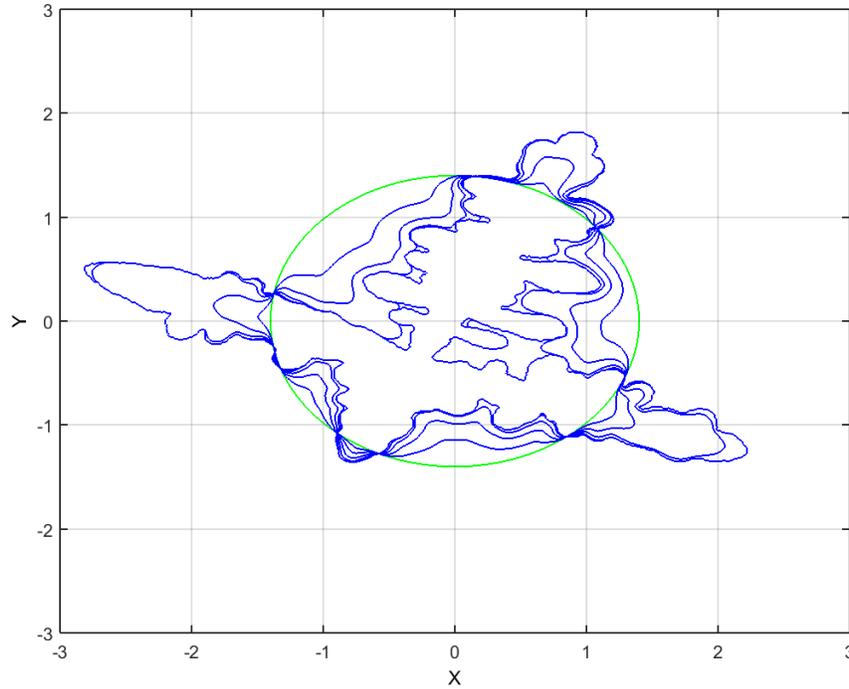


Fig. 5.15. Time evolution of the controlled system using NH_4Br dendrite crystal reference. Time plots are shown in increments of $\Delta t_{plot} = 0.1$ seconds

5.6.3. Controller performance comparison

In this section an evaluation of the performance of the proposed method using different control strategies, namely P, PI and PID in a two dimensional space is presented. The evolution is viewed within the previously used domain $\Omega: [-3, 3] \times [-3, 3]$ and the numerical grid resolution is set to 300×300 points, giving a spatial step size of $\Delta x = \Delta y = 0.02$. The analysis is performed using as a reference a relatively simple interface in the form of a six point star, together with the initial curve Γ_0 in the form of a circular seed with radius equal to $R = 1.4$ (Fig. 5.19). The model parameters $a_i, i = \overline{0,4}$ remain the same as in section 5.6.1. The controller tuning was performed empirically, and the model then simulated for $T_{sim} = 0.5$ seconds. The controller gains are set to be $K_p = 5, K_i = 10, K_D =$

$5 \cdot 10^{-4}$. By using the previously defined global error G_{error} , a measure of the performance capability between controller types can be defined as:

$$G_{P-PI}(t) = G_{error}^P(t) - G_{error}^{PI}(t) \quad (5.28)$$

$$G_{P-PID}(t) = G_{error}^P(t) - G_{error}^{PID}(t) \quad (5.29)$$

$$G_{PI-PID}(t) = G_{error}^{PI}(t) - G_{error}^{PID}(t) \quad (5.30)$$

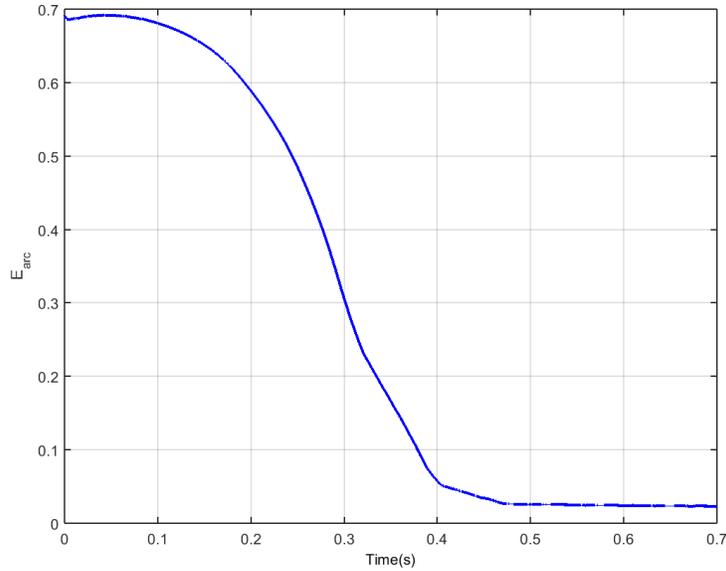


Fig. 5.16. Arc length error NH_4Br dendrite crystal reference

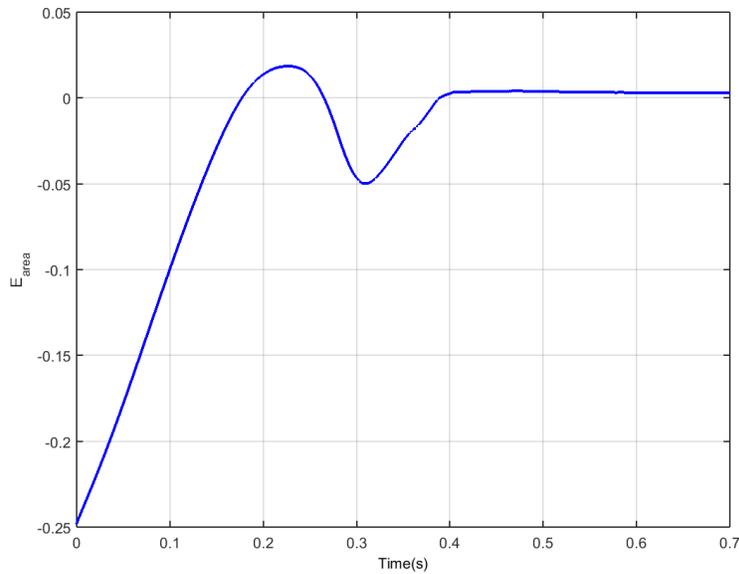


Fig. 5.17. Area error NH_4Br dendrite crystal reference

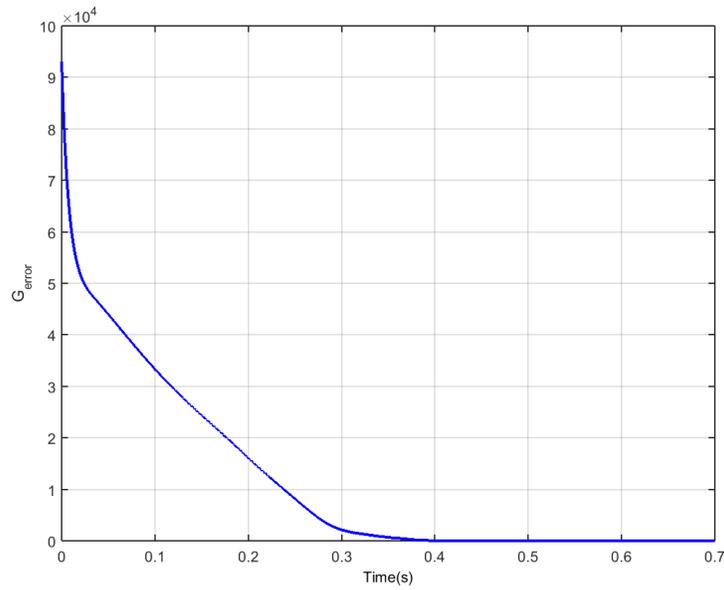


Fig. 5.18. Global error NH_4Br dendrite crystal reference

where $G_{error}^P, G_{error}^{PI}(t), G_{error}^{PID}(t)$ are the absolute global errors obtained through simulation using the respective control strategy (P, PI and PID). The comparative performances of the formulated problem are shown in Fig. 5.20. Overall, the PID controller has a marginally better performance than the PI and P controller configurations, demonstrating the capability of the derivative term.

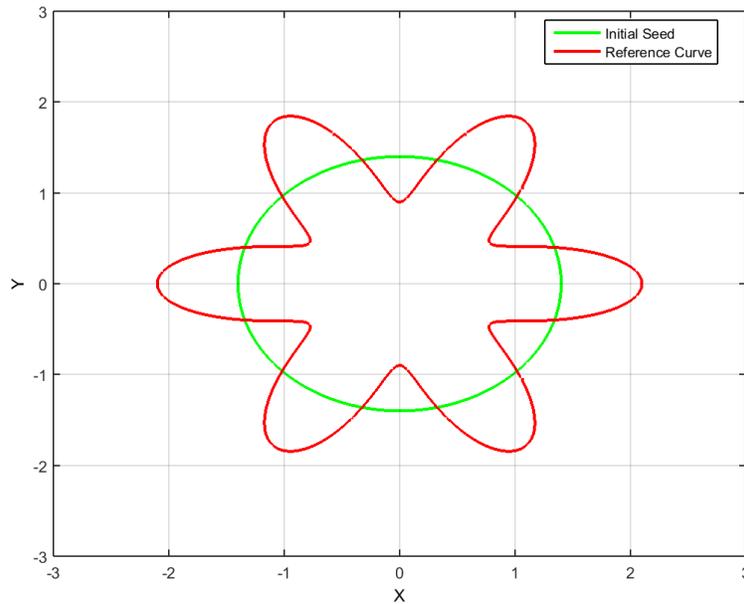


Fig. 5.19. Controller comparison initial setup

It can be seen that the control method is more sensitive to numerical discretization when shapes with a high degree of complexity are involved. Although better results are obtained with large spatial resolutions, extra effort is needed in numerical computations. The time constant Δt used for numerical integration used when discretizing equation (5.8) needs to be lowered in order to insure stability, which results in longer simulation times.

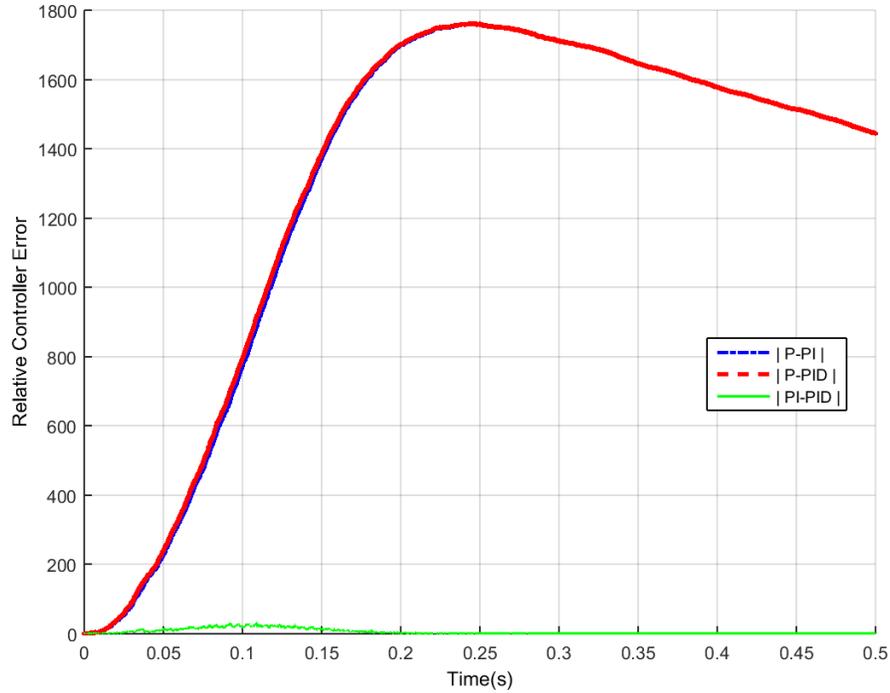


Fig. 5.20. Response comparison of P, PI and PID controllers

5.7. Conclusions

This chapter introduced a novel distributed method for controlling the evolution of a solid-fluid interface, such that of a dendritic crystal, using temperature dependent geometrical model. The free interface (crystal boundary) is represented using a level set formulation. The closed loop control method is also formulated using a level set approach and was presented for both two and three dimensional cases, with the temperature taken to be the manipulated variable.

Numerical simulations with synthetic data were performed, using a generic geometrical model of interface evolution, in both two and three dimensional spaces. A number of error measures were defined to evaluate the performance of the control strategy. Experimentally obtained data was analyzed and the complex spatio-temporal pattern of a NH_4Br dendrite crystal was successfully used to validate

the new approach. The performance and effectiveness of P, PI and PID control strategies was analyzed, using a relatively simple reference shape.

The proposed control method could be implemented in practice using laser arrays (Shuja and Yilbas, 2012) which allow controlling the temperature around the boundary of the evolving interface, or by using a boundary control approach to achieve a desired temperature distribution inside the domain. Time-lapse imaging, segmentation and tracking algorithms could be used to observe the position of the interface in real-time.

6. A model-based control approach for managing wildland fires

6.1. Introduction

Wildfire modelling and simulation approaches provide the basis for understanding and predicting the spread of wildfires, as well as for implementing decision support systems to help minimize their environmental, economic and health impacts (Martell, 2001). However, the underlying processes governing the dynamics of wildfires are highly complex and nonlinear, which makes it difficult to predict and control their evolution (Rehm and McDermott, 2009).

Historically, there are several established approaches to managing the load of wildland fires (Martell, 2001), such as fuel management, fire prediction and prevention or fire detection. Each approach actively contributes to minimizing the impact of occurring fires to both biological and economic sectors. The fuel management aims to alter the biofuel density and composition around the fire such that the fire spread undergoes a damped evolution, which will lead to full extinction. The fire prevention and prediction component is targeted mostly at reducing the rate of recurrence of human triggered fires, but also fires caused by natural phenomena, such as lightning. Fire detection is a crucial component, since a timely finding of an evolving fire (ideally as soon as vegetation ignition occurs) enables relevant firefighting authorities to better assess the situation, prioritize against existing risk factors and prepare a more efficient strategy to extinguish the fire. Depending on the location of high fire risk areas (populated or remote areas etc.), several detection approaches might be employed. Often, infrared fitted sensing devices (satellites, look-out towers, patrol aircraft etc.) are used to this scope.

A vast range of wildfire spread models have been proposed in literature (Sullivan, 2007a and 2007b). This chapter focuses on an approximate quasi-empirical model of wind-aided fire spread, proposed by Fendell and Wolff (2001) and refined further by Mallet et al. (2009). Specifically, a novel model-based approach for controlling the evolving complex-shaped fire front using a level-set formulation of the model is proposed. A derivation of how key parameter functions such as terrain topography and fuel loading, extracted directly from satellite imaging data, can be incorporated in the model is presented in detail.

The numerical method is computationally efficient and examples are given with both synthetic and real data to illustrate the robustness of the containment method against changing wind speed, fuel loading and terrain topography. The Chapter is organized as follows. Section 6.2 introduces a novel model-based containment method based on a level set formulation aimed at managing wild land fires. The third part,

Section 6.3 is dedicated to the numerical analysis of the proposed method. Artificially generated data (terrain properties, atmospheric conditions etc.) is used to demonstrate the containment method. Using terrain elevation models, a data fusion method is presented, that easily allows real time, measured data to be incorporated into the control strategy.

6.2. Model-based containment of wildfires using geometrical models of interface evolution

Let the moving fire front boundary be described by a time varying closed planar curve $\Gamma(t)$. Following the description in Chapter 4, consider the following nonlinear evolution model that gives the rate of propagation for the fire front, over a domain of interest $\Omega \subset \mathbb{R}^2$:

$$F(\bar{x}_\Gamma(t), \bar{v}(\bar{x}_\Gamma, t), \theta_\Gamma(t)) = \begin{cases} \varepsilon_0 + a(\bar{x}_\Gamma, t) \sqrt{|\bar{v}(\bar{x}_\Gamma, t)| \cdot \cos(\theta_\Gamma(t))^3}, & |\theta_\Gamma| \leq \frac{\pi}{2} \\ \varepsilon_0(\alpha + (1 - \alpha)|\sin(\theta_\Gamma(t))|), & |\theta_\Gamma| > \frac{\pi}{2} \end{cases} \quad (6.1)$$

where $\bar{v}: \Omega \times [0, T] \rightarrow \mathbb{R}_+^2$ is the wind velocity field, $\theta_\Gamma: [0, S(t)) \times [0, T] \rightarrow [-\pi, \pi]$ denotes the angle between the wind direction and the fire front normal measured in an anticlockwise direction along the boundary $\Gamma(t)$, $\varepsilon_0 \in \mathbb{R}_+$ is the front velocity at the flanks, $\alpha \in [0, 1]$ is the ratio between the velocity at the rear at the flanks of the fire front and $a: \Omega \rightarrow \mathbb{R}_+^2$ describes the spatial distribution of fuel loads, which can be estimated in practice using high-resolution synthetic aperture radar (Reigber et al., 2013).

Assuming that all relevant data is available (measured, observed or estimated), the goal is to derive a control strategy such that at any time t after the fire has been detected, the fire front $\Gamma(t)$ is confined inside a “safe zone”, i.e. the fire front evolution is suppressed and contained. Let the domain in which the fire grows (is observed) to be a bounded, rectangular domain with $\Omega: [-l, l] \times [-w, w]$, $l, w \in \mathbb{R}_+$. In practice, the safe zone contour could be the urban–wildlife limit which could contain forests, fields, rural or urban settlements etc. To prevent the fire from reaching any area within Ω , a “safe zone” contour Γ_r is introduced, in which the fire may be allowed to possibly evolve, without causing significant damage (i.e. the fire is not allowed to reach any area outside Γ_r).

Let $\varphi_r(x), \varphi_r: \Omega \times [0, T] \rightarrow \mathbb{R}$ and $\varphi_\Gamma(x, t), \varphi_\Gamma: \Omega \times [0, T] \rightarrow \mathbb{R}$ be the level set functions attached to with Γ_r and $\Gamma(t)$ respectively, with $T \geq 0$ representing the maximum time window. Using equation (3.22), the fire evolution model (6.1) can then be written in level set formulation as:

$$\begin{cases} \frac{\partial}{\partial t} \varphi_\Gamma + \left(\varepsilon_0 + a(\bar{x}_\Gamma, t) \sqrt{|\bar{v}(\bar{x}_\Gamma, t)| \cdot \cos(\theta_\Gamma(t))^3} \right) \cdot |\nabla \varphi_\Gamma| = 0, & |\theta| \leq \frac{\pi}{2} \\ \frac{\partial}{\partial t} \varphi_\Gamma + \left(\varepsilon_0(\alpha + (1 - \alpha)|\sin(\theta_\Gamma(t))|) \right) \cdot |\nabla \varphi_\Gamma| = 0, & |\theta| > \frac{\pi}{2} \end{cases} \quad (6.2)$$

Let $\Omega_{\Gamma_r} \subset \mathbb{R}^2$ be the zone enclosed by Γ_r , i.e. the subset where $\varphi_r < 0$. A control function $\bar{v}_c(x, \varphi_r, m)$ can be expressed as:

$$\bar{v}_c(x, \varphi_r, m) = \begin{cases} -\frac{|\varphi_r(x)|^m}{\max(|\varphi_r(x)|^m)} + 1, & x \in \Omega_{\Gamma_r} \\ 1, & x \in \mathbb{R}^2 \setminus \Omega_{\Gamma_r} \end{cases}, m \in \mathbb{R}^+ \quad (6.3)$$

where the parameter m is a tunable parameter, used to define the amount of available firefighting resource available within the observed domain Ω . Having $\bar{v}_c(x, \varphi_r, m)$, a feedback control law for the fire front velocity can be defined as:

$$F_c(\bar{x}_\Gamma(t), \bar{v}(\bar{x}_\Gamma(t), \theta_\Gamma(t), \bar{v}_c(\bar{x}_\Gamma(t), \varphi_r, m))) = [1 - \bar{v}_c(\bar{x}_\Gamma(t), \varphi_r, m)] \cdot F(\bar{x}_\Gamma(t), \bar{v}(\bar{x}_\Gamma(t), \theta_\Gamma(t))) \quad (6.4)$$

By applying the fire fighting agent on the boundary, the effective front temperature is gradually reduced, until the full extinction of the evolving fire. Using (6.4), the evolution equation (6.2) can be written in controlled form as:

$$\begin{cases} \frac{\partial}{\partial t} \varphi_\Gamma + T[1 - \bar{v}_c(\bar{x}_\Gamma(t), \varphi_r, m)] \cdot \left(\varepsilon_0 + a(\bar{x}_\Gamma(t), \sqrt{|\bar{v}(\bar{x}_\Gamma(t), \theta_\Gamma(t))|^3}) \cdot |\nabla \varphi_\Gamma| = 0, & |\theta| \leq \frac{\pi}{2} \\ \frac{\partial}{\partial t} \varphi_\Gamma + T[1 - \bar{v}_c(\bar{x}_\Gamma(t), \varphi_r, m)] \cdot \left(\varepsilon_0(\alpha + (1 - \alpha)|\sin(\theta_\Gamma(t))|) \right) \cdot |\nabla \varphi_\Gamma| = 0, & |\theta| > \frac{\pi}{2} \end{cases} \quad (6.5)$$

where T represents the temperature of the fire front. The effects of selecting different values for the tunable variable m is shown in (Fig. 6.1) where Γ_r is set to be a square region of side length $L = 500$, within a domain $\Omega: [0,600] \times [0,600]$.

The control law (6.4) is designed based on a level set formulation of the model such that the fire evolution is stopped within a predefined containment area by actively controlling the temperature of the fire front. To fully define the controlled equation (6.5), the following statements are taken to hold throughout the domain Ω :

- Direct wildfire suppression treatments (wetting, cooling, smothering, chemical quenching) are applied to burning fuel along the fire perimeter to slow down the front velocity. The optimum treatment can be selected depending on the individual environmental setup.
- No bounds are imposed on the control $\bar{v}_c(\bar{x}_\Gamma(t), \varphi_r, m)$, i.e. the control can be applied everywhere on the fire front and unlimited amounts of resources are available for fire combat.
- The resource needed is available without any time delay

The set of equations (6.5) describe a model based containment control method of the geometrical model of fire front evolution (6.1), where the interface velocity is influenced by wind speed, wind direction, biofuel density and the applied control $\bar{v}_c(\cdot)$. The model (6.5) is in fact a Hamilton-Jacobi equation:

$$\varphi_t + H(\nabla \varphi) = 0 \quad (6.6)$$

with a nonlinear Hamiltonian $H(\nabla\varphi)$. The discretization and numerical simulation approaches described in Chapter 4, Section 4.3 were used in the next sections of this chapter.

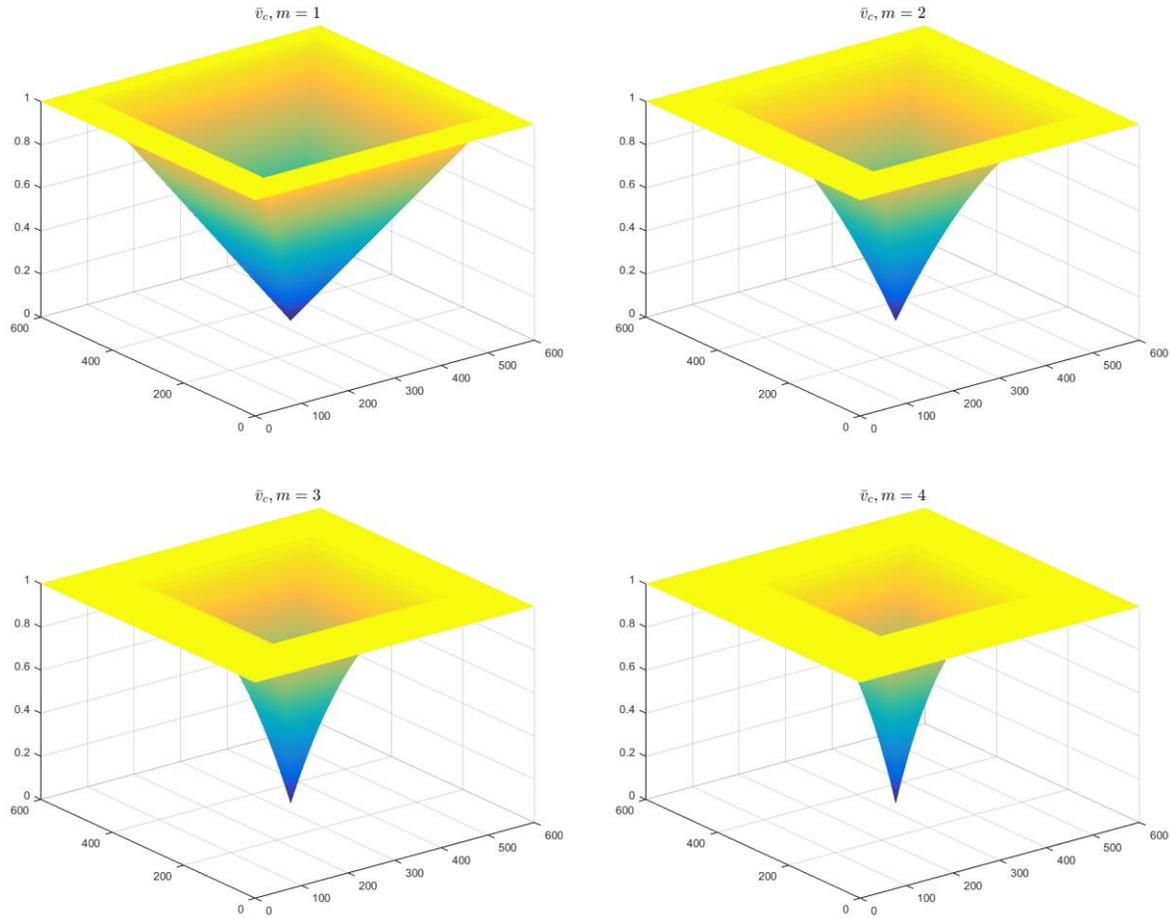


Fig. 6.1. Profiles obtained by varying the power term m

6.3. Numerical implementation of the wildfire suppression method

This section presents examples of applied control of the fire interface, using the containment method previously defined. In this dissertation both synthetic and real data are used to illustrate the proposed method. The method is highly robust to changes in a wide range of influencing parameters such as atmospheric conditions (wind speed and wind direction), fuel density or terrain topography. For real data simulations, satellite terrain images are analyzed and an estimate of the fuel loading is then derived. Terrain topography maps are created for various regions using a series of terrain elevation matrices in the form of DTED (Digital Terrain Elevation Data) information files. These files are processed to obtain representative terrain elevation models, which are then used to estimate the terrain slope in the control scenarios.

6.3.1. Synthetic data simulations

To illustrate the proposed fire suppression approach, several cases are presented, representing controlled and uncontrolled scenarios, using artificially generated data. For simplicity let us assume that the evolving interface is described by simple, non-intersecting closed curve, i.e. no ‘islands’ are formed throughout the burning process. The initial fire front, observed at detection, is therefore set to be circular seed (I_0) of radius $R = 1$, centered within the observation domain Ω . The shape of the safety zone (I_r) is defined square region of side length $L = 60$, also centered within Ω (Fig 6.2). The two dimensional space in which the growth is observed is set as the square $\Omega: [-40, 40] \times [-40, 40]$, which is discretized on a uniform Cartesian grid with resolution of 300×300 points.

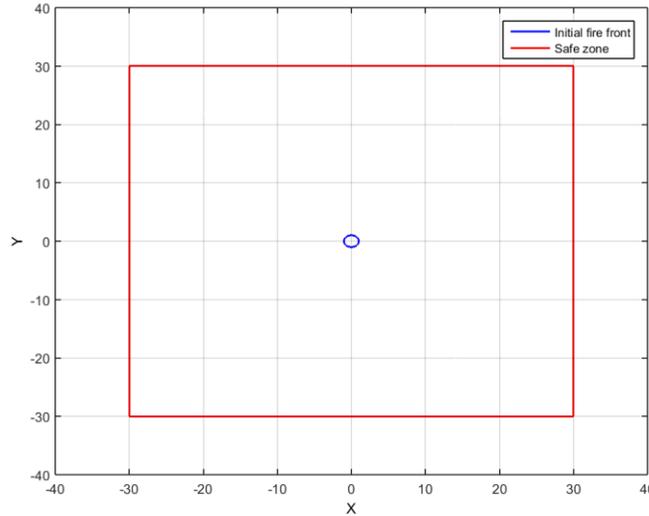


Fig. 6.2. Initial setup for wildfire evolution

Assuming a uniformly distributed fuel loading ($a(x) = 0.5$) over a flat terrain, under constant wind speed and wind direction, the fire advances rapidly, and eventually passes the safety zone boundary (Fig. 6.3.). Here it is assumed that the wind is blowing in an S-N direction under $\bar{v} = 1$, with south bearing being at the bottom of the graph. The velocity at the flanks is taken to be $\varepsilon_0 = 0.2$ and the velocity ratio $\alpha = 0.5$. Under the same topographical and atmospheric conditions, but with the applied distributed suppression method ($m = 4$), the evolution of the fire front is greatly damped, taking a significantly longer time to reach the designated safety perimeter (Fig. 6.4), and eventually leading to full extinction. This allows more time for evacuation purposes, development of new fire suppression strategies etc.

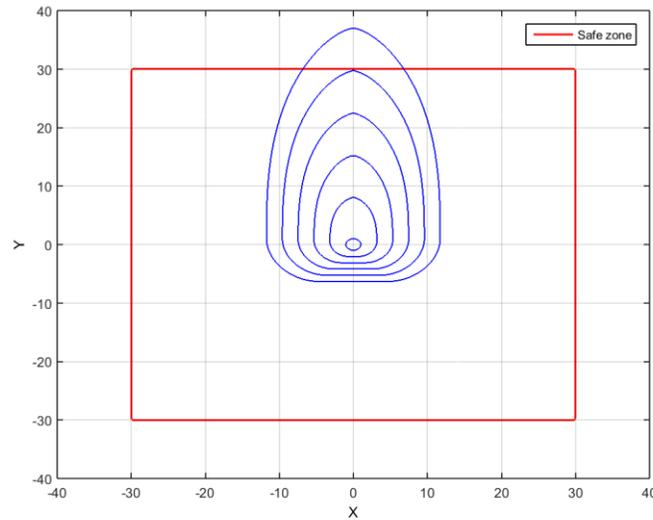


Fig. 6.3. Uncontrolled wildfire evolution

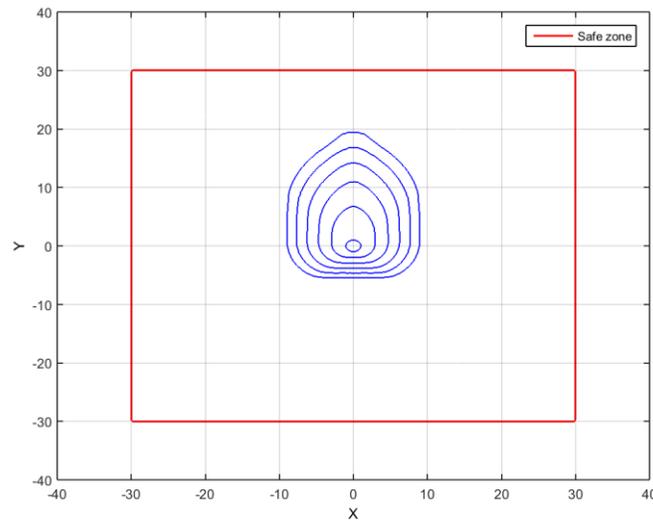


Fig. 6.4. Controlled wildfire evolution

In the event of fire spotting, several fire fronts might be growing in parallel, and the wind might favor ignition events closer to the safety zone. Since the suppression method is based on actively analyzing the region within a predefined boundary, the control agent will be applied to each front independently, meaning that fronts closest to the boundary will be given a higher importance. To illustrate this scenario consider the setup shown in Fig. 6.5. Through the proposed scenario, the atmospheric conditions are altered as follows: the wind speed is increased gradually, each 0.2 seconds, varying from 1 to 16; the wind direction is initially taken to be the S-N heading, and gradually rotating towards N-E heading by 5° each 0.2 seconds. The fuel loading is considered constant throughout the whole domain.

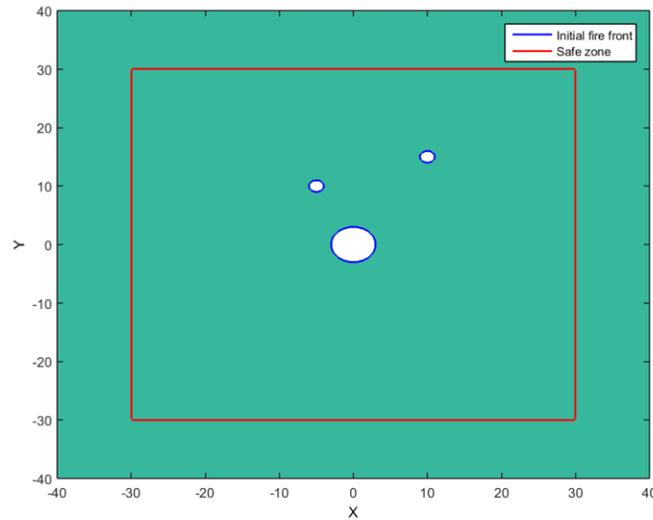


Fig. 6.5. Multiple wildfire detection

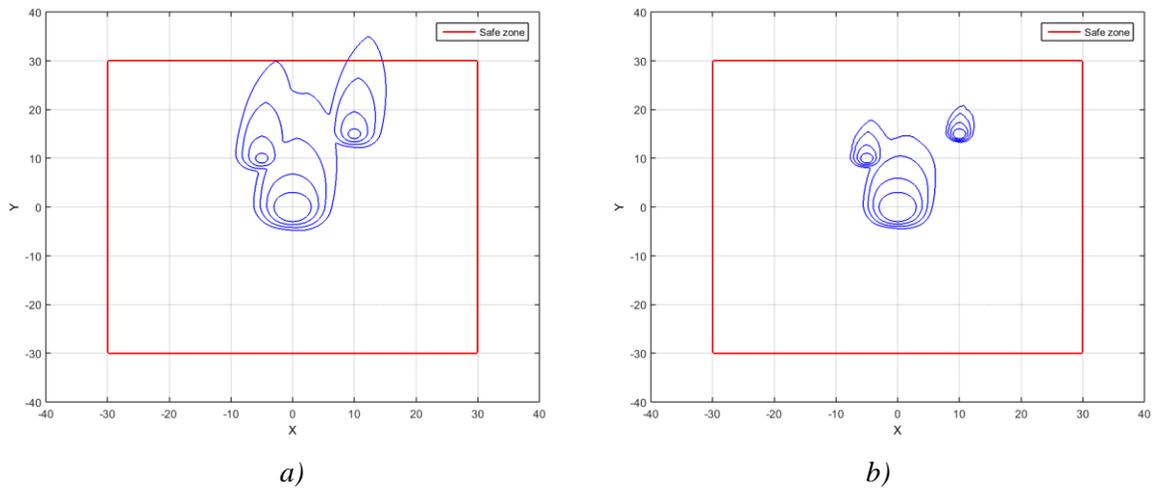


Fig. 6.6. Multiple wildfires – a) uncontrolled evolution b) suppression active

It can be seen that both fronts are actively managed, and due to the fact that the smaller spot fires that are ahead of the primary fire have consumed significantly less fuel, and the main fire front can be put out without any extraordinary effort due to its decelerated growth rate.

6.3.2. Satellite terrain data simulations

In most forest fire scenarios, timely detection is crucial in formulating a valid firefighting strategy. We have seen from the previous section that the suppression method is highly effective if the parameters incorporated in the model can be obtained, so that an accurate prediction of the front dynamics can be achieved. If the properties of the observed area are known a priori, as in the case of experimental man

made fires, then the suppression method presented earlier can be applied easily (under the hypothesis that an unlimited amount of fire suppressing agent is readily available). In real situations, the properties of the fuel bed, atmospheric conditions and the terrain topography are usually unknown beforehand, and need to be acquired as soon as an event is discovered. In this section, a method to exploit satellite imaging data to estimate key properties of the terrain, such as fuel loading or terrain elevation, which are then incorporated in the controlled model (6.5), is proposed and demonstrated.

The atmospheric conditions could be estimated using on site tools or by using geostationary (GOES) or polar orbiting (POES) meteorological satellites (Bevis et al., 1992, Tang et al., 2008, Griffith et al., 1978, Kidd et al., 2003). The POES satellites usually provide much more accurate data than their geostationary counterparts due to the lower altitude at which they are orbiting the Earth ($\approx 850 \text{ km}$ above sea level). These special satellites provide weather information by observing the Earth at several wavelengths of the electromagnetic spectrum. Depending on the information of interest, the spectral bands might be in the visible range to NIR (near infrared) $[0.6, 1.6]\mu\text{m}$ for cloud monitoring, infrared band $[3.9, 7.3]\mu\text{m}$ which is used to monitor water vapor (and gas) absorption, and long wave infrared $[9, 12]\mu\text{m}$ which is used for thermal imaging (Atlantic region, Europe and Africa – illustrated in Fig. 6.7 to Fig. 6.9 – courtesy of <http://tropic.ssec.wisc.edu/tropic.php>).

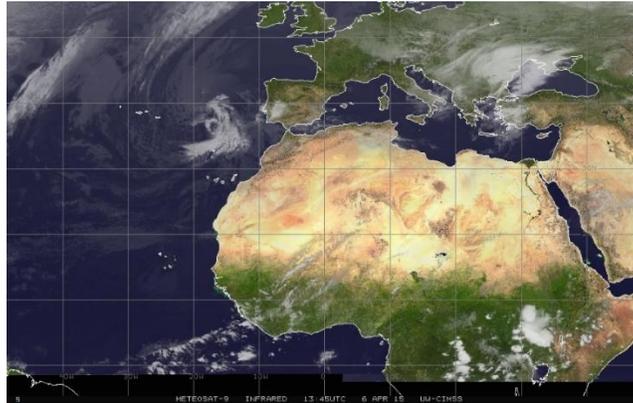


Fig. 6.7. Region of interest – Color background view

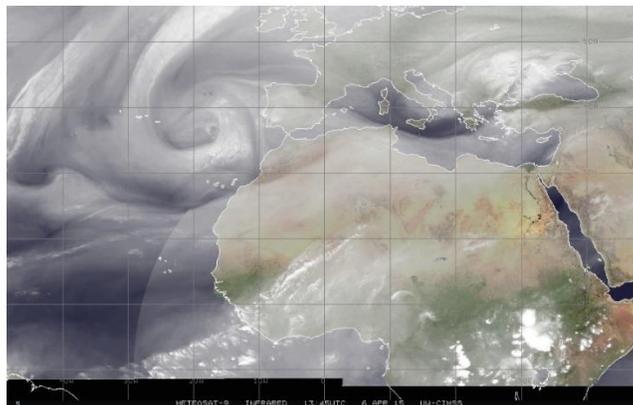


Fig. 6.8. Region of interest – Water vapor view

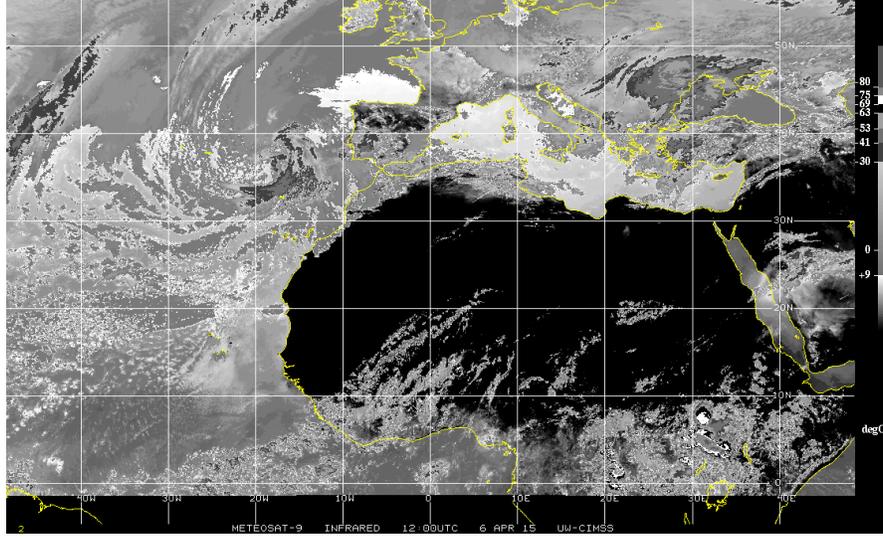


Fig. 6.9. Region of interest – Infrared enhancement view

To demonstrate the suppression method using real data, several satellite images of terrains (Fig. 6.10) - courtesy of Google Maps, containing various fuel loading distributions were used.

The selected terrain images are captured in the additive RGB color model. The maps for each picture are represented by a three dimensional matrix of 421×421 , 502×502 and 514×514 pixels respectively. The images are used to estimate the fuel loading map, that is then incorporated in the fire evolution model. Since not all the information contained within the snapshots is relevant for control purposes, a color based segmentation algorithm in the RGB color space is used (Selvarasu et. al., 2010). It is natural to consider that each pixel $P(i, j)$ in a snapshot is defined by the color intensity vector $C_{ij}^{RGB} = [C_{ij}^R \ C_{ij}^G \ C_{ij}^B]$, which contains the three values associated with the red, green and blue components. For the selected snapshots, the sampling depth was taken to be 8 bits, which provides adequate spatial continuity in the image texture. In the current case, the color of interest is pure green, which can be represented in the RGB space by the vector $C^{0G0} = [0 \ 255 \ 0]$ or simply $C^G = 255$. To identify if a pixel represents a green value, a typical Euclidean distance $d(q, p)$ between C_{ij}^{RGB} and C^{0G0} is calculated for each pixel $P(i, j)$ from:

$$d(C_{ij}^{RGB}, C^{0G0}) = \sqrt{\sum_{p=1}^3 (C_{ij}^{RGB}(p) - C^{0G0}(p))^2} \quad (6.7)$$

a) Terrain 1: <http://goo.gl/maps/RoOuV>b) Terrain 2: <http://goo.gl/maps/LFBUL>c) Terrain 3: <http://goo.gl/maps/7VDhM>*Fig. 6.10.* Satellite acquired terrain data

Then the pixel $P(i, j)$ is categorized as being green if the value $d(C_{ij}^{RGB}, C^{0G0})$ is below a predefined threshold d_g . After the ‘green’ pixels have been detected (i.e. the regions with alive, burnable vegetation), the information needs to be incorporated into equation (6.5). The fuel loading map a is constructed by associating the ‘greenness’ of each pixel with discrete fuel loading value. Therefore, if $d(C_{ij}^{RGB}, C^{0G0}) < d_g$ the discrete location (i, j) on the discrete fuel loading map will be assigned a value of 1, otherwise it

will be assigned a value of 0. By applying (6.7) to the snapshots shown in Fig. 6.10 a, b, c, and assuming a threshold value of $d_g = 0.3$, the binary fuel loading maps are obtained (Fig. 6.11 a, b, c). The maps created contain a diverse range of fuel loading, with Terrain 1 being covered with the most biofuel, Terrain 2 with a low amount of green fuel, and Terrain 3 somewhere in between the first two.

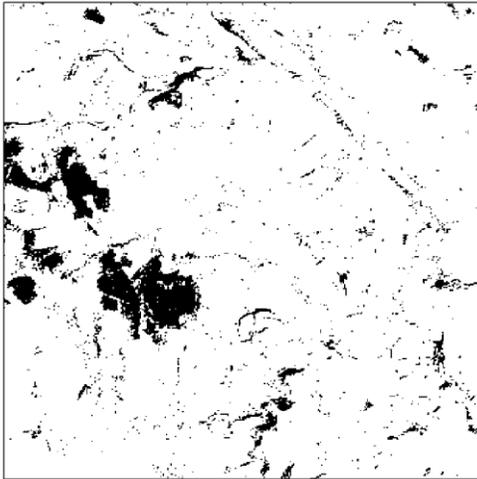
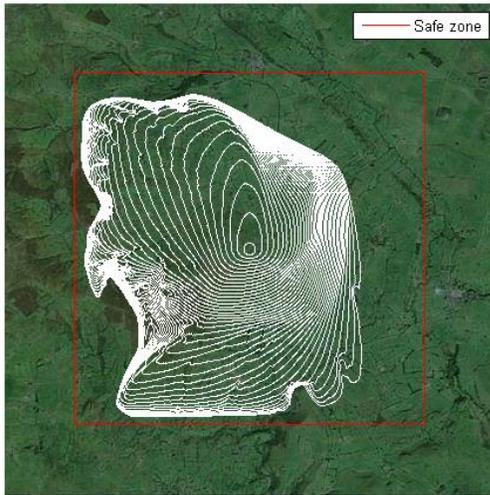
*a)**b)**c)*

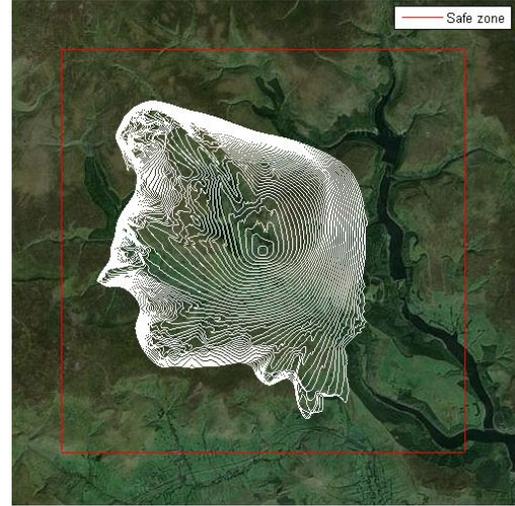
Fig. 6.11. Fuel loading map – a) Terrain 1, b) Terrain 2, c) Terrain 3

The above binary fuel loading maps can now be directly used in the model (6.5). To this regard, a circular initial fire front was placed at the center of each map. The atmospheric conditions are taken to vary

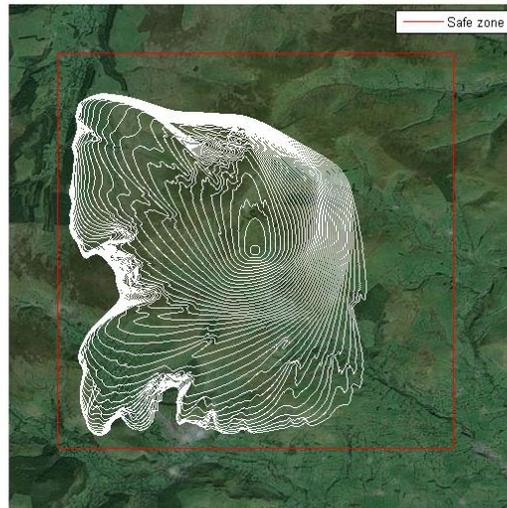
rapidly throughout the life of the fire front. The wind speed is set to gradually increase over the interval $[1, 81]$, over a period of $T_{sim} = 1200$ seconds, whilst undergoing a rapid change of direction from the S-N heading, towards N-W heading by 5° each 30 seconds. The tuning parameter in (6.3) is taken to be $m = 2$. In each case, the safe zone considered is represented by a square region of suitable dimensions. Simulation results are shown in Fig. 6.12.



a)



b)



c)

Fig. 6.12. Wildfire evolution under suppression agent, level sets are shown in time increments of $\Delta t_{plot} = 30$ seconds – a) Terrain 1, b) Terrain 2, c) Terrain 3

Under the described scenarios, it can be seen that the evolution of wind aided wildfires is, as expected, favored in regions with higher fuel density (*Fig. 6.12a*) as opposed to regions with low amounts of fuel (*Fig. 6.12b*). However, due to the use of the fire suppression agent along the evolving interface, the growth is slowed down and doesn't exceed the designated safe zone.

6.3.3. Terrain elevation from DTED models

To accurately describe the terrain topography a measure of the terrain elevation is needed. The Digital Terrain Elevation Data (DTED) models represent a series of uniformly sampled spatial matrices that contain elevation data, which provide valuable numerical information for numerous applications, such as map generation, relief maps, GPS navigation, various flight simulators, terrain slope estimation or terrain roughness estimation. The data was gathered via satellites, by measuring signals reflected from the Earth's surface, and analyzed to provide numerical values of local elevation. Initially, the elevation models were generated by the Defense Mapping Agency (DMA) and are currently provided by the U. S. National Geospatial Intelligence Agency (NGA) (DMA, 1996). Several DTED models have been created (Table 6.1.):

- DTED-0 (level 0 maps) – this is the lowest level of fidelity model, with the grid spacing between data points being 30''(arc-seconds) apart ($\approx 1Km$). Currently, this is the only elevation model available to the general public.
- DTED-1 (level 1 maps) – contains medium resolution data, with the grid spacing being 3'' ($\approx 100 m$), and are mainly used in military based applications, and are not available to the general public.
- DTED-2 (level 2 maps) – contain very high resolution terrain information, with the data points being sampled at 1'' ($\approx 30 m$), and are available only for military activities.

DTED level	Post Spacing	Ground distance	Resolution (for 1° Lat \times 1° Long surfaces)
DTED Level 0	30 arc-sec	$\approx 1Km$	120×120
DTED Level 1	3 arc-sec	$\approx 100 m$	1200×1200
DTED Level 2	1 arc-sec	$\approx 30 m$	3600×3600

Table 6.1. Digital Terrain Data accuracy

In this study, the DTED level 0 data was used to construct an estimate of the observed terrain slope that was incorporated in (6.5) in order to illustrate the robustness of the proposed suppression method on

varied terrain topography. The information contained in the DTED files is represented as elevation points in meters, within cells defined by latitude and longitude indexes or a geographic reference system. First, a region from the north-eastern hemisphere is selected, contained between the following latitude and longitude reference points:

$$Reg_{observed} : \begin{cases} [44.9958^\circ, 46.0042^\circ] N \\ [23.9958^\circ, 25.0042^\circ] E \end{cases}$$

which is a mountainous regions. After preprocessing the raw data a 3D mapping of the region is obtained, showing the peaks of altitude – the elevation map $Z_{observed}$ (Fig. 6.13), which contains values between $Z_{min} \approx 161$ m and $Z_{max} \approx 2436$ m:

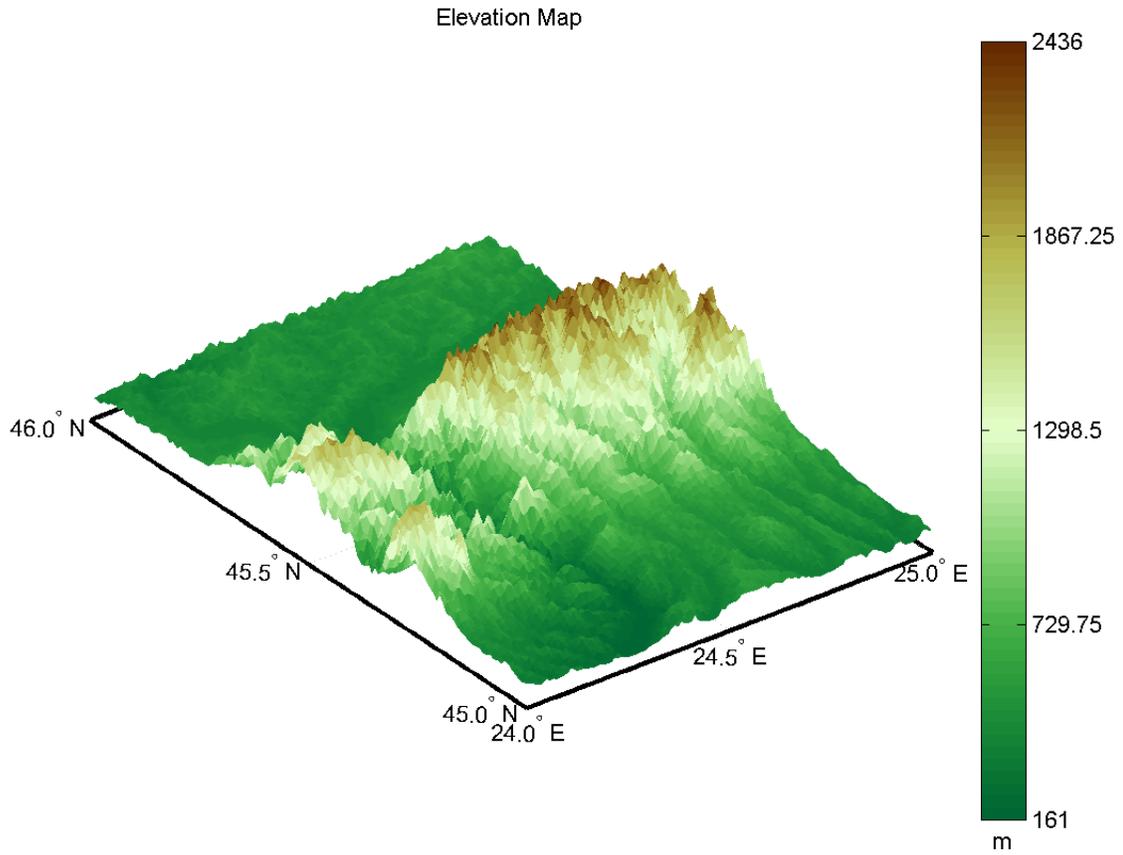


Fig. 6.13. Terrain elevation map

Based on the elevation map obtained from DTED data, the slope of a point was calculated as

$$s = \tan \left(\sqrt{\left(\frac{\partial Z}{\partial x}\right)^2 + \left(\frac{\partial Z}{\partial y}\right)^2} \right)^{-1} \quad (6.8)$$

where Z is the terrain elevation and the derivatives $\frac{\partial Z}{\partial x}$ and $\frac{\partial Z}{\partial y}$ were approximated over a 8-cell neighborhood (Fig 6.14), using the contour spacing Δ :

$$\frac{\partial Z}{\partial x} = \frac{Z_5 - Z_4}{2\Delta} \quad (6.9)$$

$$\frac{\partial Z}{\partial y} = \frac{Z_2 - Z_7}{2\Delta} \quad (6.10)$$

Using (6.8), the resulting slope angle, expressed in radians, can be estimated from the terrain elevation map $Z_{observed}$ and is depicted in Fig. 6.15.

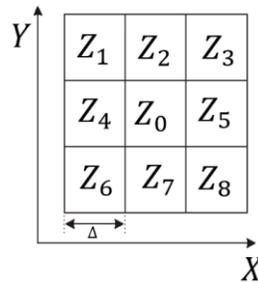


Fig. 6.14. Neighborhood of 8 points used to estimate the slope of the central point Z_0

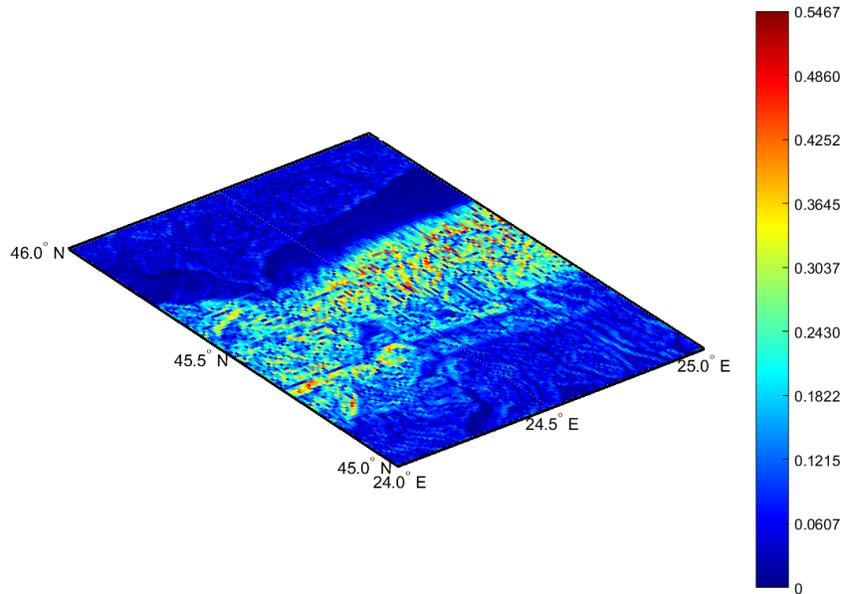


Fig. 6.15. Terrain slope estimate of $Reg_{observed}$

Until now all the information processed was derived from a-priori known elevation information contained in the DTED level 0 models. It is well known that terrain topography has a relatively low time dynamic, and landmarks are usually representative for long periods of time. To add new information to the model, for the purpose of robustness, satellite imaging data needs to be used. Following the steps of

previous section, a snapshot of the terrain was acquired and processed to obtain an estimate of the fuel loading within the selected area (Fig. 6.16 and Fig 6.17). Having now an estimate of all relevant data, the terrain slope information and fuel loading can be directly incorporated into the nonlinear evolution dynamics, creating a data fusion model.



Fig. 6.16. Terrain topography satellite image – <http://goo.gl/maps/uc1D8>

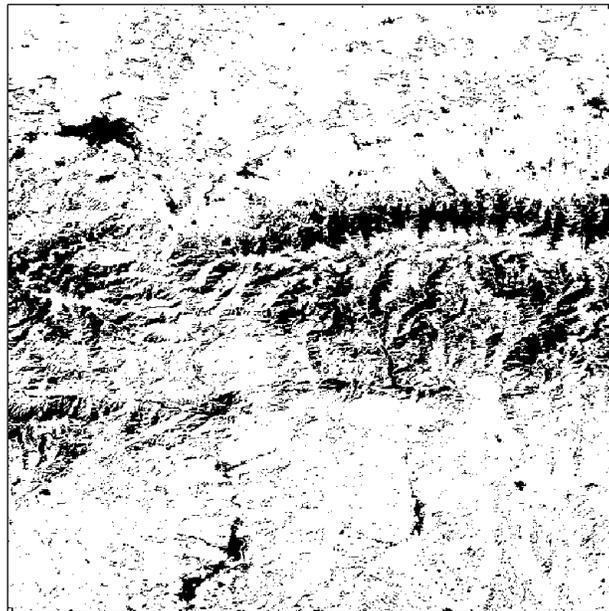


Fig. 6.17. Terrain topography – fuel loading map

Next an initial fire front was placed within the observed region, and a predefined safety zone delimited by a square was introduced to define the initial control problem. The wind is set to blow in the S-N direction

with a magnitude of $\bar{v} = 10$. The flank velocity is influenced by $\varepsilon_0 = 0.2$ and the rear/flank velocity ratio $\alpha = 0.5$. The evolution of an initial fire front was observed under natural evolution (Fig.6.18) and under the action of a fire suppression strategy present (Fig. 6.19), with $m = 4$.

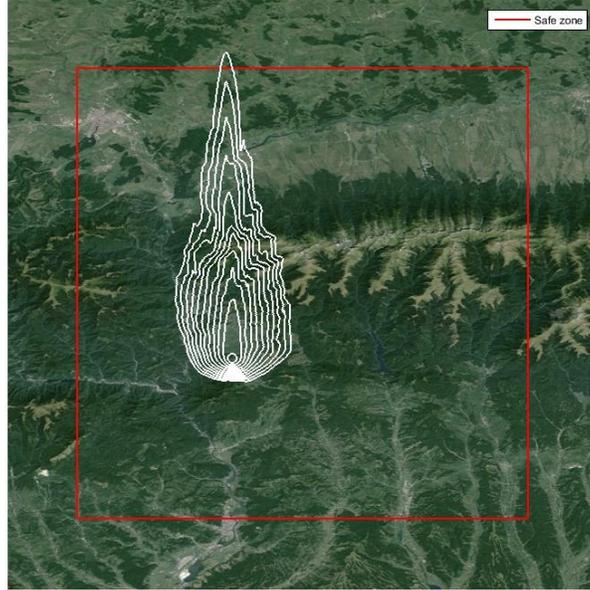


Fig. 6.18. Unconstrained wildfire evolution for the data fusion model. Time plots are shown in increments of $\Delta t_{plot} = 15$ seconds until final time $T_{sim} = 150$ seconds

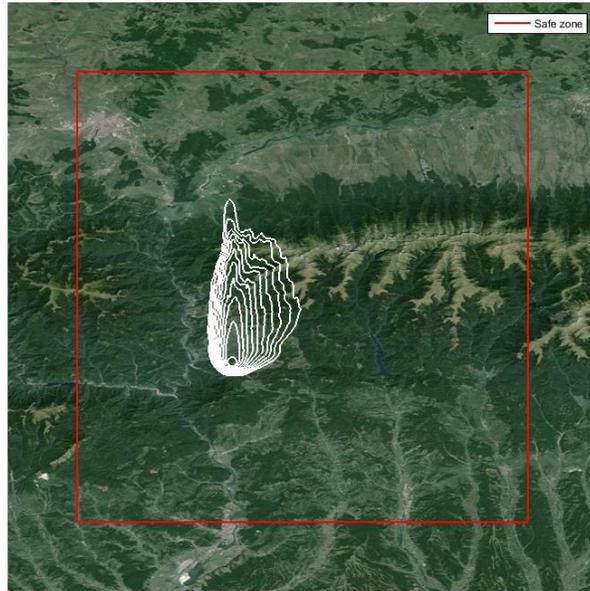


Fig. 6.19. Wildfire evolution under suppression agent for the data fusion model. Time plots are shown in increments of $\Delta t_{plot} = 40$ seconds until final time $T_{sim} = 400$ seconds

It can be clearly seen that under the specified conditions, the fire suppressing strategy has good results, severely damping the evolution of the fire front. The observation time for the controlled case is over twice as large as the uncontrolled scenario, proving the effectiveness of the method.

6.4. Conclusions

This chapter has explored a model based containment method, based on a geometrical formulation of the evolving nonlinear dynamics. The main contributions presented in this Chapter can be summarized as follows:

- Developed a novel level set formulation of a fire control problem, and proposed a law that can be robustly tuned according to a predefined contour, that is meant to enclose the evolving wildfire for the purpose of protection
- Proposed a novel data fusion algorithm that enables real-world data to be incorporated directly into the wildfire rate-of-spread models, together with fast changing measured data from atmospheric conditions
- A practical approach to prediction and control of quasi-empirical wind-aided fire spread models

The approach presented in this chapter provides a containment strategy for spatio-temporal systems, which can be used to predict and formulate an efficient tactic to tackling fire scenarios, where time and well-organized resource allocation are critical factors in delivering a positive outcome, with minimal biological degradation and economical losses. A next step in maturing the suppression method would be to incorporate constraints to the controlled variable, so that it can be used effectively in real fire combat circumstances.

7. Optimized boundary control for wildland fire fronts

7.1. Introduction

When wildfires occur, they cause severe damage to ecosystems and generate vast amounts of gaseous and particle emissions that have a significant impact on the configuration and operation of the global atmosphere and jet stream, thus actively influencing climate change (Rowell and Moore, 2000). In severe cases, suppression of wildfires can take a significant amount of time (Bayham et. al, 2011) which makes the suppression strategy, resource allocation and dispatch critical elements. These decisions affect the spatio-temporal behavior of the evolving wildfire, and dictate the severity of the damages during suppression and after fire extinction. Assessing the wildfire impacts led to the development of fire economics (Gorte and Gorte, 1979, Baumgartner and Simard, 1982, Mercer and Prestemon, 2005, Sparhawk, 1925) which is focused on estimating environmental and social relationships together with the economic compromises associated with wildfire suppression. As there will always be resource limitation in most wildfire suppression scenarios (i.e. number of firefighters, accuracy of terrain and weather information, available equipment and technology, spatial distribution of firefighting crews, financial resources etc.) it is very important to use the available resources as efficient as possible. However this is not a trivial decision, mainly due to the vast number of possibilities and priorities that might arise during a real world situation.

The problem of optimizing the resource allocation has gathered a lot of attention in recent years (Martell, 2001, Donovan and Rideout, 2003, Hu and Ntamo, 2009, Haight and Fried, 2007, Lee et. al, 2012, Castillo and Rodriguez, 2015). Depending on the main objective of interest, several approaches have been developed in literature, all having a common economical factor and various other optimization criteria. There are situations where the biodiversity of the local environment takes priority in the firefighting strategy (Richards et. al, 1999), in which the emphasis is placed on the initial attack response (Wiitala, 1999, Fiorucci et. al, 2002, Fiorucci, 2004), where the resource sharing between several firefighting units might be considered (Tsang et. al, 2013, Figueras i Jove et. al, 2013) or just reducing the cost and spread of the wildfire (Holmes et. al, 2008, Finney et. al 2009, Gebert et. al, 2007).

The optimization problems for wildfire containment and suppression mainly focus on known, available firefighting quantities and their costs of maintenance, transport and operation, together with loss costs due to the fire, but there are relatively few papers that consider the spatial distribution of firefighting resources, given the spatio-temporal structure (geometry) of the evolving fire front. Such approaches, which consider

the shape of the evolving fire front, presented in (Parlar, 1983, Bressan, 2007, Bressan and De Lellis, 2009, Wei et. al, 2015) use an optimization algorithm to allocate regions around the fire front where walls might be constructed by means of removing the vegetation/soaked in water to prevent further growth of the wildfire. A similar optimization problem is considered in (Chi et. al, 2003, HomChaudhuri et. al, 2010, HomChaudhuri et. al, 2013, Arca et. al, 2015) using genetic algorithms, where a set of quadratic fire lines are constructed around an evolving elliptical fire front, given a simplified artificially generated terrain geometry and atmospheric conditions. The geometry of the fire is usually complex and depends on a large number of factors, making its time prediction a difficult task.

This chapter is organized as follows: in Section 7.2 a new optimization method for controlling the complex fire interface, based on a geometrical model of interface evolution is presented. The new method, which takes into account the geometry of the evolving fire interface, can be tuned to mimic a large number of fire-fighting scenarios. Section 7.3 provides numerical examples with artificially generated data, as well as with data extracted from real world measurements.

7.2. Optimized boundary control of wildfires using geometrical models of interface evolution

Recalling the description in the previous chapters, consider the following nonlinear evolution model that gives the rate of propagation for a time varying curve $\bar{x}_r(s, t)$ denoting fire front over a domain of interest $\Omega \subset \mathbb{R}^2$:

$$F(\bar{x}_r(t), \bar{v}(\bar{x}_r, t), \theta_r(t)) = \begin{cases} \varepsilon_0 + a(\bar{x}_r, t) \sqrt{|\bar{v}(\bar{x}_r, t)| \cdot \cos(\theta_r(t))}^3, & |\theta_r| \leq \frac{\pi}{2} \\ \varepsilon_0(\alpha + (1 - \alpha)|\sin(\theta_r(t))|), & |\theta_r| > \frac{\pi}{2} \end{cases} \quad (7.1)$$

where \bar{v} , θ_r , ε_0 , α and a are have the same definitions as in Chapter 6. In the previous chapter an idealized control strategy was presented, based on a level set formulation of (7.1) which was aimed at containing the fire front evolution within a predefined boundary, through the means of applying a suppression agent:

$$F_c(\bar{x}_r(t), \bar{v}(\bar{x}_r, t), \theta_r(t), \bar{v}_c(\bar{x}_r(t), \varphi_r, m)) = [1 - \bar{v}_c(\bar{x}_r(t), \varphi_r, m)] \cdot F(\bar{x}_r(t), \bar{v}(\bar{x}_r, t), \theta_r(t)) \quad (7.2)$$

To achieve this goal, there were no restrictions imposed on the suppression agent $\bar{v}_c(\cdot)$. However, in most real-world scenarios, this is not the case and the resource saturation plays a major role in defining the cost and damage of a wildfire (Martell, 2001). Given the growth in the amount of information available before and during fire combat, the efficiency of the initial attack response and overall time to extinction could be minimized through varying spatial allocation of the fire suppression resource. To this end, the geometry of the fire front, together with external conditions (wind speed, terrain slope, amount of available suppression

agent etc.) can be used to define an optimization problem such that the expansion of the fire interface, $\bar{x}_F(s, t)$ is minimum, given specific constraints.

The optimized control can be viewed as a two-step process: in the first step the goal is to determine the optimal location where the fire suppression agent should be applied, based on the geometry of the evolving front and the environmental parameters (wind speed, wind direction, fuel loading, and terrain topography); the second step of the suppression strategy is achieved through applying the physical control agent at the region identified in the first step, considering that there are sufficient firefighting teams distributed around the fire interface so that there is no delay between the optimal location detection and applied control.

Let $\varphi_F(x, t)$, $\varphi_F: \Omega \times [0, T] \rightarrow \mathbb{R}$ be the level set function attached to $\bar{x}_F(s, t)$, with $T \geq 0$ representing the maximum time window. Considering that all relevant environmental data is measured or estimated, and the location and initial shape of the fire front $\bar{x}_F(s, t = 0)$ is known, an optimization control algorithm is proposed, such that the area enclosed by the fire front is minimized, given a limited amount resources, in terms of amount of firefighting materials. The algorithm can be described by the following steps:

1. Calculate the centroid $c(x_{CG}, y_{CG})$ of the evolving interface $\bar{x}_F(t)$, based on the zero level set of its attached function $\varphi_F(t)$.
2. Fit a circle $C(t)$, centered at $c(x_{CG}, y_{CG})$, with radius $R(t)$ greater than the maximum distance between the centroid and the fire interface:

$$d_{max}(t) = \max \|\bar{x}_F(t) - c(x_{CG}, y_{CG})\| \quad (7.3)$$

$$R(t) = d_{max}(t) + \Delta R, \Delta R \in \mathbb{R}^+ \quad (7.4)$$

such that the fire interface is fully enclosed within $C(t)$.

3. Divide the circle $C(t)$ into M circular sectors $C_i(t)$ of equal central angles η_i , such that

$$\sum_{i=0}^M \eta_i = 2\pi \quad (7.5)$$

and embed each circular sector in M level set functions $\varphi_c^i(x, t)$, where the level set functions $\varphi_c^i(x, t)$ are set to have binary values:

$$\varphi_c^i(x, t) = \begin{cases} 0 & \text{if } x \text{ is outside the interface } C_i(t) \\ 1 & \text{if } x \text{ is on or inside the interface } C_i(t) \end{cases}, \quad i = \overline{1, M} \quad (7.6)$$

4. Using the level set functions $\varphi_c^i(t)$, select the region where the applied (finite) control is most effective, given the goal is to minimize the total enclosed area of the fire at every time step. The applied control is set as:

$$\begin{cases} u_c(\bar{x}_\Gamma(s, t), \varphi_c^i(t), t) = -\delta_c \varphi_c^{i_{opt}}(t) F(\bar{x}_\Gamma(t), \bar{v}(\bar{x}_\Gamma, t), \theta_\Gamma(t), t) \\ i = i_{opt} \end{cases} \quad (7.7)$$

Where the optimum sector, given by $\varphi_c^{i_{opt}}(t)$, where the suppression agent is physically applied, is selected such that the fire front enclosed area $A(\bar{x}_\Gamma(s, t), u_c(\cdot))$ is minimized during the time interval $[t_0, t_1]$:

$$i_{opt} = \operatorname{argmin}_i \int_{t_0}^{t_1} A(\bar{x}_\Gamma(s, t), u_c(\bar{x}_\Gamma(s, t), \varphi_c^i(t), t)) dt, \quad i = \overline{1, M} \quad (7.8)$$

given the control constraint:

$$\begin{cases} \int_{L_j(t)}^{L_{j+1}(t)} u_c(\bar{x}_\Gamma(s, t), \varphi_c^{j+1}, t) ds = u_{max}(t) \\ u_{max}(t) = -\delta_c \varphi_c^{j+1}(t) F(\bar{x}_\Gamma(t), \bar{v}(\bar{x}_\Gamma, t), \theta_\Gamma(t), t) \frac{R(0)}{R(t)} \end{cases}, \quad j = \overline{0, M-1} \quad (7.9)$$

Where δ_c is a control weighting coefficient, and $\Delta L = L_{j+1}(t) - L_j(t)$ represents the arc length of the fire front region situated in the $(j+1)^{th}$ sector.

5. Iterate steps 1 - 4 for time $t \in [0, T]$.

Given the time evolution of the fire front is:

$$\begin{cases} \frac{\partial}{\partial t} \bar{x}_\Gamma(s, t) = F(\bar{x}_\Gamma(t), \bar{v}(\bar{x}_\Gamma, t), \theta_\Gamma(t), t) \cdot \vec{N} \\ \bar{x}_\Gamma(s, 0) = \Gamma_0 \end{cases} \quad (7.10)$$

the controlled evolution can be formulated as:

$$\begin{cases} \frac{\partial}{\partial t} \bar{x}_\Gamma(s, t) = \left(1 - u_c(\bar{x}_\Gamma(s, t), \varphi_c^{i_{opt}}(t), t)\right) \cdot F(\bar{x}_\Gamma(t), \bar{v}(\bar{x}_\Gamma, t), \theta_\Gamma(t), t) \cdot \vec{N} \\ \bar{x}_\Gamma(s, 0) = \Gamma_0 \end{cases} \quad (7.11)$$

where the agent applied on the boundary, $u_c(\bar{x}_\Gamma(s, t), \varphi_c^{i_{opt}}(t), t)$, is given by (7.3 – 7.9). Using (7.7)

the controlled evolution model (7.11) can then be written in level set formulation as:

$$\begin{cases} \frac{\partial}{\partial t} \varphi_\Gamma + \left(1 - u_c(\bar{x}_\Gamma(s, t), \varphi_c^{i_{opt}}(t), t)\right) \cdot \left(\varepsilon_0 + a(\bar{x}_\Gamma, t) \sqrt{|\bar{v}(\bar{x}_\Gamma, t)| \cdot \cos(\theta_\Gamma(t))^3}\right) \cdot |\nabla \varphi_\Gamma| = 0, & |\theta| \leq \frac{\pi}{2} \\ \frac{\partial}{\partial t} \varphi_\Gamma + \left(1 - u_c(\bar{x}_\Gamma(s, t), \varphi_c^{i_{opt}}(t), t)\right) \cdot \left(\varepsilon_0 (\alpha + (1 - \alpha) |\sin(\theta_\Gamma(t))|)\right) \cdot |\nabla \varphi_\Gamma| = 0, & |\theta| > \frac{\pi}{2} \end{cases} \quad (7.12)$$

The optimization problem (7.3 – 7.9) takes into account the geometry of the fire front together with any relevant influencing parameters that effect the fire spread. Therefore the control $u_c(\cdot)$ will always be applied in the region where the fire growth is predicted to have the greatest impact. However, there

might be situations where the fire needs to be contained within a region, in order to preserve and protect the environment and minimize economic loss. To capture this information in the control strategy, the information related to the “safe zone” contour Γ_r is needed. Therefore, if the contour Γ_r is known, and $\varphi_r(x), \varphi_r: \Omega \times [0, T] \rightarrow \mathbb{R}$ is its attached level set function, a measure of the distance between the circular sectors $C_i(t)$ and Γ_r can be obtained as:

$$d_{i,\Gamma_r}(t) = \int_{\Omega} -\varphi_r \cdot \left[\max\left(\varphi_c^i, H(1 - \varphi_r(t))\right) - H(1 - \varphi_r(t)) \right] dx dy, \quad i = \overline{1, M} \quad (7.13)$$

$$d_{\Gamma_r}(t) = [d_{1,\Gamma_r}(t), \dots, d_{M,\Gamma_r}(t)] \quad (7.13a)$$

$$\bar{d}_{\Gamma_r} = \frac{d_{\Gamma_r}(t)}{\max d_{i,\Gamma_r}(t)} \quad (7.13b)$$

where \bar{d}_{Γ_r} is the normalized distance and $H(\phi)$ is the two dimensional Heaviside function given by:

$$H(\phi) = \begin{cases} 0, & \phi < 0 \\ 1, & \phi > 0 \end{cases} \quad (7.14)$$

Having obtained the distance vector $d_{\Gamma_r}(t) \in \mathbb{R}^M$, the selection criteria in Step 4 is augmented with a decisional element through:

$$i_{opt} = \begin{cases} \operatorname{argmin}_i \int_{t_0}^{t_1} A(\bar{x}_r(s, t), u_c(\varphi_c^i(t), t)) dt, & \min d_{\Gamma_r} > \delta_{min} \\ \operatorname{argmin}_i d_{\Gamma_r}, & \min d_{\Gamma_r} \leq \delta_{min} \end{cases}, \quad i = \overline{1, M}, \quad \delta_{min} \in \mathbb{R}^+ \quad (7.15)$$

where δ_{min} is the threshold distance below which the control is switched to the sector closest to the “safe zone” Γ_r .

7.3. Numerical implementation of the optimized control problem

This section presents examples of applied control to the fire interface, using the optimized containment method presented in the previous section, with both artificial data and real data extracted from satellite imaging and DTED models.

7.3.1. Numerical discretization of the control algorithm

In the following it is assumed that the domain of interest where the fire growth is observed is given by a bounded, rectangular domain $\Omega \subset \mathbb{R}^2, \Omega: [-l, l] \times [-w, w]$ and is discretized on a uniform

rectangular grid, as in Section 3.4.1. Given the set of P vertices (x_i, y_i) of the evolving curve $\bar{x}_r(s, t)$, the enclosed area and the centroid coordinates are given by (Bockman, 1989):

$$A = \frac{1}{2} \sum_{i=0}^{P-1} (x_i y_{i+1} - x_{i+1} y_i) \quad (7.16)$$

$$x_{CG} = \frac{1}{6A} \sum_{i=0}^{P-1} (x_i + x_{i+1}) \cdot (x_i y_{i+1} - x_{i+1} y_i) \quad (7.17)$$

$$y_{CG} = \frac{1}{6A} \sum_{i=0}^{P-1} (y_i + y_{i+1}) \cdot (x_i y_{i+1} - x_{i+1} y_i) \quad (7.18)$$

Based on the center point given by (x_{CG}, y_{CG}) the circular sectors and their associated level set functions from Step 3 of the control method are generated using:

$$C_i(t) : \begin{cases} x_{CG} + R(t) \cos(\zeta_i) \\ y_{CG} + R(t) \sin(\zeta_i) \end{cases}, \zeta_i \in [\eta_i, \eta_{i+1}] \quad (7.19)$$

$$\varphi_c^i(x, t) = \begin{cases} 1, & \|x - (x_{CG}, y_{CG})\| \leq R(t) \\ 0, & \|x - (x_{CG}, y_{CG})\| > R(t) \end{cases} \quad (7.20)$$

which can now be directly used to calculate equations (7.7)-(7.9). To properly evaluate the integral in (7.13) a suitable approximation of the Heaviside function is needed. Here a smeared out, first order approximation is used (Osher and Fedkiw, 2003):

$$H(\phi) = \begin{cases} 0 & , \phi < -\epsilon \\ \frac{1}{2} + \frac{\phi}{2\epsilon} + \frac{1}{2\pi} \sin\left(\frac{\pi\phi}{\epsilon}\right) & , -\epsilon \leq \phi \leq \epsilon \\ 1 & , \epsilon < \phi \end{cases} \quad (7.21)$$

where ϵ is a tunable parameter that determines the bandwidth of numerical smearing, chosen to be $1.5\Delta x$.

7.3.2. Synthetic data simulations

To illustrate the proposed optimized fire suppression approach, several cases are presented, representing controlled and uncontrolled scenarios, using artificial fire fronts. The evolving interface is described by a simple, non-intersecting closed curve, i.e. no ‘islands’ are formed throughout the burning process. The initial fire front, observed at detection, is chosen to be a circle of radius $R = 50$, centered within the observation domain Ω ; the safety zone (Γ_r) is given by the square region of length $L = 350$, also centered within Ω (Fig 7.1). The two dimensional space in which the growth is observed is set as the square $\Omega: [0, 421] \times [0, 421]$, which is discretized on a uniform Cartesian grid with resolution of 421×421 points. The fire growth model parameters are set to be $\bar{v} = 50$, $\epsilon_0 = 0.2$, $\alpha = 0.5$ and $a =$

0.5. The angle at which the wind is blowing is initially set to be 90° (i.e. S-N direction) and is decreased at every $\Delta t_{\theta_r} = 2$ seconds by 5° (i.e. the wind direction is shifting towards the east).

The number of divisions used to evaluate the effectiveness of the control is chosen to be $M = 4$, with $\Delta R = 10$ resulting in four sectors (Fig. 7.2, Fig. 7.3) with initial radius at $t = 0$ to be $R(0) = 50$ with the control weighting term δ_c is set to 0.5. The initial normalized distance function $\bar{d}_{r_r}(0)$, shown in Fig. 7.4, displays the evaluation of the “closeness” of each sector to the predefined safety zone. Although the initial front is centered within the domain, due to numerical discretization, the initial distance values are marginally different. The threshold for (7.15) is set to be $\delta_{min} = 10^6$. The model is then simulated for the time period $T = [0,20]$ seconds, with interface snapshots taken at constant time intervals $\Delta t_{plot} = 2$ seconds. The time evolution of the fire interface in the controlled and uncontrolled scenarios is depicted in Fig.7.5a and Fig.7.5b. The normalized distance function at final simulation time for the two cases is illustrated in Fig. 7.6a and Fig. 7.6b.

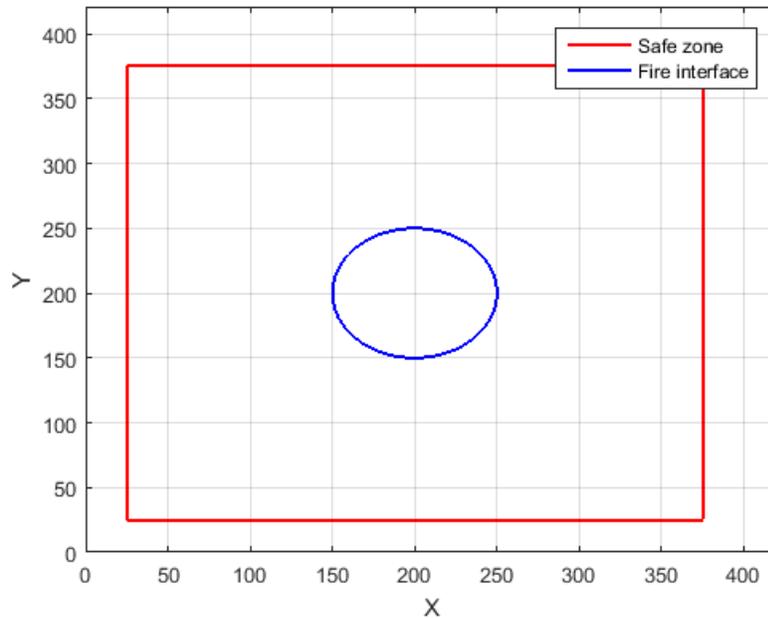


Fig. 7.1. Initial setup for wildfire evolution

With the described evolution setup, the simulated scenario would be representative of a rapidly growing fire, with relatively limited firefighting resources, where the fire attack is spread over a large area. Although the scenario does not foresee an extinction time, the fire growth is minimized within the sector where the destruction would be the greatest with respect to the selection criteria (7.15) previously defined, with the sector nearest to the safety zone being selected as Sector 1.

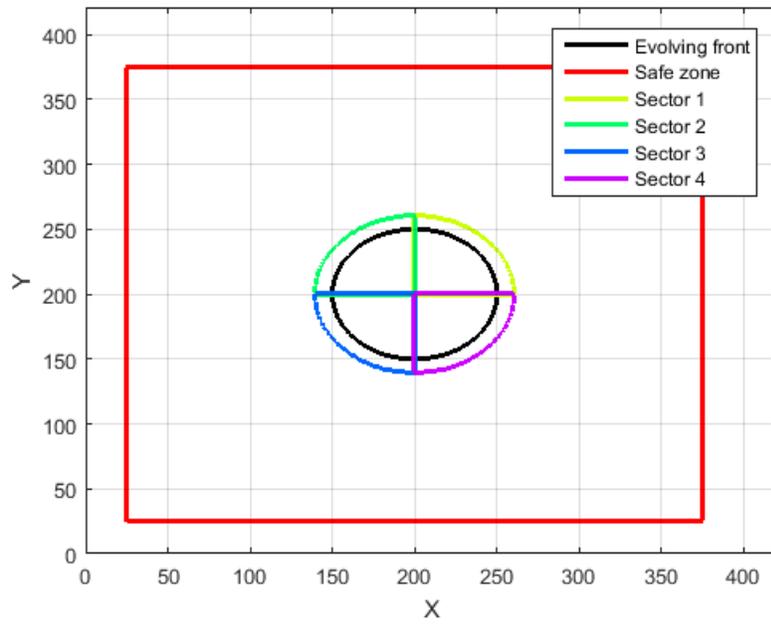


Fig. 7.2. Sector division at time $t = 0$

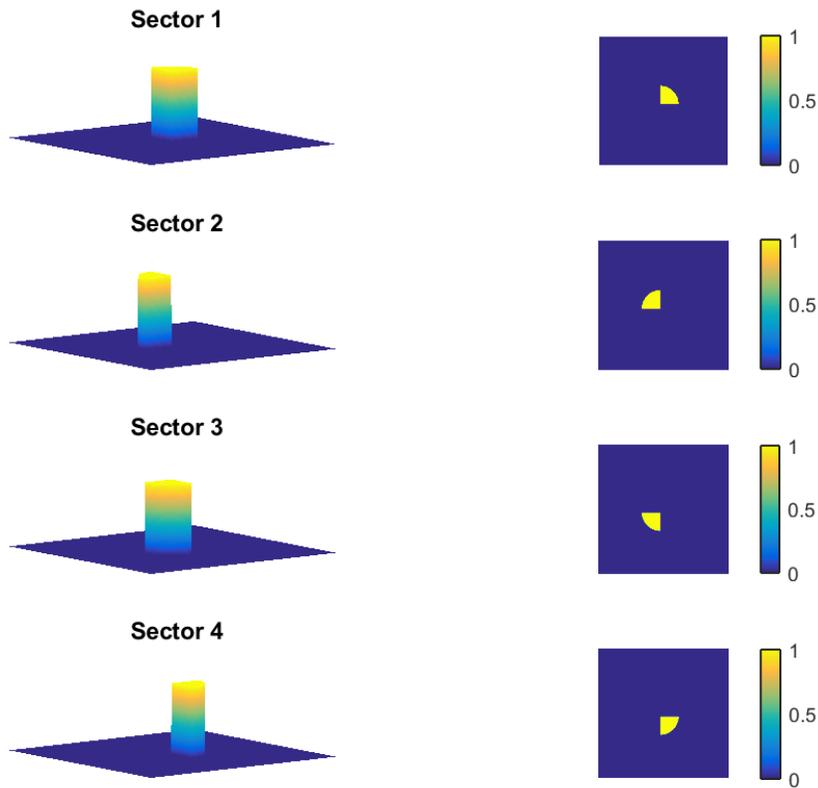


Fig. 7.3. Level set functions $\varphi_C^1, \varphi_C^2, \varphi_C^3, \varphi_C^4$ attached to the four sectors at time $t = 0$

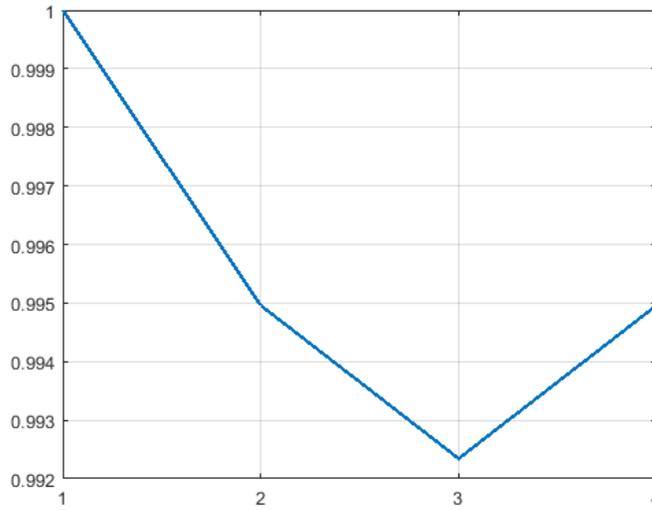


Fig. 7.4. Normalized distance function $\bar{d}_r(0)$

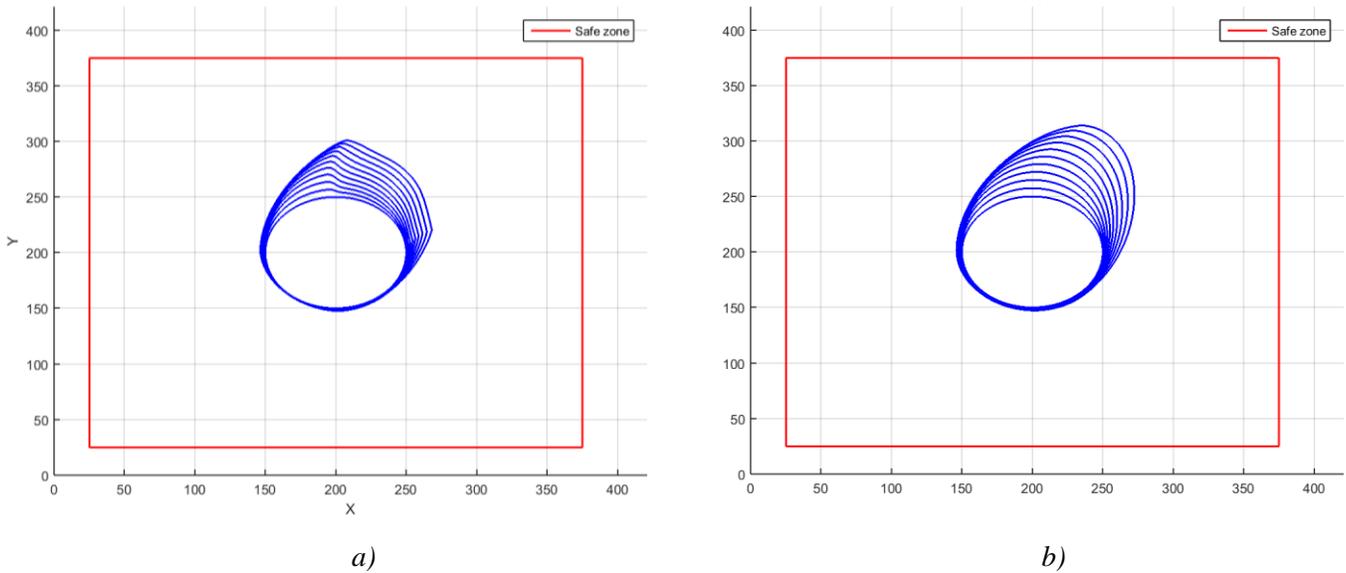


Fig. 7.5. a) Controlled fire front evolution b) Uncontrolled fire front evolution

The control selection strategy defined by (7.8, 7.9 and 7.15) can be tuned through three independent terms: δ_c , δ_{min} and M . The control weight δ_c together with the number of divisions M and the minimum safe distance δ_{min} dictate the power of the initial attack and the effectiveness of the suppression method.

As δ_c is augmented, a larger amount of firefighting resource is available to tackle the evolving fire front. The effect of increasing the control weighting can be viewed in Fig. 7.7, where the front evolution is compared for varying values of δ_c , using a low number of divisions, with $M = 4$. All remaining parameters remain unchanged with regards to the previous example. As δ_c is larger, the area growth of

the fire front is seen to decrease comparatively with time, indicating a shorter active combat period and a faster extinction time.

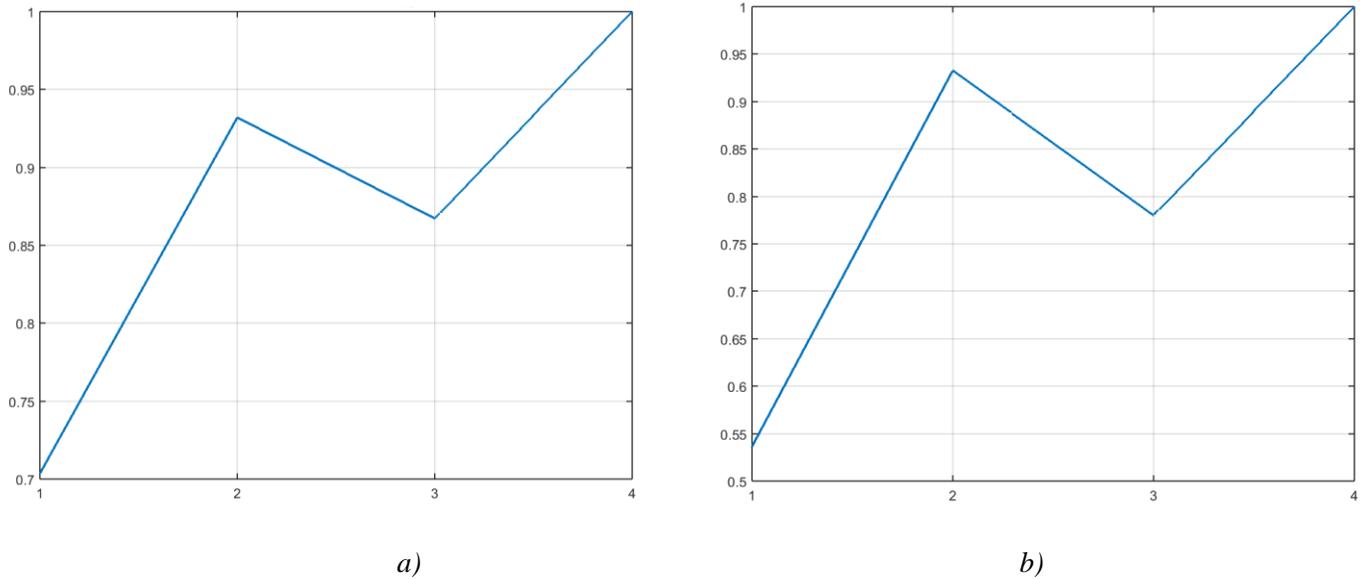


Fig. 7.6. a) Normalized distance function \bar{d}_{r_r} at time $t = 20$ - Controlled case b) Normalized distance function \bar{d}_{r_r} at time $t = 20$ - Uncontrolled case

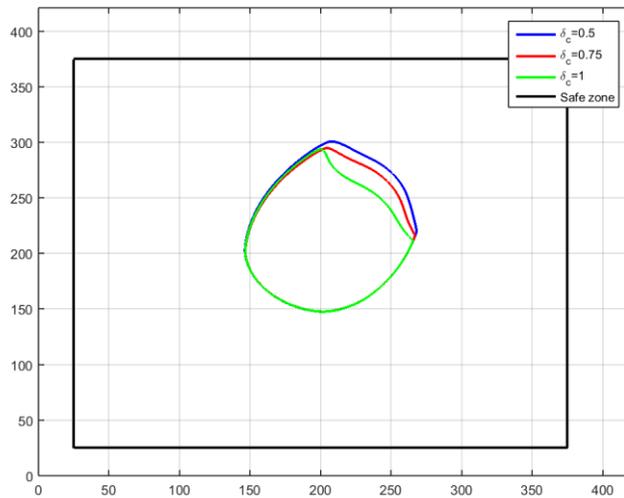


Fig. 7.7. Suppressed fire evolution under varying δ_c

The increase of the number of divisions M enables a more accurate evaluation of the predicted fire front and allows the control to be applied with a finer detail, enhancing the overall suppression efficiency for a selected region, given a limited (low) amount of resources. This implies that for the purpose of suppressing the fire growth, a series of M firefighting squads are allocated. These squads are then spatially distributed according to the sector division, around the evolving interface, such that if an action

is needed, it is supplied at demand, with the intensity given by δ_c , for the selected region. However, if a vast amount of resources are available or if the evolving fire front is small in size, choosing a smaller number of divisions will increase the prediction accuracy and the efficiency of the suppression method. This effect is illustrated in Fig. 7.8, where three scenarios are presented, with two, four and eight divisions respectively. The initial setup for the problem is chosen to be the same as in Fig. 7.1, with $R(0) = 50$, $\Delta R = 10$, $\delta_c = 0.5$, $\delta_{min} = 2 \cdot 10^5$ and all remaining parameters remain unchanged (fire evolution model parameters, fuel loading, wind speed, wind direction change rule).

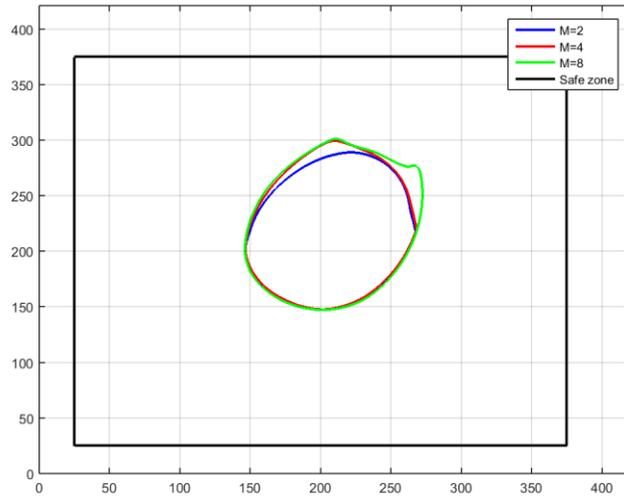


Fig. 7.8. Suppressed fire evolution under varying division numbers M

The minimum safe distance term δ_{min} indicates the threshold when the fire front has extended into a critical region and all effort must be focused on that area. This is representative of a scenario where the fire is close to an area which contains endangered species for example or the fire is near the urban-wildlife interface and immediate action is necessary in order to preserve and protect the delicate ecosystem and to minimize human and financial losses. The effect of varying δ_{min} is illustrated in Fig. 7.9, where the suppressed fire front evolution is compared between three distinct values. The number of divisions used to evaluate the effectiveness of the control is chosen to be $M = 4$, with $\Delta R = 10$ together with initial radius at $t = 0$ to be $R(0) = 50$ and the control weighting term $\delta_c = 0.5$. The model is then simulated for the time period $T = [0,10]$ seconds, with interface snapshots taken at constant time intervals $\Delta t_{plot} = 2$ seconds. The angle at which the wind is blowing is initially set to be 90° (i.e. S-N direction) and is decreased at every $\Delta t_{\theta_r} = 2$ seconds by 5° (i.e. the wind direction is shifting towards the east). At $t = 5$ seconds the wind direction is set to change from its current direction to -90° (i.e. N-S direction), indicating a counterflow wind change, after which the direction is increased at every $\Delta t_{\theta_r} = 2$ seconds by 5° (i.e. the wind direction is shifting towards the west). The wind speed is set to be $\bar{v} = 100$.

With increasing values of δ_{min} the fire front regions that are closer to the safe zone take priority quicker, regardless of the fire evolution in other directions.

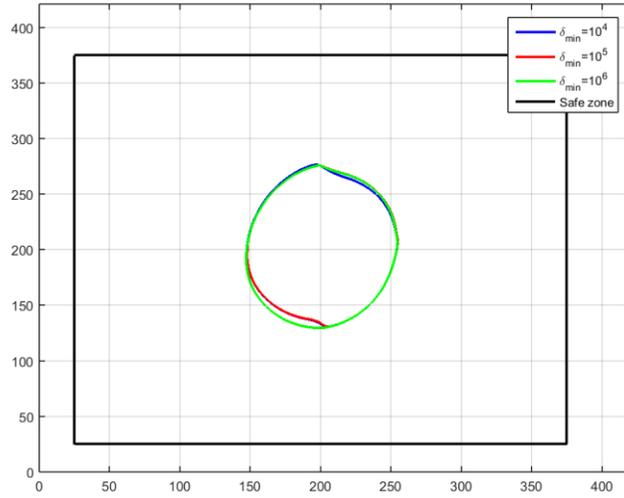


Fig. 7.9 Suppressed fire evolution under varying δ_{min}

7.3.3. Satellite and DTED data simulations

As with most firefighting incidents, possessing as much information as possible enables better resource planning and more precise predictions of future fire growth. Therefore, real world information can be used to enhance the potential of the optimized suppression method. The data fusion algorithm presented in Section 6.3.3 could be used to incorporate real world information directly into the proposed control algorithm. Taking the observed region to be the same as in Fig. 6.16, together with estimated fuel loading (Fig. 6.17) and estimated terrain slope (Fig. 6.15), the optimized control problem can be defined as in the previous sections. The number of divisions used is set to $M = 3$, with $R(0) = 50, \Delta R = 10, \delta_c = 0.5$ and $\delta_{min} = 10^4$. The safe region is defined by a square of side length $l = 500$ centered on the observed domain. The flank velocity is influenced by $\varepsilon_0 = 0.2$ and the rear/flank velocity ratio $\alpha = 0.5$. The model is then simulated for the time period $T = [0, 50]$ seconds, with interface snapshots taken at constant time intervals $\Delta t_{plot} = 5$ seconds. The angle at which the wind is blowing is initially set to be 90° (i.e. S-N direction) and is increased at every $\Delta t_{\theta_r} = 5$ seconds by 5° (i.e. the wind direction is shifting towards the west). The wind speed is set to blow in the S-N direction with a magnitude of $\bar{v} = 40$, and is gradually increased, at every $\Delta t_{\bar{v}} = 5$ seconds, reaching a value of $\bar{v} = 60$ at the final simulation time. The comparison between the controlled and unconstrained evolution using the parameters given above is shown in Fig. 7.10 a, b.

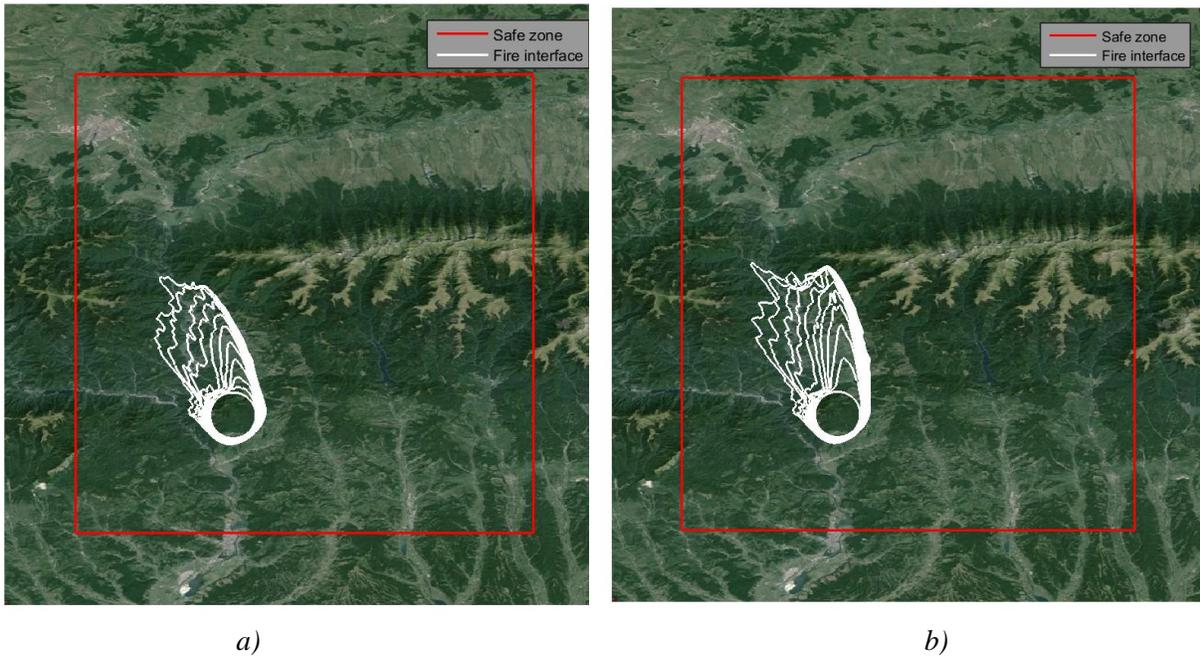


Fig. 7.10. a) DTED model controlled evolution b) DTED model unconstrained evolution

Even in cases where the atmospheric conditions are varying, such as changing wind speed and erratic wind direction, the suppression method is shown to effectively slow down the evolution of the fire front, given a limited amount of resource which is distributed in three areas around the fire front.

7.4. Conclusions

This chapter applied an optimized model based containment method, based on a nonlinear geometrical interface evolution model for wind-aided fire spread. This has resulted in several contributions, namely

- An optimized fire suppression method that incorporates the spatial distribution and number of available resources together with the geometry of an arbitrarily complex ‘safe zone’ within the control strategy to determine the most effective action to be taken with respect to a cost functional.
- A dynamically tunable three term cost functional that allows the prediction and evaluation of efforts needed to achieve a predefined target goal, based on firefighting resource availability, wildfire dimensions and priority zones.
- A practical approach to formulating an optimized firefighting strategy based on the geometry of the evolving fire front together with all influencing parameters.

8. Conclusions

8.1. Introduction

An extremely large number of natural systems can be categorized within the class of spatio-temporal systems, with a diverse range varying from microscopic (stem cell colonies, crystal growth etc.) to large scale dynamics (wildfire evolution, weather patterns etc.). The aim of this research has been to explore, formulate and present an approach to solving the non-trivial task of controlling the nonlinear dynamics in systems which exhibit spatio-temporal patterns, namely the dynamics observed in dendritic crystal growth and wind-aided wildfire spread. Three separate approaches have been developed, using a continuous representation in the form of PDE systems. In essence, the developed control methods consist of:

- Placing the control problem in a finite observable time – space domain,
- Incorporating real world data in the parametric models from either analyzing an experimentally observed pattern formation process or estimating key parameter values from a set of static (low time dynamics) observations and
- Employing higher dimensional transformations to the two or three dimensional processes under analysis to obtain a local set of properties used to synthesize the control laws.

The novel approaches have been proven to deliver excellent results with both synthetically generated and physical data sets, thus fully achieving the aims set out in Section 1.4 of this thesis.

This chapter is structured as follows. The second part, Section 8.2 provides an overview of the thesis and describes the main contributions. The final fragment, Section 8.3, looks at possible future directions of research within the area of modeling and control of spatio-temporal systems, based on the developments made within this thesis.

8.2. Thesis summary

The thesis begins with a short introduction to spatio-temporal models, with emphasis on dendritic crystal growth and wind aided wildfire evolution. Some theoretical aspects of the processes are described, which highlight the complexity of the systems under analysis. The topic addressed in this thesis, namely control of highly complex dynamical systems, represents a new direction in research, as systems with the fore mentioned properties have not been addressed in numerous studies to date.

Chapter 2 discusses several approaches to model spatio-temporal dynamics, together with their mathematical formulation and provides a clear reasoning as to why PDE models are used throughout this work to achieve the research objectives. Also, a short introduction to developing dynamics is given, in the form of geometric driven interface evolution, which is a key property for the study of pattern formation.

The work undertaken in Chapters 3 and 4 provides a detailed and comprehensive overview of the theoretical aspects of modeling and numerical simulations related to the two systems under analysis. These chapters provide the basis for the later developed formulation of controlled dynamics.

In Chapter 3, both local and nonlocal models of interface evolution are analyzed, with focus on dendritic crystal growth. In view of purely local evolution models, the crystal interface growth is formulated as a nonlinear function that describes the rate of change in terms of curvature flow, namely higher order polynomials of curvature and its derivatives with respect to arc length. Nonlocal models were studied through the standard model of solidification, which was adapted to include the interface kinematics found in dendritic growth through a Gibbs-Thomson relation – the modified two-phase Stefan problem. Although these two approaches are able to describe the crystal growth process, numerical discretization and simulation prove to pose several challenges. Because these models do not have a general analytical solution, interface capturing techniques are frequently used, which increase the complexity even further. The higher dimensional transformation used in the level set method enables study of the effects of evolving curves under normal speed dependent on local curvature or an externally generated velocity field.

Chapter 4 provides a thorough analysis of quasi-empirical wind aided fire spread models, through the scope of geometrical representations. An immense variety of scenarios can be generated through altering the spatial and temporal distribution of key influencing parameters in the fire growth, which makes prediction process non trivial. A level set formulation is used to embed the nonlinear parametric evolution model into a time evolving surface, by use of a signed distance transformation, which can then be used to keep track of the fire front. The numerical study provided shows the influence of parameter variations on the fire interface.

The frameworks described in Chapter 5, 6 and 7 represent the key innovations of this thesis. They describe a methodology of taking a complex model, with spatio-temporal dynamics, and creating a control loop control around the process of interest such that several performance criteria are met.

A novel closed loop distributed control method for dendritic crystal growth is shown in Chapter 5. The method is built around the geometrical models presented in Chapter 3, and uses the temperature within a bounded domain as the control variable. Initially the closed loop method is derived in a two dimensional space, and then further extended to include dynamics in a three dimensional domain. When looking at the performance of the control strategy, quantitative measures of representative interface properties are used.

As part of the validation process, several scenarios were studied, with excellent results. Following a comparison of different control approaches, the conclusion that emerged was that in the developed framework, basic control laws can be effectively used to control the nonlinear model. It was shown that experimentally observed crystal growth data can be directly incorporated into the new control method, enabling the design of arbitrarily complex interface shapes. Nonetheless, the study also pointed out some of the limitations of the method, that is, the complexity of the desired crystal shape is a key factor in determining an appropriate discretization for the control strategy.

In Chapter 6 a new framework for containment and suppression for the evolution of wind assisted wildfire spread was presented. More precisely, the temperature of the fire is gradually reduced, until full extinction is achieved. In this regard, the study followed a similar principle as the one in Chapter 5, by using a similar geometrical representation of the evolving dynamics in a level set formulation. However the introduction of new concepts was needed to define the containment strategy – a predefined containment area, which encloses the initial fire front. Since the method is designed to aid in real time situations, a prediction of the wildfire behavior together with the information regarding the intensity of the control is derived. To aid this, a parameter estimation was carried out over a finite area, with respect to terrain topography and biofuel quantities, information which was then incorporated in straightforward manner in the control strategy. Although significant positive results have been achieved, proving the effectiveness of the proposed algorithm, it is yet to be considered a complete suppression method, mainly due to the complexity of the fire spread mechanism and the environmental factors.

Chapter 7 explored a new optimization strategy for controlling wildland fire fronts based on a geometrical model of interface evolution. To achieve this, a cost function was defined, that provides the optimal location where the fire resource needs to be allocated, based on the complex fire geometry, the terrain topography and the atmospheric conditions that influence the fire. Several tuning parameters have been incorporated within the cost function, among which the number of spatial divisions, the control weighing and the minimum safe distance to a safe zone are most important. The careful selection of these parameters allows for a large variety of scenarios to be simulated and analyzed. The method is shown to be ready applicable to real-world data, with DTED data and fuel loading estimation being easily incorporated in the control algorithm. A more comprehensive optimization might be achieved through real-time data fusion between measured atmospheric data and the actual fire evolution, such that a more accurate prediction of the fire evolution is achieved, and therefore a more effective firefighting strategy is deployed.

As a final point it is worth noting that this study was successful on bringing together various fields of research, such as computational physics theory, phase transitions theory, flame propagation theory and spatio-temporal dynamics theory, which together aided the development of new control strategies that will definitely have significant practical applications in real world scenarios.

8.3. Future research directions

The thesis mainly focused on presenting original model based control methods together with a comprehensive numerical analysis for two distinct systems with spatio-temporal dynamics. It is known that the dynamics of most spatio-temporal systems have yet to be fully understood and there is currently great interest in determining the basic principles which govern pattern formation in such systems. Therefore, given that the control of such systems is still in its early stages, several opportunities of future research may be considered, some of which are briefly presented below:

- a) Advance the distributed control methodology to achieve interior and boundary control
 - These control approaches would be of great interest, since the aim of the project is to develop an efficient and physically attainable control methodology for dendritic crystal growth models.
 - Another issue that needs addressing is the problem of controllability. The system dynamics is described by a parabolic PDE. We are allowed to act on the trajectories of the system by means of a suitable control (internal, boundary) and, given a time interval, the initial and final states of the system, we need to find a control such that the solution matches the desired state at the final time. Controllability of such approaches is uncertain given the control problem (shape control of the two dimensional heat equation)
- b) Extend existing geometrical models of boundary evolution to include long-range interactions
 - Current geometrical models have been extended to include variations in temperature, but model can be further extended to include changes in solute concentration, in order to capture more efficiently the dynamics of the crystal growth.
 - The modified two-phase Stefan problem would be an excellent candidate for this approach, since additional variations in non-local parameters can be incorporated in a straightforward manner in the growth dynamics.
- c) Modelling and analysis and control of other types of crystal growth processes
 - Another interesting pattern formation exhibited in crystal growth processes comes in the form of spiral crystal growth. As with dendritic crystal growth, modeling the boundary evolution of spiral crystal growth systems poses several difficulties. Using experimental data, an identification of the growth rule (system model) in spiral growth can be determined, which can then be used to design a control method for this type of interface evolution.
- d) Understanding the effects due to variations in fuel vertical structure and moisture on fire front dynamics

- Although ignition events are rarely discovered as soon as they occur, the rate at which an initial fire front evolves in time will be influenced by the fuel structure, which is not necessarily heterogeneously distributed in the vertical plane. This is also reflected in the amount of moisture that the respective layer is capable of holding at a specific time. Therefore the fuel loading estimates obtained from satellite imaging suffer from the limitation that this information is not directly accessible, which is detrimental to the accurate prediction of wildfires. Since the parameters are linked closely to climate conditions, a parametric model which reflects the weather history could significantly improve the effectiveness of the suppression method.
- e) Advanced control and containment strategies for protection against wildfires
- The nonlinear and complex dynamics of wildfires poses a significant challenge in the design of a reliable parametric control method. Functional optimization methods can be developed around the parametric evolution models with the aim of improving the spatial effectiveness of the suppression agent. To this end, model based predictive control could be used in conjunction with the time evolution of several key influencing parameters to determine time and energy efficient suppression approaches, together with increased spatial awareness of where the fire attack should be focused on.

Bibliography

- Adalsteinsson, D. and Sethian, J.A. (1995), 'A Fast Level Set Method for Propagating Interfaces', *Journal of Computational Physics*, 118 (2), 269-77. doi: 10.1006/jcph.1995.1098
- Adalsteinsson, D. and Sethian, J. A. (1997), 'The Fast Construction of Extension Velocities in Level Set Methods', *Journal of Computational Physics*, 148 (1), 2-22. doi: 10.1006/jcph.1998.6090
- Albini, Frank A (1976), *Estimating wildfire behavior and effects* (Intermountain Forest and Range Experiment Station, Forest Service, US Department of Agriculture).
- Albini, Frank A (1985), 'A Model for Fire Spread in Wildland Fuels by-Radiation†', *Combustion Science and Technology*, 42 (5-6), 229-58.
- Alexander, Martin E, Stocks, Brian James, and Lawson, Bruce D (1990), *Fire behavior in black spruce-lichen woodland: the Porter Lake project* (Forestry Canada-Northwest Region).
- Almeida, Rodolfo Maduro and Macau, Elbert EN (2011), 'Stochastic cellular automata model for wildland fire spread dynamics', *Journal of Physics: Conference Series* (285: IOP Publishing), 012038.
- Almgren, Fred and Wang, Lihe (2000), 'Mathematical existence of crystal growth with Gibbs-Thomson curvature effects', *The Journal of Geometric Analysis*, 10 (1), 1-100.
- Anderson, Hal E (1968), *Sundance Fire: an analysis of fire phenomena* (Intermountain Forest and Range Experiment Station, Forest Service, US Department of Agriculture).
- Anderson, Hal E and FOREST, INTERMOUNTAIN (1969), *Heat transfer and fire spread* (Intermountain Forest and Range Experiment Station, Forest Service, US Department of Agriculture).
- Anderson, DH, et al. (1982), 'Modelling the spread of grass fires', *The Journal of the Australian Mathematical Society. Series B. Applied Mathematics*, 23 (04), 451-66.
- Arca, Bachisio, Ghisu, Tiziano, and Trunfio, Giuseppe A (2015), 'GPU-accelerated multi-objective optimization of fuel treatments for mitigating wildfire hazard', *Journal of Computational Science*.
- Arfken, George B (2013), *Mathematical methods for physicists* (Academic press).
- Balbi, Jacques Henri, et al. (2009), 'A physical model for wildland fires', *Combustion and Flame*, 156 (12), 2217-30.
- Banks, D., et al. (2008), 'Optimal Control of a Moving Boundary by Laser Heating in a 2-Phase Stefan Problem', *World Congress of Nonlinear Analysts* (Florida, USA).
- Baumgartner, David C and Simard, Albert J (1982), 'Wildland fire management economics: a state of the art review and bibliography'.
- Bayham, Jude and Yoder, Jonathan (2011), 'Survival analysis of wildfire duration, cost, and size'.
- Ben-Jacob, E, et al. (1984), 'Boundary-layer model of pattern formation in solidification', *Physical Review A*, 29 (1), 330.
- Bernauer, M. K. and Herzog, R. (2011), 'Optimal Control of the Classical Two-Phase Stefan Problem in Level Set Formulation', *SIAM Journal on Scientific Computing*, 33 (1).

- Bevis, Michael, et al. (1992), 'GPS meteorology- Remote sensing of atmospheric water vapor using the Global Positioning System', *Journal of Geophysical Research*, 97 (D14), 15787-801.
- Billings, Stephen A (2013), *Nonlinear system identification: NARMAX methods in the time, frequency, and spatio-temporal domains* (John Wiley & Sons).
- Billings, SA and Coca, Daniel (2002), 'Identification of coupled map lattice models of deterministic distributed parameter systems', *International Journal of Systems Science*, 33 (8), 623-34.
- Billings, SA and Yang, Yingxu (2003), 'Identification of the neighborhood and CA rules from spatio-temporal CA patterns', *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, 33 (2), 332-39.
- Bockman, SF (1989), 'Generalizing the formula for areas of polygons to moments', *American Mathematical Monthly*, 96 (2), 131-32.
- Brakke, Kenneth A (2015), *The Motion of a Surface by Its Mean Curvature.(MN-20)* (Princeton University Press).
- Bressan, Alberto (2007), 'Differential inclusions and the control of forest fires', *Journal of Differential Equations*, 243 (2), 179-207.
- Bressan, Alberto and De Lellis, Camillo (2009), 'Existence of optimal strategies for a fire confinement problem', *Communications on Pure and Applied Mathematics*, 62 (6), 789-830.
- Brower, R. , et al. (1983), 'Geometrical Approach to Moving-Interface Dynamics', *Physical Review Letters*, 51 (13), 1111-14. doi: 10.1103/PhysRevLett.51.1111
- Brower, R. , et al. (1984), 'Geometrical models of interface evolution', *Physical Review A*, 29 (3), 1335-42. doi: 10.1103/PhysRevA.29.1335
- Bruce, HD, Pong, WY, and Fons, WL (1961), *The Effect of Density and Thermal Diffusivity of Wood on the Rate of Burning Wood Cribs* (US Department of Agriculture, Forest Service, Pacific Southwest Forest and Range Experiment Station).
- Bunimovich, Leonid A (1995), 'Coupled map lattices: one step forward and two steps back', *Physica D: Nonlinear Phenomena*, 86 (1), 248-55.
- Burrows, Neil, Ward, Bruce, and Robinson, Alex (1991), 'Fire behaviour in spinifex fuels on the Gibson Desert nature reserve, Western Australia', *Journal of Arid Environments*, 20 (2), 189-204.
- Butler, BW, et al. (2004), 'A radiation-driven model for crown fire spread', *Canadian Journal of Forest Research*, 34 (8), 1588-99.
- Caginalp, Gunduz and Fife, Paul (1986), 'Phase-field methods for interfacial boundaries', *Physical Review B*, 33 (11), 7792.
- Canada, Environment 'Lightning and Forest Fires',
<http://ec.gc.ca/foudre-lightning/default.asp?lang=En&n=48337EAE-1>.
- Center, National Interagency Fire 'Wildland Fire Statistics, Historically Significant Wildland Fires and Total Wildland Fires and Acres (1960-2009)',
http://www.nifc.gov/fireInfo/fireInfo_stats_histSigFires.html
<http://wildland-fires.findthedata.org/>

http://www.nifc.gov/fireInfo/fireInfo_stats_totalFires.html.

- Chang, Y. C. , et al. (1996), 'A level set formulation of Eulerian interface capturing methods for incompressible fluid flows', *Journal of Computational Physics*, 124 (2), 449-64.
- Chen, Long-Qing (2002), 'Phase-field models for microstructure evolution', *Annual review of materials research*, 32 (1), 113-40.
- Chen, XX and He, WY (2008), 'On the Calabi flow', *American journal of mathematics*, 130 (2), 539-70.
- Chen, H. , Min, C. , and Gibou, F. (2009), 'A numerical scheme for the Stefan problem on adaptive Cartesian grids with supralinear convergence rate', *Journal of Computational Physics*, 228 (16), 5803-18. doi: 10.1016/j.jcp.2009.04.044
- Chen , S. , et al. (1997), 'A Simple Level Set Method for Solving Stefan Problems', *Journal of Computational Physics*, 135 (1), 8-29. doi: 10.1006/jcph.1997.5721
- Chi, Sung-Do, et al. (2003), 'A simulation-based decision support system for forest fire fighting', *AI* IA 2003: Advances in Artificial Intelligence* (Springer), 487-98.
- Chopp, David L. (1993), 'Computing Minimal Surfaces via Level Set Curvature Flow', *Journal of Computational Physics*, 106 (1), 77-91. doi: 10.1006/jcph.1993.1092
- Chopp, David L. (2009), 'Another Look at Velocity Extensions in the Level Set Method ', *SIAM Journal on Scientific Computing*, 31 (5), 3255-73.
- Chopp , David L. and Sethian , J. A. (1999), 'Motion by Intrinsic Laplacian of Curvature', *Interfaces and Free Boundaries*, 1 (1), 107-23. doi: 10.4171/IFB/6
- Coca, D and Billings, SA (2001), 'Identification of coupled map lattice models of complex spatio-temporal patterns', *Physics Letters A*, 287 (1), 65-73.
- Coen, Janice L, et al. (2013), 'WRF-Fire: Coupled Weather-Wildland Fire Modeling with the Weather Research and Forecasting Model', *Journal of Applied Meteorology & Climatology*, 52 (1).
- Crank, John (1984), 'Free and moving boundary problems', *AMC*, 10, 12.
- D'Ambrosio, Donato, et al. (2006), 'A model for the simulation of forest fire dynamics using cellular automata', *Proceedings of the iEMSs Third Biennial Meeting: Summit on Environmental Modelling and Software*.
- Dantzig, J. A. and Rappaz, M. (2009), *Solidification* (EPFL Press).
- DMA (1996), 'Performance Specification Digital Terrain Elevation Data (DTED)', *Defence Mapping Agency*, (MIL-PRF-89020A Superseding MIL-D-89020).
- Do Carmo, Manfredo Perdigao (1976), *Differential geometry of curves and surfaces* (2: Prentice-hall Englewood Cliffs).
- Donovan, Geoffrey H and Rideout, Douglas B (2003), 'An Integer Programming Model to Optimize Resource Allocation for Wildfire Containment', *Forest Science*, 49, 2.
- Eden, Murray (1961), 'A two-dimensional growth process', *Dynamics of fractal surfaces*, 4, 223-39.
- Eisenhart, Luther Pfahler (1947), *An introduction to differential geometry: with use of the tensor calculus* (3: Princeton University Press Princeton, New Jersey).

- Emmons, Howard W and Shen, Tom (1971), 'Fire spread in paper arrays', *Symposium (International) on Combustion* (13: Elsevier), 917-26.
- Evans, Go W (1951), 'A note on the existence of a solution to a problem of Stefan', *Appl. Math*, 9 (185-193), 162.
- Evans, Lawrence C (1982), 'A new proof of local C^1, α regularity for solutions of certain degenerate elliptic PDE', *Journal of Differential Equations*, 45 (3), 356-73.
- Evans, LC (1998), *Partial Differential Equations. Graduate studies in Mathematics, 19* (American Mathematical Society).
- Fabbri, M and Voller, VR (1997), 'The phase-field method in the sharp-interface limit: A comparison between model potentials', *Journal of Computational Physics*, 130 (2), 256-65.
- Fendell, FE and Wolff, MF (2001), 'Wind-aided fire spread', *Forest Fires: Behaviour and Ecological Effects*, 171-224.
- Fernandes, Paulo A Martins (2001), 'Fire spread prediction in shrub fuels in Portugal', *Forest ecology and management*, 144 (1), 67-74.
- Fernandes, P, Botelho, H, and Loureiro, C (2002), 'Models for the sustained ignition and behaviour of low-to-moderately intense fires in maritime pine stands', *Forest Fire Research & Wildland Fire Safety*, Viegas (ed.), 1-11.
- Figueras i Jove, Jaume, et al. (2013), 'Simulation and optimization for an experimental environment to wildfire resource management and planning: firefight project modelling and architecture', *Simulation Conference (WSC), 2013 Winter* (IEEE), 1950-60.
- Filippi, Jean-Baptiste, et al. (2010), 'Discrete event front-tracking simulation of a physical fire-spread model', *Simulation*, 86 (10), 629-46.
- Finney, Mark A and Andrews, Patricia L (1994), 'The farsite fire area simulator: Fire management applications and lessons of summer 1994', in *Proceedings of Interior West Fire Council Meeting and Program*, 209-16.
- Finney, Mark A and Ryan, Kevin C (1995), 'Use of the FARSITE fire growth model for fire prediction in US National Parks', *International emergency management and engineering conference*, edited by JD Sullivan, JL Wybo, and L. Buisson. Paris, France: International Emergency Management and Engineering Society (Citeseer).
- Finney, Mark, Grenfell, Isaac C, and McHugh, Charles W (2009), 'Modeling containment of large wildfires using generalized linear mixed-model analysis', *Forest Science*, 55 (3), 249-55.
- Fiorucci, Paolo, Gaetani, Francesco, and Minciardi, Riccardo (2002), 'Dynamic models for preventive management and real time control of forest fires', *Proceedings of 15th IFAC world congress in automatic control, Barcelona, Spain*.
- Fiorucci, P, et al. (2004), 'Real time optimal resource allocation in natural hazard management', *Proceedings of IEMSS* (Citeseer), 14.
- Flannigan, M. D. and Wotton, B. M. (2001), 'Climate, weather, and area burned', *Forest fires: behavior and ecological effects*. Academic Press, San Francisco, CA, 351-76.
- Frandsen, William H (1971), 'Fire spread through porous fuels from the conservation of energy', *Combustion and Flame*, 16 (1), 9-16.

- Fried, M. (2004), 'A level set based finite element algorithm for the simulation of dendritic growth', *Computing and Visualization in Science* 7, 97-110.
- Fried, Jeremy S and Fried, Burton D (1996), 'Simulating wildfire containment with realistic tactics', *Forest Science*, 42 (3), 267-81.
- Fujii, Nobuo, et al. (1980), 'A nonstationary model of firespreading', *Applied Mathematical Modelling*, 4 (3), 176-80.
- Gage, Michael E (1984), 'Curve shortening makes convex curves circular', *Inventiones mathematicae*, 76 (2), 357-64.
- Gage, Michael and Hamilton, Richard S (1986), 'The heat equation shrinking convex plane curves', *Journal of Differential Geometry*, 23 (1), 69-96.
- Gebert, Krista M, Calkin, David E, and Yoder, Jonathan (2007), 'Estimating suppression expenditures for individual large wildland fires', *Western Journal of Applied Forestry*, 22 (3), 188-96.
- Gibou, F. and Fedkiw, R. (2005), 'A Fourth Order Accurate Discretization for the Laplace and Heat Equations on Arbitrary Domains, with Applications to the Stefan Problem', *Journal of Computational Physics*, 202 (2), 577-601. doi: 10.1016/j.jcp.2004.07.018
- Gibou, F., et al. (2003), 'A Level Set Approach for the Numerical Simulation of Dendritic Growth', *Journal of Scientific Computing*, 19 (1-3), 183-99. doi: 10.1023/A:1025399807998
- Glicksman, ME and Lupulescu, AO (2004), 'Dendritic crystal growth in pure materials', *Journal of Crystal Growth*, 264 (4), 541-49.
- Glicksman, M. E., Schaefer, R. J., and Ayers, J. D. (1976), 'Dendritic growth - a test theory', *Metallurgical and Materials Transactions A*, 7 (11), 1747-59.
- Godrèche, Claude (1991), *Solids far from Equilibrium* (1: Cambridge University Press).
- Gonzalez, Rafael C, Woods, Richard Eugene, and Eddins, Steven L (2004), *Digital image processing using MATLAB* (Pearson Education India).
- Gorte, Julie Fox and Gorte, Ross D (1979), 'Application of economic techniques to fire management', *General technical report INT*; 53.
- Govindaraju, N. and Li, B.Q. (2002), 'A macro/micro model for magnetic stirring and microstructure formation during solidification', *Energy Conversion and Management*, 43 (3), 335-44
- Griffith, Cecilia Girz, et al. (1978), 'Rain estimation from geosynchronous satellite imagery-visible and infrared studies', *Monthly Weather Review*, 106 (8), 1153-71.
- Guo, Yuzhu, Billings, Steve A, and Coca, Daniel (2008), 'Identification of N-state spatio-temporal dynamical systems using a polynomial model', *International Journal of Bifurcation and Chaos*, 18 (07), 2049-57.
- Gupta, Sushil Chandra (2003), *The classical Stefan problem: basic concepts, modelling and analysis* (Elsevier).
- Haight, Robert G and Fried, Jeremy S (2007), 'Deploying wildland fire suppression resources with a scenario-based standard response model', *INFOR: Information Systems and Operational Research*, 45 (1), 31-39.
- Hill, James M. (1987), *One-dimensional Stefan problems: an introduction* (31: Longman Sc & Tech).

- Hinze, M. and Ziegenbalg, S. (2007), 'Optimal control of the free boundary in a two-phase Stefan problem', *Journal of Computational Physics*, 223 (2).
- Hirt, Cyril W and Nichols, Billy D (1981), 'Volume of fluid (VOF) method for the dynamics of free boundaries', *Journal of Computational Physics*, 39 (1), 201-25.
- Holmes, Thomas P, Huggett Jr, Robert J, and Westerling, Anthony L (2008), 'Statistical analysis of large wildfires', *The Economics of Forest Disturbances* (Springer), 59-77.
- HomChaudhuri, Baisravan, Kumar, Manish, and Cohen, Kelly (2010), 'Optimal fireline generation for wildfire fighting in uncertain and heterogeneous environment', *American Control Conference (ACC), 2010* (IEEE), 5638-43.
- Homchaudhuri, Baisravan, Kumar, Manish, and Cohen, Kelly (2013), 'Genetic Algorithm Based Simulation–Optimization for Fighting Wildfires', *International Journal of Computational Methods*, 10 (06), 1350035.
- Hottel, HC, Williams, GC, and Kwentus, GK (1971), 'Fuel preheating in free-burning fires', *Symposium (International) on Combustion* (13: Elsevier), 963-70.
- Hu, Xiaolin and Ntaimo, Lewis (2009), 'Integrated simulation and optimization for wildfire containment', *ACM Transactions on Modeling and Computer Simulation (TOMACS)*, 19 (4), 19.
- Hwang, CC and Xie, Yusheng (1984), 'Flame Propagation Along Matchstick Arrays On, Inclined Base Boards', *Combustion Science and Technology*, 42 (1-2), 1-12.
- Ilachinski, Andrew (2001), *Cellular automata: a discrete universe* (World Scientific).
- Ivancevic, Vladimir G and Ivancevic, Tijana T (2011), 'Ricci flow and nonlinear reaction–diffusion systems in biology, chemistry, and physics', *Nonlinear Dynamics*, 65 (1-2), 35-54.
- Javierre, E, et al. (2006), 'A comparison of numerical models for one-dimensional Stefan problems', *Journal of Computational and Applied Mathematics*, 192 (2), 445-59.
- Javierre-Perez, E (2003), *Literature Study: Numerical methods for solving Stefan problems* (Delft University of Technology).
- Jiang, Guang-Shan and Peng, Danping (2000), 'Weighted ENO schemes for Hamilton--Jacobi equations', *SIAM Journal on Scientific Computing*, 21 (6), 2126-43.
- Johnson, Edward Arnold and Miyanishi, Kiyoko (2001), *Forest fires: behavior and ecological effects* (593: Academic Press San Diego, CA).
- José, João Paulo Carvalho Marco Carola and Tomé, AB (2006), 'Forest fire modelling using rule-based fuzzy cognitive maps and voronoi based cellular automata'.
- Juric, D. and Tryggvason, G. (1996), 'A Front -Tracking Method for Dendritic Solidification', *Journal of Computational Physics*, 123 (1), 127-48.
- Kaneko, Kunihiko (1985), 'Spatiotemporal intermittency in coupled map lattices', *Progress of Theoretical Physics*, 74 (5), 1033-44.
- Kaneko, Kunihiko (1989), 'Pattern dynamics in spatiotemporal chaos: Pattern selection, diffusion of defect and pattern competition intermittency', *Physica D: Nonlinear Phenomena*, 34 (1), 1-41.

- Kang, S. and Zabaras, N. (1995), 'Control of the freezing interface motion in two-dimensional solidification processes using the adjoint method', *International Journal for Numerical Methods in Engineering*, 38, 63-80.
- Karafyllidis, Ioannis and Thanailakis, Adonios (1997), 'A model for predicting forest fire spreading using cellular automata', *Ecological Modelling*, 99 (1), 87-97.
- Kessler, D. A. and Levine, H. (1986), 'Stability of dendritic crystals', *Physical Review Letters*, 57 (24), 3069-72.
- Kessler, D. A., Koplik, J., and Levine, H. (1984), 'Geometrical models of interface evolution II. Numerical simulation', *Physical Review A*, 30 (6), 3161-74. doi: 10.1103/PhysRevA.30.3161
- Kessler, D. A., Koplik, J., and Levine, H. (1985), 'Geometrical models of interface evolution III. Theory of dendritic growth', *Physical Review A* 31 (3), 1712-17. doi: 10.1103/PhysRevA.31.1712
- Kessler, David A, Koplik, Joel, and Levine, Herbert (1986), 'Steady-state dendritic crystal growth', *Physical Review A*, 33 (5), 3352.
- Kessler, David A, Levine, Herbert, and Reynolds, William N (1990), 'Coupled-map lattice model for crystal growth', *Physical Review A*, 42 (10), 6125.
- Kidd, Chris, et al. (2003), 'Satellite rainfall estimation using combined passive microwave and infrared algorithms', *Journal of Hydrometeorology*, 4 (6), 1088-104.
- Kim, Yung-Tae, Goldenfeld, Nigel, and Dantzig, Jonathan (2000), 'Computation of dendritic microstructures using a level set method', *Physical Review E*, 62 (2), 2471.
- Kobayashi, Ryo (1993), 'Modeling and numerical simulations of dendritic crystal growth', *Physica D: Nonlinear Phenomena*, 63 (3-4), 410-23 doi: doi:10.1016/0167-2789(93)90120-P
- Krantz, Steven G (2008), *A guide to complex variables* (32: MAA).
- Kurz, W. and Fisher, D.J. (1998), *Fundamentals of Solidification*
- Langer, J.S. (1980), 'Instabilities and pattern formation in crystal growth', *Reviews of Modern Physics*, 52 (1), 1-28.
- Langer, J. S. and Müller-Krumbhaar, H. (1983), 'Mode selection in a dendritelike nonlinear system', *Physical Review A*, 27 (1), 499-514. doi: 10.1103/PhysRevA.27.499
- Larini, M, et al. (1998), 'A multiphase formulation for fire propagation in heterogeneous combustible media', *International Journal of Heat and Mass Transfer*, 41 (6), 881-97.
- Lautenberger, Chris (2013), 'Wildland fire modeling with an Eulerian level set method and automated calibration', *Fire Safety Journal*, 62, 289-98.
- Lee, Alfred Y. and Myerson, Allan S. (2006), 'Particle Engineering: Fundamentals of Particle Formation and Crystal Growth', *MRS Bulletin*, 31, 881-86.
- Lee, Yohan, et al. (2012), 'Deploying initial attack resources for wildfire suppression: spatial coordination, budget constraints, and capacity constraints', *Canadian Journal of Forest Research*, 43 (1), 56-65.
- Leung, S., Lowengrub, J., and Osher, S. (2004), 'An Adaptive Level Set Method for Stefan Problems'.
- Linn, Rodman, et al. (2002), 'Studying wildfire behavior using FIRETEC', *International journal of wildland fire*, 11 (4), 233-46.

- Linn, Rodman R, et al. (2000), 'Studying Complex Wildfire Behavior Using FIRETEC', *Third Symposium on Fire and Forest Meteorology*.
- Luke, R Harry and McArthur, Alan Grant (1978), 'Bush fires in Australia', *Bush Fires in Australia*.
- Mallet, Vivien, Keyes, DE, and Fendell, FE (2009), 'Modeling wildland fire propagation with level set methods', *Computers & Mathematics with Applications*, 57 (7), 1089-101.
- Marsden-Smedley, JB and Catchpole, WR (1995), 'Fire behaviour modelling in Tasmanian buttongrass moorlands. II. Fire behaviour', *International journal of wildland fire*, 5 (4), 215-28.
- Martell, DAVID L (2001), 'Forest fire management', (Academic Press, San Diego, CA), 527-83.
- Martin, O. and Goldenfeld, N. D. (1987), 'Origin of sidebranching in dendritic growth', *Physical Review A*, 35 (3), 1382-90.
- McCarthy, J.F. and Blake, N.W. (1996), 'A front tracking model for the rapid solidification of dendritic alloys', *Acta Materialia*, 44 (5), 2093-100
- Mercer, D Evan and Prestemon, Jeffrey P (2005), 'Comparing production function models for wildfire risk analysis in the wildland-urban interface', *Forest Policy and Economics*, 7 (5), 782-95.
- Morgan, John W and Tian, Gang (2007), *Ricci flow and the Poincaré conjecture* (3: American Mathematical Soc.).
- Morita, Kenichi, Margenstern, Maurice, and Imai, Katsunobu (1999), 'Universality of reversible hexagonal cellular automata', *RAIRO-Theoretical Informatics and Applications*, 33 (06), 535-50.
- Muraleetharan, Murugiah (2006), 'Evolution of curves by curvature flow', (Lehigh University).
- Muzy, Alexandre, et al. (2005), 'Specification of discrete event models for fire spreading', *Simulation*, 81 (2), 103-17.
- Nakaya, Ukichiro (1954), *Snow Crystals: Natural and Artificial* (Harvard University Press).
- Nelson, Ralph M (2001), 'Water relations of forest fuels', *Forest fires: behavior and ecological effects*. Academic Press, San Francisco, CA, 79-143.
- Osher, Stanley (1993), 'A level set formulation for the solution of the Dirichlet problem for Hamilton-Jacobi equations', *SIAM Journal on Mathematical Analysis*, 24 (5), 1145-52.
- Osher, S. and Sethian, J. (1988), 'Fronts propagating with curvature dependent speed: Algorithms based on a Hamilton-Jacobi formulation', *Journal of Computational Physics*, 79 (1), 12:49. doi: doi:10.1016/0021-9991(88)90002-2
- Osher, S.J. and Fedkiw, R. (2003), *Level Set Methods and Dynamic Implicit Surfaces* (New York: Springer).
- Pagnutti, C, Anand, M, and Azzouz, M (2005), 'Lattice geometry, gap formation and scale invariance in forests', *Journal of theoretical biology*, 236 (1), 79-87.
- Pak, Sang Il and Hayakawa, Tomohisa (2011), 'Forest fire modeling using cellular automata and percolation threshold analysis', *American Control Conference (ACC), 2011* (IEEE), 293-98.
- Pan, Yi and Billings, Stephen A (2008), 'The identification of complex spatiotemporal patterns using coupled map lattice models', *International Journal of Bifurcation and Chaos*, 18 (04), 997-1013.

- Parks, George M (1964), 'Development and application of a model for suppression of forest fires', *Management Science*, 10 (4), 760-66.
- Parlar, Mahmut (1983), 'Optimal forest fire control with limited reinforcements', *Optimal Control Applications and Methods*, 4 (2), 185-91.
- Pastor, E, et al. (2003), 'Mathematical models and calculation systems for the study of wildland fire behaviour', *Progress in Energy and Combustion Science*, 29 (2), 139-53.
- Peng, D. , et al. (1999), 'A PDE-Based Fast Local Level Set Method', *Journal of Computational Physics*, 155 (2), 410-38 doi: 10.1006/jcph.1999.6345
- Quartieri, Joseph, et al. (2010), 'A cellular automata model for fire spreading prediction', *Latest Trends on Urban Planning and Transportation*, 173-78.
- Rehm, R and McDermott, R (2009), 'Mathematical modeling of wildland–urban interface fires', *Mathematics and fire workshop* (Zaragoza).
- Reigber, Andreas, et al. (2013), 'Very-high-resolution airborne synthetic aperture radar imaging: Signal processing and applications', *Proceedings of the IEEE*, 101 (3), 759-83.
- Reiter, Clifford A (2005), 'A local cellular model for snow crystal growth', *Chaos, Solitons & Fractals*, 23 (4), 1111-19.
- Richards, Shane A, Possingham, Hugh P, and Tizard, James (1999), 'Optimal fire management for maintaining community diversity', *Ecological Applications*, 9 (3), 880-92.
- Richter, Hendrik (2008), 'Coupled map lattices as spatio-temporal fitness functions: Landscape measures and evolutionary optimization', *Physica D: Nonlinear Phenomena*, 237 (2), 167-86.
- Rolley, E. , Balibar, S. , and Graner, F. (1994), 'Growth shape of He3 needle crystals', *Physical Review E*, 49 (2), 1500-06.
- Rothermel, Richard C (1983), *How to predict the spread and intensity of forest and range fires* (US Department of Agriculture, Forest Service, Intermountain Forest and Range Experiment Station Ogden, UT, USA).
- Rothermel, Richard C (1991), 'Predicting behavior and size of crown fires in the Northern Rocky Mountains'.
- Rothermel, Richard C and Forest, Intermountain (1972), 'A mathematical model for predicting fire spread in wildland fuels'.
- Rothermel, Richard C, Anderson, Hal E, and Forest, Intermountain (1966), 'Fire spread characteristics determined in laboratory'.
- Rowell, Andrew and Moore, Peter F (2000), *Global review of forest fires* (Citeseer).
- Rubinshteĭn, LI (1971), *The stefan problem* (27: American Mathematical Soc.).
- Russo, G. and Smereka, P. (2000), 'A Remark on Computing Distance Functions', *Journal of Computational Physics*, 163 (1), 51-67.
- Saito, K (2001), 'Flames', *Forest fires: behavior and ecological effects*, 12-51.
- Sakaguchi, H. and Ohtaki, M. (1999), 'A coupled map lattice model for dendritic patterns', *Physica A*, 272, 300-13.

- Scott, Joe H and Burgan, Robert E (2005), 'Standard fire behavior fuel models: a comprehensive set for use with Rothermel's surface fire spread model', *The Bark Beetles, Fuels, and Fire Bibliography*, 66.
- Selvarasu, N, Nachiappan, Alamelu, and Nandhitha, NM (2010), 'Euclidean Distance Based Color Image Segmentation of Abnormality Detection from Pseudo Color Thermographs', *assessment*, 1793, 8201.
- Sethian, J. A. (1985), 'Curvature and the evolution of fronts', *Communications in Mathematical Physics*, 101 (4), 487-99. doi: 10.1007/BF01210742
- Sethian, J. A. (1989), 'Hypersurfaces Moving With Curvature-Dependent Speed: Hamilton-Jacobi Equations, Conservation Laws And Numerical Algorithms', *Journal of Differential Geometry*, 31, 131-61.
- Sethian, J.A. (1990), 'Numerical algorithms for propagating interfaces: Hamilton-Jacobi equations and conservation laws', *Journal of Differential Geometry*, 31, 131-61.
- Sethian , J. A. and Strain, J. (1992), 'Crystal Growth and Dendritic Solidification', *Journal of Computational Physics*, 98 (2), 231-53. doi: 10.1016/0021-9991(92)90140-T
- Shu, Chi-Wang and Osher, S. (1989), 'Efficient Implementation of Essentially Non-oscillatory Shock-Capturing Schemes II', *Journal of Computational Physics*, 83 (2), 32-78. doi: 10.1016/0021-9991(88)90177-5
- Sparhawk, William Norwood (1925), 'The use of liability ratings in planning forest fire protection'.
- Stepanov, Alexander and Smith, James MacGregor (2012), 'Modeling wildfire propagation with Delaunay triangulation and shortest path algorithms', *European Journal of Operational Research*, 218 (3), 775-88.
- Strain, J. (1990), 'Velocity effects in unstable solidification', *SIAM Journal of applied mathematics*, 50 (1), 1-15.
- Sullivan, A.L. (2007a), 'A review of wildland fire spread modelling, 1990-present, 1: Physical and quasi-physical models', *arXiv preprint arXiv:0706.3074*.
- Sullivan, AL (2007b), 'A review of wildland fire spread modelling, 1990-present 2: Empirical and quasi-empirical models', *arXiv preprint arXiv:0706.4128*.
- Sussman, M. , Smereka, P. , and Osher, S. (1994), 'A Level Set Approach for Computing Solutions to Incompressible Two-Phase Flow', *Journal of Computational Physics*, 114 (1), 146-59. doi: 10.1006/jcph.1994.1155
- Suzuki, Toshio, et al. (2002), 'Phase-field model of dendritic growth', *Journal of Crystal Growth*, 237, 125-31.
- Tan, Lijian and Zabaras, Nicholas (2006), 'A level set simulation of dendritic solidification with combined features of front-tracking and fixed-domain methods', *Journal of Computational Physics*, 211 (1), 36-63.
- Tang, Bohui, et al. (2008), 'Generalized split-window algorithm for estimate of land surface temperature from Chinese geostationary FengYun meteorological satellite (FY-2C) data', *Sensors*, 8 (2), 933-51.
- Tsang, Alan, Larson, Kate, and McAlpine, Rob (2013), 'Resource Sharing for Control of Wildland Fires', *AAAI*.

- Van Wagner, CE (1967), 'Calculations on forest fire spread by flame radiation'.
- Vaz, GC, André, JCS, and Viegas, DX (2004), 'Fire spread model for a linear front in a horizontal solid porous fuel bed in still air', *Combustion Science and Technology*, 176 (2), 135-82.
- Von Neumann, John (1951), 'The general and logical theory of automata', *Cerebral mechanisms in behavior*, 1-41.
- Von Neumann, John and Burks, Arthur W (1966), 'Theory of self-reproducing automata', *IEEE Transactions on Neural Networks*, 5 (1), 3-14.
- Vuik, C (1993), 'The solution of a one-dimensional Stefan problem', *PITMAN RESEARCH NOTES IN MATHEMATICS SERIES*, 149-49.
- Wang , S.L. and Sekerka, R. F. (1996), 'Algorithms for Phase Field Computation of the Dendritic Operating State at Large Supercoolings', *Journal of Computational Physics*, (127), 110-17.
- Ward, D (2001), 'Combustion chemistry and smoke', *Forest Fires: Behavior and Ecological Effects*, edited by EA Johnson, and K. Miyanishi, 55-77.
- Warren, J. A. (1995), 'How does a metal freeze? A phase-field model of alloy solidification', *IEEE Computational Science & Engineering*, 2 (2), 38 - 49 doi: 10.1109/99.388957
- Weber, RO (1989), 'Analytical models for fire spread due to radiation', *Combustion and Flame*, 78 (3), 398-408.
- Weber, RO (2001), 'Wildland fire spread models', *Forest fires: behavior and ecological effects. San Diego, CA: Academic Press. p*, 151-69.
- Wei, L. , et al. (2011), 'A cellular automaton model for the solidification of a pure substance', *Applied Physics A: Materials Science Processing*, 103, 123-33.
- Wei, Yu, et al. (2015), 'A chance-constrained programming model to allocate wildfire initial attack resources for a fire season', *Forest Science*, 61 (2), 278-88.
- Wettlaufer, JS, Jackson, Martin, and Elbaum, M (1994), 'A geometric model for anisotropic crystal growth', *Journal of Physics A: Mathematical and General*, 27 (17), 5957.
- Wiitala, Marc R (1999), 'A dynamic programming approach to determining optimal forest wildfire initial attack responses', *Fire Economics, Planning, and Policy: Bottom Lines*, 115.
- Williams, FA (1977), 'Mechanisms of fire spread', *Symposium (International) on Combustion* (16: Elsevier), 1281-94.
- Wolff, MF, Carrier, GF, and Fendell, FE (1991), 'Wind-aided firespread across arrays of discrete fuel elements. II. Experiment', *Combustion Science and Technology*, 77 (4-6), 261-89.
- Wolfram, Stephen (1983), 'Statistical mechanics of cellular automata', *Reviews of Modern Physics*, 55 (3), 601.
- Wolfram, Stephen (1994), *Cellular automata and complexity: collected papers* (1: Addison-Wesley Reading).
- Wuensche, Andrew and Adamatzky, Andrew (2006), 'On spiral glider-guns in hexagonal cellular automata: activator-inhibitor paradigm', *International Journal of Modern Physics C*, 17 (07), 1009-26.

- Yanagita, Tatsuo (1993), 'Coupled map lattice model for boiling', *ICHMT DIGITAL LIBRARY ONLINE*, 18.
- Yanagita, Tatsuo and Kaneko, Kunihiko (1997), 'Modeling and characterization of cloud dynamics', *Physical Review Letters*, 78 (22), 4297.
- Yang, Yingxu and Billings, SA (2000), 'Neighborhood detection and rule selection from cellular automata patterns', *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on*, 30 (6), 840-47.
- Ye, Rugang (1994), 'Global existence and convergence of Yamabe flow', *J. Differential Geom*, 39 (1), 35-50.
- Yeh, Anthony and Li, Xia Gar-On (2002), 'A cellular automata model to simulate development density for urban planning', *Environment and Planning B*, 29 (3), 431-50.
- Zabaras, N. , Ruanb, Y. , and Richmond, O. (1992), 'Design of two-dimensional stefan process with desired freezing front motions', *Numerical Heat Transfer, Part B: Fundamentals*, 21 (3), 307-25.
- Zhao, Y, Billings, SA, and Coca, D (2009), 'Cellular automata modelling of dendritic crystal growth based on Moore and von Neumann neighbourhoods', *International Journal of Modelling, Identification and Control*, 6 (2), 119-25.
- Zhao, Yifan, Wei, Hua-Liang, and Billings, SA (2012), 'A new adaptive fast cellular automaton neighborhood detection and rule identification algorithm', *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, 42 (4), 1283-87.
- Zhao, Y. , et al. (2007), 'Identification of a spatio-temporal model of crystal growth based on boundary curvature', *Research report (University of Sheffield. Department of Automatic Control and Systems Engineering)* (University of Sheffield).
- Zhao, Y., et al. (2009), 'Identification of the Transition Rule in a Modified Cellular Automata Model: the Case of Dendritic Nh4br Crystal Growth', *International Journal of Bifurcation and Chaos*, 19 (7), 2295-305.
- Zhao , Y , et al. (2011), 'Identification of geometrical models of interface evolution for dendritic crystal growth', *Physics Letters A*, 375 (7), 1084-91 doi: 10.1016/j.physleta.2011.01.012
- Zhu, MF and Hong, CP (2001), 'A modified cellular automaton model for the simulation of dendritic growth in solidification of alloys', *Isij International*, 41 (5), 436-45.