

ATHENS UNIVERSITY OF ECONOMICS AND BUSINESS
School of Information Sciences and Technology
Department of Computer Science

Wildfire Simulation Software for Intermittently Connected Networks

A dissertation submitted in partial satisfaction of the
requirements for the degree
Doctor of Philosophy

in

Computer Science

by

Ioannis Karakonstantis

Athens
July 2026

Copyright
Ioannis Karakonstantis, 2026
All rights reserved.

TABLE OF CONTENTS

Table of Contents	iii
List of Figures	vi
List of Tables	viii
Acknowledgements	ix
Vita and Publications	x
Abstract of the Dissertation	xii
Chapter 1 Introduction	1
1.1 Motivation	1
1.1.1 Wildfires	1
1.1.2 Mitigating Wildfire Risk with Computer Simulation	2
1.1.3 Edge vs. Cloud Simulations	3
1.1.4 Real-Time Wildfire Simulation for the Edge	5
1.1.5 Feasibility and Reliability Considerations	7
1.1.6 Wildfire Risk Assessment and Public Awareness Applicability	8
1.2 Contributions	9
1.3 Dissertation Outline	11
Chapter 2 Wildfire Spread Modeling	13
2.1 Computational Models	13
2.2 Workflow for Fire Dynamics Modeling	14
2.3 A Taxonomy of Wildfire Propagation Modeling Approaches	15
2.4 Raster-Based / Discrete Solution Space Wildfire Propagation Models	17
2.5 Cellular Automata	18
2.5.1 A Stochastic CA Definition	19
2.5.2 CA with Continuous State	23
2.6 The State of the Art in CA Models for Wildfire Simulation	24
2.6.1 Baseline Multivariable CA	24
2.6.2 Dynamic Neighborhood Configuration	25
2.6.3 Accessible Data-Driven CA	26
2.6.4 Operation-Ready Implementation	26
2.6.5 Continuous-State CA Implementation	27
2.6.6 Variable Simulation Time Step CA	28
2.6.7 Synergistic Vector-Based and CA Approach	29

	2.6.8	Enhancing Spatial CA Accuracy	29
	2.6.9	Lightweight Vegetation-Aware CA Model	31
	2.6.10	CA for Wildland–Urban Interface Areas	31
	2.6.11	CA Comparative Matrix	33
Chapter 3		Enhancing Wildfire Simulation Through Volunteered Geographic Information	36
	3.1	Crowdsourced Wildfire Spread and Status	36
	3.2	Crowdsourcing vs. Remote Sensing	37
	3.3	Crowdsourcing vs. Local Sensing	39
	3.4	Existing Crowdsourcing Approaches	40
	3.4.1	CITISENS	40
	3.4.2	FireLoc	41
	3.4.3	FDWithoutFire	42
	3.4.4	Early Wildfire Detection via Vapor Pressure Deficit	43
	3.4.5	A Multi-Source Data Fusion Framework for WUI Fire Detection	43
	3.4.6	Social Media Driven Fire Detection	44
	3.4.7	Multimodal Fire Spread Prediction Framework	45
	3.5	A Web-Based Edge Simulator for Limited Connectivity Scenarios	46
Chapter 4		A Lightweight Ontology for Real-Time Wildfire Simulation	53
	4.1	A Semantic Model for Wildfire Simulation	53
	4.2	Architectural Decisions	54
	4.3	Knowledge Representation Requirements	55
	4.3.1	Ontology Modeling Challenges	58
	4.4	Existing Ontological Approaches	59
	4.4.1	OntoFire	59
	4.4.2	Onto-Safe	60
	4.4.3	SoKNOS	60
	4.4.4	BeAWARE	61
	4.4.5	SIADEx / BACAREX	61
	4.4.6	WUI Fire System Ontology	62
	4.4.7	Fire Detection Ontology	62
	4.4.8	NOA Ontology	63
	4.4.9	Comparison and Limitations	63
	4.5	A Wildfire Ontology for Near Real-Time Simulation	65
	4.5.1	Handling of Duplicate and Ambiguous Inputs	66
	4.5.2	Representation Scheme	67
	4.5.3	Formulation of the Minimal Ontology	69

Chapter 5	Geometric Primitive Analysis in Noisy Crowdsourced Data . . .	73
5.1	Geospatial Uncertainty in Crowdsourced Inputs	73
5.2	From Crowdsourced Observations to Simulation Inputs . .	74
5.2.1	Conflict Handling	75
5.3	An Ant Colony Optimization Algorithm for Integrating User Inputs	78
5.3.1	A Baseline ACO Approach	79
5.3.2	VGI Fusion via Modified Ant Colony Optimization	81
5.3.3	Why an ACO-Based Correspondence Solver? . . .	87
5.3.4	A Modified Pheromone Update Model	90
5.4	Experimental Results	92
5.5	Result Analysis and Future Extensions	97
Chapter 6	Parameter Fine-Tuning in a CA Wildfire Simulator	99
6.1	Importance of Parameter Adjustment in CA Wildfire Mod- els	99
6.1.1	Calibration and Parameter Estimation in CA Wild- fire Models	102
6.2	CA Simulation Algorithms Considered	103
6.2.1	Baseline	103
6.2.2	Weather Coupled	104
6.2.3	Dynamic Neighborhood	105
6.3	Ant Colony Optimization for Continuous Optimization Problems	105
6.3.1	ACO _R	106
6.3.2	Objective Function	108
6.3.3	Theoretical Elliptical Envelope Algorithm	109
6.3.4	Formulating CA Parameter Tuning as a COP . .	113
6.3.5	Parameter Tuning Complexity	115
6.4	Experimental Results	118
6.4.1	Area of Interest	118
6.4.2	Meteorological Conditions	118
6.4.3	Simulation Procedure	120
6.4.4	Baseline CA	122
6.4.5	Weather-Coupled CA	123
6.4.6	Dynamic Neighborhood CA	125
6.4.7	Analysis of Simulation and Fine-Tuning Results .	126
Chapter 7	Conclusions	131
Appendix A	Acronyms	136
Bibliography		138

LIST OF FIGURES

Figure 1.1:	Architecture of the proposed wildfire simulation and optimization framework.	6
Figure 1.2:	The contributions of this thesis	7
Figure 2.1:	A taxonomy of wildfire models.	16
Figure 2.2:	Cellular Automata with rectangular or hexagonal cells.	34
Figure 2.3:	Illustration of P_{burn} within a Moore neighborhood configuration.	35
Figure 2.4:	Example of combined vector-based and CA approach.	35
Figure 3.1:	Overview of the VGI interface.	48
Figure 3.2:	User inserted fuel data and wildfire updates.	50
Figure 3.3:	High-level representation of the proposed <i>VGI / DTN</i> distributed architecture.	51
Figure 4.1:	Overlapping hierarchical structures in an ontology.	57
Figure 4.2:	A hierarchical structure with relationships of different types.	58
Figure 4.3:	Digitization of the same forested area by different users.	67
Figure 4.4:	High-level representation of factors contributing to wildfire behavior.	70
Figure 4.5:	An overview of the proposed WildFire Simulation Ontology (WSO).	72
Figure 5.1:	Data integration process for heterogeneous and noisy user observations.	74
Figure 5.2:	Categories of input conflicts addressed by the WSO and ACO.	76
Figure 5.3:	User input conflict taxonomy handling.	78
Figure 5.4:	Bipartite graph topology for the QAP-formulated contour-matching ACO algorithm.	80
Figure 5.5:	Flowchart of a mobile wildfire simulator.	84
Figure 5.6:	Technical overview of the ACO-based fusion pipeline for crowd-sourced simulation inputs.	85
Figure 5.7:	Stochastic consensus shape fusion using ACO-based algorithm.	90
Figure 5.8:	The same vegetated area in East Attica as modeled by four users.	93
Figure 5.9:	A vegetated area in East Attica projected on the map by a volunteer.	93
Figure 5.10:	(a) Optimal assignment between I (random user) and J (remaining users) sets (b) simplified primitive.	94
Figure 5.11:	A noisy representation of a fireline obtained from the same group of volunteers.	95
Figure 5.12:	Similarity scores of baseline and modified ACO methods.	97

Figure 6.1:	Landscape discretization in a CA simulator with two classes of forestry fuels.	101
Figure 6.2:	Standard and extended Moore neighborhood.	106
Figure 6.3:	Discrete approximation of the continuous search space.	107
Figure 6.4:	Comparison of fire ellipse shapes for different L/B ratios.	111
Figure 6.5:	Progressive reconstruction of theoretical elliptical envelope. . .	112
Figure 6.6:	Dynamic fine-tuning of CA parameters.	116
Figure 6.7:	Study area near Glyfada, Greece.	119
Figure 6.8:	Environmental conditions in the Glyfada, Greece area.	121
Figure 6.9:	Comparison of burned area (black) after 6-hours (a) without and (b) after parameter fine-tuning.	123
Figure 6.10:	Comparison of burned area (black) after 6-hours (a) without and (b) after parameter fine-tuning.	125
Figure 6.11:	Comparison of burned area (black) after 6-hours (a) without and (b) after parameter fine-tuning.	128
Figure 7.1:	Conceptual architecture for a wildfire digital twin	134

LIST OF TABLES

Table 2.1:	A comparative overview of state-of-the-art CA models.	33
Table 4.1:	A comparative overview of ontology-based approaches.	64
Table 4.2:	WildFire Simulation Ontology (WSO)	71
Table 5.1:	ACO parameters and baseline weights used in the experiments. .	96
Table 6.1:	Assigned P_{den} and P_{veg} coefficients for each CLC class used in the CA model.	120
Table 6.2:	Parameters used in the initial CA simulation.	122
Table 6.3:	Parameters used for the ACO _R algorithm.	122
Table 6.4:	Optimal parameter set for the baseline CA model	123
Table 6.5:	Parameters used in the initial CA simulation for the weather- coupled variant.	124
Table 6.6:	Parameters used for the ACO _R algorithm.	124
Table 6.7:	Optimal parameter set for the weather-coupled CA model. . . .	125
Table 6.8:	Parameters used in the initial CA simulation for the dynamic neighborhood configuration variant.	126
Table 6.9:	Optimal parameter set for the dynamic neighborhood CA model.	126
Table 6.10:	Comparison of burned areas for each examined test scenario. . .	127

ACKNOWLEDGEMENTS

I am sincerely grateful to the members of my advisory committee: Professor George Xylomenos, Professor George C. Polyzos and Professor Vasilios A. Siris. I would like to take this opportunity to express my sincere gratitude to my supervisor George Xylomenos for his invaluable guidance, continuous support, and constructive feedback throughout the course of my doctoral studies.

I would also like to extend my sincere thanks to the remaining members of my doctoral examination committee: Professor Iordanis Koutsopoulos, Associate Professor Stavros Toumpis, Associate Professor Michael Chalaris and Assistant Professor Palaiologos Palaiologou for taking the time to carefully review this dissertation. Their valuable comments, perceptive observations, and constructive suggestions have strengthened the quality of this work, reflecting the breadth of expertise represented across different scientific disciplines. I greatly appreciate the time, effort, and expertise they dedicated to the review process.

Finally, I would like to thank my colleagues in the *Mobile Multimedia Laboratory* at Athens University of Economics and Business for fostering a collaborative, stimulating, and supportive research environment. In no particular order, I would like to acknowledge Dr. Nikos Fotiou, Dr. Merkouris Karaliopoulos, Dr. Ioannis Thomas, Dr. Konstantinos Tsioutas, Dr. Livia Chatzieleftheriou, Dr. Christos Karapapas, Dr. Iakovos Pittaras and Andreas Papidas who joined the lab at the same time with me. Working alongside them has been one of the most rewarding aspects of my doctoral journey. In addition, I would like to express my gratitude to all the PhD and MSc students of MMLab. Although I may not have mentioned everyone individually, I sincerely appreciate your support, collaboration, and the many discussions and experiences we shared throughout these years.

VITA

2013	B. Sc. in Computer Science, University of Piraeus, Greece
2015	M. Sc. in Computational Science, National and Kapodistrian University of Athens, Greece
2019	M. Sc. in Geoinformatics, National and Technical University of Athens, Greece
2023	M. Sc. in Analysis and Management of Manmade and Natural Disasters, International Hellenic University, Greece
2026	Ph. D. in Computer Science, Athens University of Economics and Business, Greece

PUBLICATIONS

Journal Publications

I. Karakonstantis and G. Xylomenos, “Optimizing Fire Perimeter Geometry in Cellular Automata Wildfire Models Using Ant Colony Optimization,” in *Fire*, MDPI, vol. 9, no. 5, pp.212, 2026.

I. Karakonstantis and G. Xylomenos, “A Review of Two-Dimensional Cellular Automata Models for Wildfire Simulation: Methods, Capabilities, and Limitations,” in *Fire*, MDPI, vol. 9, no. 3, pp.108, 2026.

I. Karakonstantis and G. Xylomenos, “Influence of Input Data Uncertainty on Cellular Automata-Based Wildfire Spread Simulation,” in *Information*, MDPI, vol. 17, no. 3, pp.289, 2026.

I. Karakonstantis and G. Xylomenos, “Improving Wildfire Simulations via Geometric Primitive Analysis in Noisy Crowdsourced Data,” in *Applied Sciences*, MDPI, vol. 15, no. 16, pp.8844, 2025.

I. Karakonstantis and A. Vlachos, “Bat Algorithm Applied to Continuous Constrained Optimization Problems,” in *Journal of Information and Optimization Sciences*, Taylor & Francis, vol. 42, no. 1, pp.57-75, 2021.

I. Karakonstantis and A. Vlachos, “Hybrid Ant Colony Optimization for Continuous Domains for Solving Emission and Economic Dispatch Problems,” in *Journal of Information and Optimization Sciences*, Taylor & Francis, vol. 39, no. 3, pp.651-671, 2018.

I. Karakonstantis and A. Vlachos, “Ant Colony Optimization for Continuous Domains Applied to Emission and Economic Dispatch Problems,” in *Journal of Information and Optimization Sciences*, Taylor & Francis, vol. 36, no. 1-2, pp.23-42, 2015.

I. Karakonstantis and A. Vlachos, “The Ant Colony Optimisation Solving Continuous Problems,” in *International Journal of Computational Intelligence Studies*, Inderscience, vol. 2, no. 3-4, pp.350, 2013.

Refereed Conference Papers

I. Karakonstantis and G. Xylomenos, “A Web-Based Simulation Interface for Fire Dynamics Analysis and Decision Support,” in *Proceedings of the International Conference on Telecommunications (ICT)*, 2026.

I. Karakonstantis and G. Xylomenos, “An Improved Wildfire Simulation Engine,” in *Proceedings of the International Conference on Emerging Internet, Data & Web Technologies (EIDWT)*, 2026.

I. Karakonstantis and G. Xylomenos, “An Ontology for Dynamic Wildfire Simulations,” in *Proceedings of the IEEE Conference on Standards for Communications and Networking (CSCN)*, 2025.

G. Xylomenos, G. Pavlou, I. Psaras, and I. Karakonstantis, “Named Functions at the Edge,” in *Proceedings of the IEEE Symposium on Computers and Communications (ISCC)*, 2019.

ABSTRACT OF THE DISSERTATION

Wildfire Simulation Software for Intermittently Connected Networks

by

Ioannis Karakonstantis

Doctor of Philosophy in Computer Science

Athens University of Economics and Business, Athens, 2026

Professor George Xylomenos, Chair

Over the past few years, wildfire incidents have been increasing at an alarming rate, particularly in wildland-urban interface zones, posing severe threats to human life, infrastructure, and ecological systems. The conventional wildfire simulation software used by fire fighting teams suffers from a fundamental limitation: it relies on predefined, statically calibrated environmental parameters and fuel classification schemes that cannot adapt as conditions evolve during a fire event. This dissertation addresses this limitation through the design of a software system that continuously ingests near-real-time field observations submitted by first responders via Volunteered Geographic Information (VGI) platforms over both traditional and Delay Tolerant Networks (DTN) and uses them to drive and recalibrate an active wildfire simulation in near-real-time. The result is a new class of simulation

software, capable of adapting to the evolving fire behavior, bridging geographic information science, computer science, and computational fire modeling.

This thesis presents three scientific contributions to the development of this class of simulation software. The first is a Wildfire Simulation Ontology (WSO), a domain ontology implemented as a software module that provides structured, machine interpretable representations of the environmental components relevant to wildfire simulation. WSO addresses the heterogeneity inherent in crowdsourced observations arriving in diverse geometric formats, at varying spatial resolutions, and with inconsistent attribute vocabularies. Its aim is to normalize them into semantically consistent, GeoJSON encoded outputs that can be directly consumed by downstream simulation software components.

The second component is a data fusion pipeline that resolves the record linkage problem arising when multiple participants independently describe the same environmental feature at overlapping or proximate times. Raw crowdsourced data are inherently redundant and geometrically inconsistent, so they must be linked to a single representative geometric representation before they can serve as simulation inputs. We address this problem as a two-stage density-based clustering of WSO instances, followed by a contour correspondence process based on Ant Colony Optimization (ACO) that reconstructs a semantically consistent representation of the geometric primitives that can serve as an input to the simulation engine.

The third software component is a real-time calibration engine that keeps the simulator continuously aligned with observed fire behavior as new field data arrives. The wildfire simulator is implemented as a Cellular Automata (CA) model, in which the temporal evolution of simulation cells is governed by a set of continuous parameters encoding fuel ignitability, wind coupling coefficients, and topography. In conventional software solutions, these parameters are fixed during initialization and cannot be adapted to reduce any divergence between the simulated and observed fire behavior. We propose a dynamic fine-tuning mechanism based on Ant Colony Optimization for Continuous Domains (ACO_R) that searches the parameter space to identify the optimal parameter configuration, and show that it can be applied to diverse CA-based simulation models.

Chapter 1

Introduction

1.1 Motivation

1.1.1 Wildfires

For millions of years, wildfires have served as a natural mechanism to restructure and regenerate landscapes and ecosystems. In recent years, however, an alarming increase in wildfire events, especially near inhabited areas, has raised grave concerns about the dangers to human lives [1]. Even more alarmingly, researchers project a continued and accelerating increase in both wildfire danger and total burned area in the coming decades [2]. The ecological impact is also severe, as wildfires release massive amounts of CO₂, degrade land, and disrupt ecosystems for decades, leading to wildlife displacement [3]. Additional land and water quality deterioration is expected after a wildfire, due to poor water infiltration and ash deposition in burned areas [4]. Moreover, critical communication and transportation infrastructure, agricultural lands, pastures, and residential buildings are frequently among the most severely affected assets, as human activity increasingly encroaches upon previously uninhabited landscapes [3].

It is crucial to recognize that fire is not merely a destructive force, but a fundamental evolutionary component that has shaped our planet throughout its history. Wildfires can promote nutrient cycling by transforming accumulated biomass into ash, thereby releasing essential elements such as nitrogen and phosphorus in

forms that can be rapidly absorbed by soil microorganisms and the remaining vegetation [5]. Furthermore, many ecosystems—particularly Mediterranean shrublands and boreal forests—have evolved fire-adapted traits, with reproduction and regeneration in many species being dependent on periodic fire, a phenomenon described as fire-dependent reproduction [6, 7, 8]. Additionally, low-intensity fires can function as a natural regulatory mechanism by reducing accumulated surface fuel loads, such as dead wood and leaf litter, and by influencing pest dynamics, including bark beetle populations, which may otherwise contribute to widespread timber mortality [9]. Building upon this ecological necessity, prescribed (or controlled) burning has become a vital land management tool that mimics these natural fire regimes under safe conditions. By deliberately introducing a low-intensity fire, planners can reduce hazardous fuel loads (e.g., dead wood, underbrush) that would otherwise feed catastrophic wildfires [10]. However, given the inherent risks of escaping control, the safe and effective implementation of prescribed fire requires predicting and monitoring fire spread, exactly as in an uncontrolled wildfire.

1.1.2 Mitigating Wildfire Risk with Computer Simulation

Despite their ecological benefits, wildfires can have devastating consequences when they occur in the *Wildland–Urban Interface* (WUI), the transitional zone between wildland vegetation and human development [11, 12, 13]. In these environments, fires can rapidly threaten communities and critical infrastructure like power lines and transportation networks. Consequently, accurate wildfire spread prediction has become essential for preparedness and response, with simulation software serving as a valuable tool for anticipating fire behavior and supporting operational decision making in tasks such as evacuation planning and fire suppression.

Simulations are equally valuable for prescribed burning; simulations can be executed throughout the planning process, serving as an additional decision-support system for determining when environmental conditions are safe enough to proceed. Indeed, models proposed for general wildfire simulations have also been applied to prescribed burning scenarios [14].

Using computational models to predict wildfire behavior is far from a new

approach. Extensive research has been conducted to incorporate weather conditions, terrain data, fuel types, and vegetation composition in fire behavior simulations, so as to improve their accuracy. Authorities and firefighting personnel usually rely on these simulations to revise evacuation plans, deploy crews on key points, and identify areas subject to extreme risk. Moreover, costly and time-consuming proactive measures can be tested before being implemented in the real world. For example, the effectiveness of logging in dense forestry areas, the construction of firebreaks, and the trimming of trees can be evaluated before being implemented, saving time and costs. Beyond short-term simulations, we can adjust our fire suppression strategies by projecting how wildfires may evolve under different scenarios under different climate change scenarios. In other words, a wildfire simulation helps transform uncertainty about future fire behavior into actionable intelligence that supports effective planning and decision-making.

Simulation techniques generally fall into two broad categories: vector-based methods with an infinite solution space (e.g., partial differential equation solvers, Huygens’ principle-based models) and raster-based methods with a discrete solution space, such as *Cellular Automata* (CA) and Markov-based approaches. A substantial body of research has been conducted on CA due to their conceptual simplicity, computational efficiency, robustness, and extensive validation across a wide range of scientific publications and real world applications. CA are based on discrete, typically square, lattices, which naturally lend themselves to highly efficient parallel computation. This raster-based representation of the simulation solution space further simplifies implementation and allows the large-scale modeling of vast wildlands in a unified manner. The increasing popularity of CA has resulted in the development of enhanced variants that extend beyond the classical definition of CA [15].

1.1.3 Edge vs. Cloud Simulations

Wildfire simulation applications require the rapid processing of continuously changing environmental data, including weather conditions, terrain characteristics, vegetation properties, and sensor measurements when they are available. Cloud

computing is considered the default option for computationally intensive simulations due to its massive processing power, centralized resource management, and scalability. Cloud platforms enable the execution of large-scale simulations at fine resolutions that would otherwise exceed the computational capabilities of local, portable devices. However, while cloud computing is well suited for offline analysis and long-term predictive modeling, its architectural characteristics limit its suitability for real-time wildfire simulation, which requires continuous execution by ground personnel to accurately track wildfire behavior as environmental conditions evolve. Although the necessity for frequent invocation of the simulation engine may appear to support the adoption of a cloud-based architecture, a centralized simulation depends on reliable and continuous network connectivity in order to gather all inputs at the cloud and propagate all outputs to the edge.

In emergency situations such as wildfires, however, network congestion or temporary loss of internet connectivity can prevent simulations from executing or updating, precisely when timely information is most critical for decision-making. Edge computing addresses this issue by enabling computational resources to remain operational even when connectivity to the cloud is limited or unavailable.

In addition, the effectiveness of a cloud-based wildfire simulation depends heavily on the availability of timely and accurate input data, as the use of outdated data can significantly reduce the accuracy of the simulation results. While cloud-based systems can operate using pre-existing datasets, such as periodically updated satellite-obtained products, these datasets may not accurately capture the rapidly evolving conditions of an active wildfire. To achieve a more accurate representation of the current fire state, they may also need to access data collected from field-deployed sensors, *Unmanned Aerial Vehicles* (UAVs), weather stations, or firefighting personnel. However, this approach reinforces the dependence on network connectivity, as all observations must first be transmitted to remote cloud servers before simulations can be updated. In this context, the communication network becomes a critical component of the simulation pipeline, reinforcing the need to shift the computational paradigm toward an edge-based architecture.

1.1.4 Real-Time Wildfire Simulation for the Edge

This dissertation investigates the gaps arising in the development of a near-real-time wildfire simulation at the network edge, specifically on mobile, resource constrained devices (e.g., smartphones with limited CPU and battery capacity), where users (firefighters, stakeholders) operating in the field may experience poor connectivity and are therefore unable to access external APIs, fuel databases, or other remote data services. To alleviate these constraints, we adopt a CA-based fire spread simulator due to its lightweight computational footprint, along with a *Delay-Tolerant Networking* (DTN) scheme capable of operating effectively in intermittently connected environments.

Within this context, we developed a simple yet comprehensive wildfire simulation framework (Figure 1.1), designed to operate under such constraints, leveraging fire-related crowdsourced data collected by firefighters during suppression operations, who also constitute its primary *end users*. During the design and development of this framework, we identified several key limitations of CA models and gaps in the existing literature, which we addressed in order to provide a coherent and meaningful contribution. Specifically, this thesis focuses on three open problems that conventional wildfire simulation software leaves unresolved.

- The absence of a standardized, machine interpretable representation of the wildfire environment for simulation software. Existing tools rely on rigid, predefined data schemas that cannot accommodate the heterogeneous, multi-source observations produced during field operations. Furthermore, these tools often require high network bandwidth to access up-to-date data sources. There is no established software mechanism for ingesting crowdsourced environmental data which vary in geometry, resolution, and vocabulary and normalizing them into a form that a simulation engine can directly consume.
- The lack of any data fusion capability in current simulation pipelines. When multiple field observers independently report on the same environmental feature, existing software has no means of reconciling these redundant and geometrically inconsistent contributions into a single, coherent simulation in-

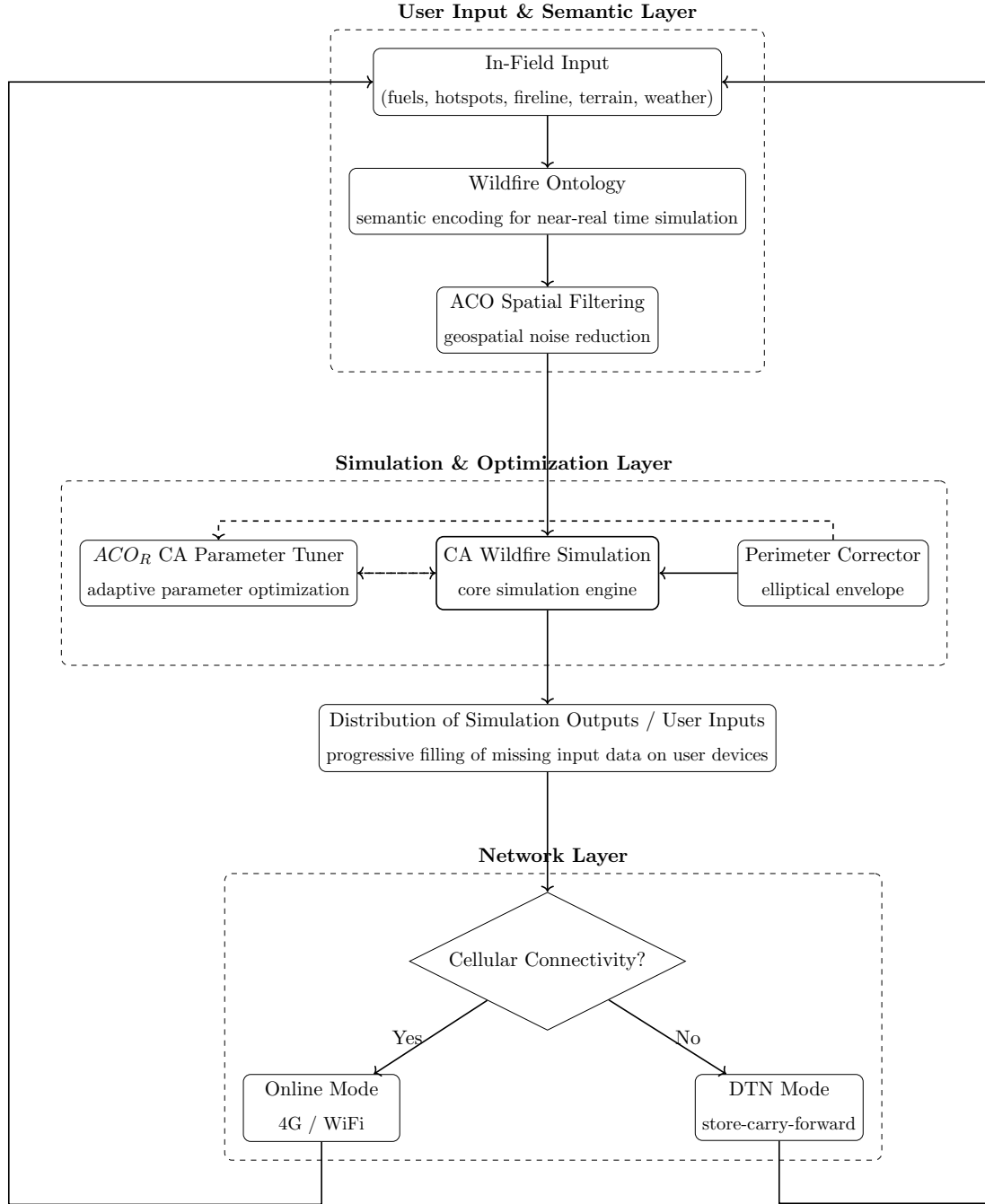


Figure 1.1: Architecture of the proposed wildfire simulation and optimization framework. The system integrates semantic preprocessing, ACO-based geospatial noise filtering, cellular automata simulation, and ACO_R -based adaptive optimization, with connectivity-aware deployment modes.

put. This gap prevents the effective exploitation of the collective situational awareness that first responders can provide.

- The inability of existing simulators to recalibrate their parameters in response to incoming observational data. Once initialized, conventional simulation software maintains fixed fire behavior parameters, even when significant deviations arise between predicted and theoretically expected fire behavior. This is a critical limitation in operational scenarios where simulation output is designed to be derived from near-real-time data acquired in-field.

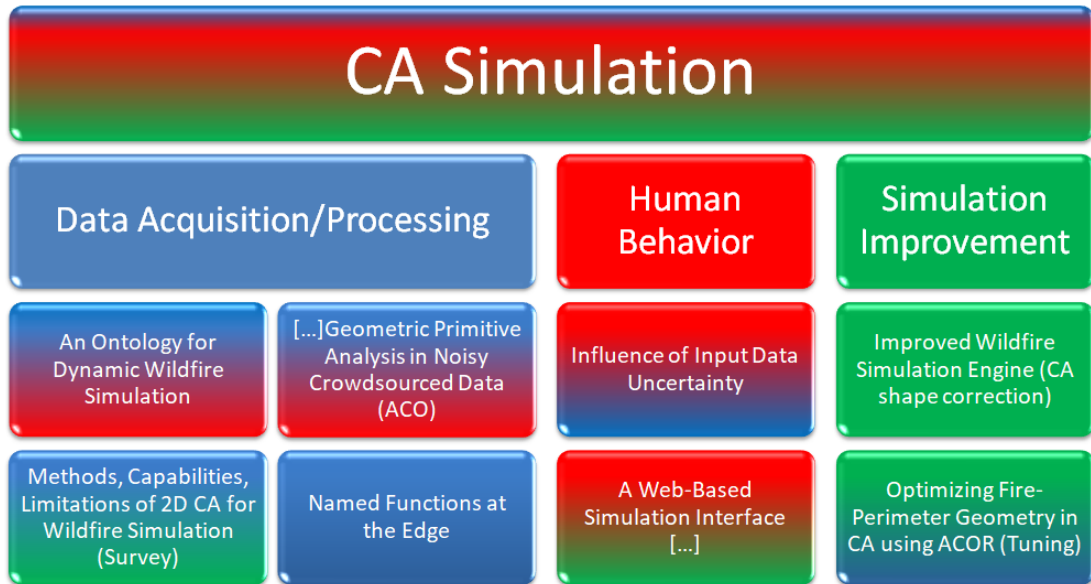


Figure 1.2: The contributions of this thesis, categorized into three areas: data acquisition, human behavior and improvement of CA simulation.

1.1.5 Feasibility and Reliability Considerations

Although the proposed approach provides several advantages in terms of responsiveness, scalability, and resilience, its practical applicability depends on whether such a system can realistically be deployed and whether the collected data can achieve sufficient accuracy for real-time wildfire simulation. It is crucial to recognize that wildfire conditions continuously evolve due to fire dynamics,

suppression tactics, and changing weather conditions. As a result, even prediction outputs from leading fire-simulation software can rapidly diverge from real-world spreading conditions, potentially becoming inaccurate within only a few minutes [16, 17, 18]. Such deviations may be more pronounced in CA-based implementations [15], which prioritize computational efficiency over detailed physical representation. However, their ability to provide fast and reliable approximations allows frequent simulation updates, particularly in environments where network connectivity to centralized computing resources is limited or unavailable.

From a deployment perspective, a VGI-based wildfire simulation approach offers important advantages compared with approaches that rely mainly on dense sensor networks, specialized monitoring equipment and remote-sensing data. The presence of firefighters already operating in or near affected regions significantly reduces the infrastructure requirements and allows observations to be collected from the affected area, without the need for permanent physical installations and network coverage.

However, the effectiveness of a VGI-based approach depends on the availability and expertise of contributors to provide reliable observations during wildfire events. The number of active contributors may vary significantly depending on the location, severity of the event, and accessibility of the affected area. Therefore, VGI should not be considered as a guaranteed replacement for dedicated sensing systems but rather as a flexible and scalable data source that can be deployed when other approaches fail. In practice, the most reliable wildfire simulation framework would likely combine multiple sources, including human observations, sensor measurements, satellite information, and UAV-based data.

1.1.6 Wildfire Risk Assessment and Public Awareness Applicability

Beyond its use as a computational model for predicting wildfire evolution, the proposed real-time wildfire simulation system can serve as a tool for improving risk assessment and increasing public awareness. The ability to continuously integrate updated observations and generate dynamic predictions provides

an opportunity to move from static wildfire risk maps toward more adaptive and situation-aware risk assessment approaches. Such a system can provide citizens, civil protection authorities, and local emergency services with updated information regarding the potential evolution of an ongoing wildfire event. Factors such as wind variations, changing fire intensity, fuel availability, and newly observed fire fronts can significantly alter the expected evolution of a fire.

Furthermore, human mobility patterns can be inferred through participant monitoring, enabling the detection of behaviors such as unexpected population gatherings in specific areas or rapid movement patterns. Such real-time behavioral observations can complement existing emergency alert systems, such as the 112 service [19], by providing additional situational awareness for evacuation planning and response coordination. While centralized alert mechanisms are essential for disseminating official warnings, integrating dynamic human mobility information can support more adaptive evacuation strategies by reflecting the actual movement and response of the population during evolving wildfire events.

From the perspective of civil protection agencies and local fire services, the system can function as a decision-support tool by providing an additional source of real-time information. Simulation results and user-submitted data can assist authorities in identifying potentially affected areas, prioritizing resources, and planning evacuation strategies. The objective is not to replace existing simulation and decision-making software tools, but to provide an additional layer that supports faster and more informed decision-making.

1.2 Contributions

The overarching goal of this thesis is the advancement and deployment of CA-based methods for near-real-time wildfire simulation [20]. To address the multifaceted challenges associated with modeling spatiotemporal fire dynamics particularly in field deployments, edge environments, and network coverage characterized by intermittent connectivity, the contributions are systematically organized within a tripartite taxonomy. The contribution taxonomy presented below is organized

into three principal core areas of investigation (Figure 1.2):

- **Data Acquisition and Processing (Blue):** it focuses on the collection, structuring, and edge-native processing of environmental and geometric inputs generated by human observers.
- **Human Behavior and Uncertainty (Red):** investigates the human element, specifically addressing the nature of crowdsourced observations and the inherent uncertainties of human provided data.
- **Simulation Enhancement (Green):** enhancements applied directly to the CA engine to increase simulation accuracy, correct geometrical artifacts, and optimize computational performance.

The modules comprising this framework span both foundational domain research and cross-cutting contributions. More specifically, in this thesis we focus on three research problems:

- In **Chapter 4** we consider the intersection of data acquisition and human behavior (Blue & Red). To address the friction between in-field human observation and computational data requirements we propose a GeoJSON-based *Wildfire Simulation Ontology* (WSO) for near-real-time wildfire simulation [21] which establishes a standardized vocabulary, translating subjective human observations into formalized geospatial entities. This contribution seeks to reduce ambiguity in user’s input data, an issue known to affect simulation quality, as demonstrated in our previous work [16], while ensuring a lightweight, platform-independent design that can be readily parsed using web technologies.
- In **Chapter 5** we consider the intersection of human interaction with computational simulation (Red & Green) that addresses the unreliability and uncertainty in crowdsourced, user-submitted simulation inputs. The contribution involves the analysis and reconciliation of the geometric primitives submitted during crowdsourced information gathering; we utilize an ACO-based algorithm to filter noise and resolve spatial conflicts. More precisely,

crowdsourced simulation inputs are clustered over time and subsequently processed using an ACO-based algorithm [22] to detect duplicates, perform shape correction, and eliminate potential noise and outliers. Evaluation of the method is conducted on geometric representations provided by volunteers, including two-dimensional polygons (e.g., fuel areas) and line strings (e.g., firelines).

- In **Chapter 6** we consider the intersection of progressive data acquisition with computational simulation (Blue & Green) that addresses the well-known limitation [15] of CA models, namely their tendency to be poorly tuned or overfitted to specific settings due to lack in mapping parameters to physical fire-related properties. This contribution introduces a fine-tuning scheme running at the edge, based on *Ant Colony Optimization for Continuous Domains* (ACO_R), that dynamically tests CA parameters against a theoretical elliptical envelope as new data become available and provides new parameters for future simulations.

1.3 Dissertation Outline

The primary aim of this dissertation is the design and implementation of a software framework that addresses fundamental open problems in existing wildfire simulation implementations. Contemporary simulation tools, despite their operational importance, share a common architectural constraint: they are built around static data pipelines in which environmental parameters (e.g., fuel characteristics, weather conditions, topographic factors) are calibrated offline and remain fixed throughout the duration of a fire event. This design assumption, while reasonable under controlled conditions, renders such software unable to respond to the dynamic and rapidly evolving nature of real wildfires, ultimately limiting both simulation accuracy and operational utility.

The remainder of the dissertation is organized as follows. Chapter 2 provides background and related work regarding computational models for wildfire simulation and a comprehensive taxonomy of the methods presented in literature.

Moreover, we describe a typical CA-based simulator for wildfire spread in two dimensions. Additionally, a brief summary of the key limitations of CA-based methods is presented based on a recent review [15]. In Chapter 3 we present advances in crowdsourcing-based platforms aimed at alleviating wildfire-related risks, particularly in WUI zones. These platforms enable early detection of fire events using ingested crowdsourced data, with the goal of mitigating risks to nearby civilians and providing timely alerts about potential hazards. In the same chapter, our wildfire simulation framework is presented from a more technical perspective, with particular emphasis on the user interface. In Chapter 4 we present our *Wildfire Simulation Ontology* (WSO), a GeoJSON-encoded ontology that provides a common framework for representing wildfire behavior. The ontology is designed to support our near-real-time edge-based simulation approach by enabling a standardized and interoperable representation of wildfire simulation data. WSO adopts a lightweight, platform-independent design that can be readily parsed and processed using standard web technologies, facilitating efficient integration across heterogeneous systems. In Chapter 5 we present an *Ant Colony Optimization* (ACO)-based data fusion algorithm, together with a modified pheromone update scheme, for fusing information derived from geometric primitives submitted during crowdsourced data collection. This approach leverages collective observations to produce a consistent and reliable representation of the evolving wildfire situation. In Chapter 6 we present a fine-tuning scheme based on *Ant Colony Optimization for Continuous Domains* (ACO_R) that dynamically calibrates the parameters of the CA model as new observations become available. This adaptive optimization process continuously improves the accuracy of the wildfire simulation by incorporating newly acquired data. Finally, in Chapter 7, we summarize the main findings of this thesis and discuss potential directions for future work.

Chapter 2

Wildfire Spread Modeling

2.1 Computational Models

The problem of wildfire prediction has gained a lot of attention in the literature and a plethora of wildfire models have been proposed. These models aim to predict how wildfires ignite, spread, and behave under various environmental conditions and assumptions. One of the most fundamental and widely acknowledged classifications of wildfire models relies on the underlying principles that they implement, which allows for their systematic categorization into three distinct types: empirical, semi-empirical, and physical models [23]. Physical models express fire spread as a physical process utilizing thermodynamics and combustion chemistry, making them quite computationally heavy with limited practical applications. Empirical models offer a macroscopic understanding of the phenomenon, derived from statistical methods, regression, and data calibration. Semi-empirical models combine both approaches, and hence are more appropriate for real-world applications, offering sufficient accuracy under reasonable computation time. Each approach balances accuracy, computational load, and data acquisition needs in a different way, so it is not surprising that the literature presents a wide range of models within each category. This stems from the fact that no single model is universally suitable for all applications [24]. Varying assumptions and inherent limitations compel researchers either to develop new models or to adapt existing ones to best fit the constraints of a specific case.

However, a wildfire model cannot be applied to real-world scenarios independently; it requires integration into a computational framework capable of executing the model’s principles and translating them into actionable outputs. A computational model is required to couple the wildfire spread model with time and spatial information. The objective is to provide a numerical representation of a wildfire characteristic (like fire line intensity or rate of spread) at a given spatial location and time step. Some of the most applied computational models for wildfire simulation are cellular automata [25, 26], wavelets [27], Markov processes [28], level-set methods [29] and agent-based methods [30, 31].

2.2 Workflow for Fire Dynamics Modeling

In order to generate meaningful outputs, a computational model must undertake a series of interdependent steps, typically executed implicitly within its internal processes. These steps transform raw input data into structured representations, apply governing rules or equations, and iteratively update the system’s state over time. Together, they form a cohesive computational workflow that enables the model to simulate fire spread, with both spatial and temporal consistency.

1. **Space digitization and time discretization:** The simulation domain is digitized by dividing time and space into computationally manageable intervals. Usually grids, meshes (discrete space) or vector representations (continuous space) are used, with topological characteristics assigned to them. At each computational step, the state of the digitized space is subject to evaluation according to some transitioning rules.
2. **Apply propagation/spread mechanism:** Use of the fire model’s formulas to compute ignition probability, ignition time, fire front velocity or any other characteristic at each location and time interval.
3. **Handle topological structures:** The computational model should deal with the geometry of the involved items, such as distinguishing fuel barriers and heterogeneous or ambiguous data, geometry simplification of complex

shapes, handling of discontinuities or missing data, merging burned areas during simulation, etc.

4. **Integrate with external sources:** Almost any computational model relies on external sources for weather data, fuel composition, and terrain elevation. The data acquisition mechanisms may range from web services to crowd-sourced data.
5. **Display output:** The output of the model should provide a digital representation of the simulated area’s state, corresponding to a specified point in time. Additional qualitative indicators derived from the model, such as the fire front perimeter and the rate of spread, constitute valuable decision-support metrics.

2.3 A Taxonomy of Wildfire Propagation Modeling Approaches

Modeling wildfire propagation encompasses a diverse and multifaceted range of approaches that can be systematically classified according to their underlying assumptions, computational representations, and data requirements. To facilitate a clearer understanding of the current literature in the field, we present the following taxonomy (Figure 2.1) of the most commonly employed methods. This taxonomy exhibits not only the conceptual diversity of wildfire models, but also the computational and data-oriented choices that shape their application in simulation and decision-making support. It should be noted that certain methods may simultaneously fall into more than one category, reflecting the inherent overlaps in methodological approaches. This classification represents one possible framework, rather than an exhaustive or definitive taxonomy scheme.



Figure 2.1: A comprehensive taxonomy of wildfire spread and simulation methods. Adapted from [15].

At the first level of the taxonomy, we identify three fundamental questions that a researcher must answer to select the appropriate simulation model for the case: specifically, what the underlying conceptual model will be, which computational model is most appropriate, and what kind of data are available to support the simulation.

1. **Conceptual model:** Depending on the required level of accuracy and the feasibility of representing the wildfire environment (e.g., spatial features, fuel characteristics), we can select a physical, empirical, or semi-empirical model. In the case of wildfires, the most applicable are semi-empirical models, since vast, heterogeneous areas must be modeled in limited time.
2. **Computational model:** Typically, in wildfires raging for kilometers, we do not require extreme spatial accuracy, so using a discrete solution space leads to reduced computational time while maintaining acceptable accuracy. In this subdivision we find the most applied techniques, such as cellular automata, agent-based, and Markov-based approaches. Hybrid methods combining vector-based approaches with cellular automata techniques leading to continuous-like solution spaces do exist, as we will see later on.
3. **Data:** Data availability, resolution, and the time elapsed between data acquisition and model execution (the age of information) are among the most crucial factors for receiving an accurate output from any simulation engine. Data may come from online APIs or they may have been gathered from past events (mapping of a burned area); they may even be acquired by crowd-sourcing methods [32].

2.4 Raster-Based / Discrete Solution Space Wildfire Propagation Models

Raster-based wildfire propagation models represent the simulated landscape as a discretized, usually square, grid, where each cell corresponds to a spatial

unit. A key assumption is that each cell is spatially uniform, such that attributes including fuel type, moisture content, and topography are considered constant throughout the cell. Fire spread is modeled as a local interaction process, where the state of each cell evolves over time steps according to transition rules and environmental conditions. These models often employ Cellular Automata, Markov-based, or agent-based approaches that model ignition and propagation dynamics.

Their strength lies in their computational efficiency and their easy to implement nature. Hence, they can be compatible with a wide range of geospatial datasets, such as satellite imagery and digital elevation models, which makes them particularly suitable for large-scale simulations where analytical or physics-based models become computationally intractable. However, their accuracy is inherently influenced by the grid’s resolution and the simplifications imposed by their discrete spatial representation, which may limit their ability to capture highly heterogeneous landscapes.

2.5 Cellular Automata

Cellular Automata (CA) are a class of discrete mathematical models highly effective in simulating complex, dynamic systems. They are usually characterized by the discretization of both space and time, with applications in 1, 2 or higher space dimensions. CA simulate a system’s evolution through the continuous application of localized (stochastic or deterministic) interaction rules (transitions). This paradigm where macroscopic system behavior is predicted via micro scale, neighborhood level interactions allows for the computationally efficient modeling of highly complex environments. Since their foundational introduction by John von Neumann in the 1950s [33], CA have been extensively adapted and integrated into multidisciplinary research, providing critical predictive capabilities in fields such as physics, biology [34], computer science, and environmental studies [35].

In applied spatial modeling, the initial construction of a cellular automaton requires establishing a grid that accurately maps the physical domain of the target phenomenon (eg., landscape). This grid acts as a discretized spatial repre-

sensation, typically consisting of homogeneous two-dimensional square or hexagonal cells, each representing a homogeneous environmental unit (Figure 2.2). The simulation engine relies on a dual discretization approach: the environment is partitioned into individual cells, and the simulation advances in synchronized, discrete time increments. The propagation of the modeled phenomenon is calculated deterministically or stochastically at each iteration. During these temporal updates, every cell recalculates its state based on a strict set of localized transition rules applied to its defined neighborhood, allowing macro scale environmental dynamics to emerge from micro scale spatial interactions. The state of a cell is associated with a state variable that captures its condition at a particular time step; this state evolves according to a given set of transition rules.

2.5.1 A Stochastic CA Definition

In the version presented in this dissertation, we adopt the approach of a two-dimensional CA wildfire simulator presented in [36, 37]. We selected this approach because it has been validated against real wildfire case studies and because it is comprehensive, incorporating key input parameters such as fuel characteristics, terrain data, weather conditions, and firefighting strategies and tactics. Given the difficulty of obtaining accurate and reliable data on aerial firefighting operations, we omit this component of the frameworks presented in [36, 37], to avoid introducing an additional source of uncertainty, including spotting behavior.

The model is defined as a two dimensional Cellular Automaton on a finite square lattice K as follows (Equation 2.1):

$$CA = \langle K, S, N, \delta, U \rangle \quad (2.1)$$

where:

- K is a finite square lattice of cells ($\mathbb{Z}^2$)
- S is a finite set of discrete states a cell may be assigned to
- N is the neighborhood scheme, defining which cells influence a given cell

- δ is a stochastic local transition function: $\delta : S^{|N|} \rightarrow S$
- U is a synchronous cell update policy: $s_i(t+1) = \delta(s_j(t) : j \in N(i))$ where $s_i(t) \in S$ is the state of the i^{th} cell at the simulation time step t

The spatial solution space of the simulation K is defined as a two dimensional square lattice of size of size $l_{\text{cell}} \times l_{\text{cell}}$ composed of a finite number of uniform cells. Each cell represents the minimum spatial unit (for a given resolution) in which the CA state is defined and evolves over time. Since a Moore neighborhood configuration is selected, in which each cell interacts with its eight adjacent cells (four orthogonal and four diagonal), the square lattice structure limits the local interactions between neighboring cells and allows the fire to advance only in eight possible ways (Figure 2.2).

The set $S = \{S_1, S_2, S_3, S_4\}$ consists of the following four discrete states that describe the status of a simulation cell in terms of fire spread at a given time:

- S_1 : **No fuel.** That state means that the cell contains noncombustible materials like rocks or water bodies. Transitions from that state are prohibited in future iterations.
- S_2 : **Combustible fuel.** The cell contains combustible fuels that will be considered for ignition at the following iterations.
- S_3 : **Burning.** The cell is in the burning state for the duration of the current iteration and cannot return to that state in the future (re-ignite).
- S_4 : **Burned.** The cell contents have been completely consumed by the fire and further transitions from that state are prohibited.

The neighborhood scheme N defines the mechanism that selects which cells will influence the state of a given cell at the next iteration according to their previous states. The Moore neighborhood is generally considered more suitable for modeling phenomena such as fire spread, where interactions are not constrained to a single direction and the spatial extent of the area to be modeled is considerable.

Moreover, approaches exist in which the neighborhood size is extended, such as the dynamic method proposed by Freire [38].

The local transition function δ specifies the rule according to which cell states change between successive iterations. It determines the state of a cell at the simulation time step $t + 1$ based on its state at time t and the state of its neighborhood. To capture realistic fire behavior, δ is formulated to incorporate probabilistic effects such as wind direction, fuel moisture, and other environmental factors, allowing the likelihood of cell ignition to vary according to local conditions. Consequently, the state $s_{ij}(t + 1)$ of the cell in the i^{th} row and j^{th} column of the lattice at the simulation time $t + 1$ is given in Equation 2.2:

$$s_{ij}(t + 1) = \begin{cases} S_1. & \text{if } s_{ij}(t) = S_1. \\ S_3 \text{ with probability } P_{burn}. & \text{if } s_{ij}(t) = S_2 \text{ AND } s_{kl}(t) \in N(s_{ij}) = S_3. \\ S_4. & \text{if } s_{ij}(t) = S_3. \\ S_4. & \text{if } s_{ij}(t) = S_4. \\ S_2. & \text{in any other case.} \end{cases} \quad (2.2)$$

These local rules dictate the set of assumptions underlying a typical CA model for wildfire spread:

1. A cell containing no fuel (S_1) remains in that state during the simulation.
2. A cell can be ignited (S_3) at the next iteration with probability P_{burn} (Figure 2.3) if one of its neighboring cells is burning (S_3) and the cell itself contains combustible fuel (S_2). Otherwise, it remains in state S_2 .
3. A cell that is burning (S_3) will be completely burned out (S_4) at the next iteration.
4. A burned cell (S_4) has been completely consumed by the fire and cannot be reignited.
5. Any changes in cell states that occur outside the defined discrete time steps of the simulation cannot be captured.

The probability P_{burn} couples landscape characteristics with weather conditions according to Equation 2.3:

$$P_{burn} = P_0(1 + P_{veg})(1 + P_{den})P_wP_sP_m, \quad (2.3)$$

where P_0 is a base probability expressing the likelihood of a cell ignited by an adjacent one, under the influence of zero wind and no elevation difference [36, 37]. P_{veg} and P_{den} denote probability coefficients for vegetation type and density, respectively, with values derived from fuel type parameterized lookup tables. Typically they are constructed using information obtained from remote sensing-based fuel characterization techniques. P_w is a probability coefficient associated with the influence of wind to fire spread and can be calculated according to Equation 2.4:

$$P_w = e^{[V(c_1 + c_2(\cos(\theta) - 1))]}, \quad (2.4)$$

where c_1, c_2 are user-specified parameters, V is the wind velocity and θ represents the angle between the prevailing wind direction and the direction of potential fire spread.

P_s is a probability coefficient that models the effect of terrain slope on fire spread. Fire propagates more rapidly when moving upslope and slows down when moving downslope. This occurs because, on an upslope, flames are tilted closer to the unburned fuel, enhancing heat transfer through radiation and convection and promoting faster preheating and ignition [39]. P_s can be calculated according to Equation 2.5:

$$P_s = e^{\alpha_s \theta_s}, \quad \theta_s = \tan^{-1} \left(\frac{E_1 - E_2}{d(E_1, E_2)} \right), \quad (2.5)$$

The coefficient α_s is a user-specified parameter associated with the terrain influence on fire spread, E_1 and E_2 are the elevations of the cells considered for fire propagation, and $d(E_1, E_2)$ is the Euclidean distance of cells E_1, E_2 .

The probability coefficient P_m accounts for the *Fuel Moisture Content* (FMC) in the fire spread process, capturing its effect on ignition probability. Higher moisture levels reduce flammability and thus slow or inhibit fire propagation. In [40], the relationship between *Rate of Spread* (ROS) and FMC is established using

nonlinear interpolation of sample fire data, capturing the negative exponential dependence of ROS on FMC. Hence, it can be modeled according to Equation 2.6, where a, b are user-specified tuning parameters:

$$P_m = \alpha e^{-b \cdot FMC}, \quad (2.6)$$

The FMC is rarely measured directly in practice, as it requires considerable time and a large number of sampling points across the study area, which is unrealistic during a wildfire event. Direct measurements are typically conducted only to validate FMC model estimates in real-world scenarios. In the context of wildfire simulation, FMC is usually obtained indirectly from weather and environmental data. Later in the dissertation, where an implementation is required, we compute the *Equilibrium Moisture Content* (EMC), a widely used and practical metric for estimating fuel moisture.

2.5.2 CA with Continuous State

In an infinite, or continuous-state CA, the possible states are not restricted to a finite set (refer to Section 2.5.1) and can vary continuously; they may be represented by real values (e.g., the percentage of consumed forestry fuels in a cell, the average fire intensity inside a cell, etc.). This allows for a more detailed and nuanced representation of the system’s behavior, though it comes with increased complexity and additional transition rules. This approach allows a cell to be in a burning state for multiple simulation steps, increasing the probability for neighboring cells to catch fire.

Finite-state CA are simpler but less accurate, since continuous-state systems can provide a much more realistic simulation of wildfire dynamics by accounting for intermediate states (e.g., varying levels of fire intensity). In finite-state CA, the assumption that an area is fully burned in a single simulation step with fire propagation occurring only in the subsequent step is unrealistic. Forest fuels are composed of materials that may burn over extended periods of time, and the rate of fire spread varies spatially due to multiple factors such as fuel composition, topography and weather. This inherent over-simplification where the fire front

instantaneously advances in uniform discrete steps inevitably introduces inaccuracies, as it fails to capture the continuous and variable nature of fire propagation in real-world scenarios. Furthermore, such an approach makes the choice of the simulation time step extremely critical. Choosing a large simulation time step will make the fire front propagate more rapidly than it would in reality. The same applies to cell size selection. A huge cell size will lead to a behavior where the fire front will be propagated to more areas than it would if a more appropriate cell size had been selected, since some spatial detail of the area inside the cell is omitted. On the other hand, smaller time steps and smaller cells make the model more computationally expensive.

2.6 The State of the Art in CA Models for Wild-fire Simulation

2.6.1 Baseline Multivariable CA

Alexandridis et al. [36] introduced a baseline CA model, which was later extended to improve coupling with weather conditions through FMC [37] and to incorporate fire suppression mechanisms via a stochastic aerial drop module. This dissertation adopts that CA formulation as its baseline, as detailed in Section 2.5.1. It consists of four discrete states and four simple local transition rules applied to each cell based on its eight adjacent cells (Moore neighborhood). Additionally, the method addresses the phenomenon of spotting fires—which occurs when flaming boughs, pinecones, or other burning materials are carried by the wind and ignite unburned or partially burned areas ahead of the main fire front—by introducing a fifth rule that probabilistically determines the direction and distance d_p that such materials can travel. As a result, a separate probability p_c is used to determine whether the falling material will ignite the cell on which it lands.

The authors validated their model by applying it to historical wildfire events in Greece, specifically the fires in Spetses and Parnitha, and comparing the model’s predicted outputs with the observed fire behavior. Based on aerial maps, three veg-

etation density classes were identified, as well as three vegetation types on Spetses and twelve on Parnitha. The selection of specific coefficient values appears to rely primarily on empirical knowledge and observations presented in the literature. More critically, there is no evidence of a unified framework capable of dynamically mapping different fuel types to numerical weighting factors, which limits the general applicability of the method.

2.6.2 Dynamic Neighborhood Configuration

The finite-state CA model proposed in [38] shares the formulation of Section 2.6.1 but introduces wind-driven neighborhood selection. The modification amplifies the effect of wind on wildfire spread by allowing distant cells, rather than only adjacent ones, to ignite on certain wind level thresholds.

Vegetation type classification for this model was conducted using datasets such as the *Corine Land Cover* (CLC) and the *Moderate Resolution Imaging Spectroradiometer* (MODIS) annual maximum green vegetation fraction, with the remaining coefficients derived from [36]. The CLC dataset is a standardized European spatial database, derived from satellite imagery that categorizes land surface characteristics into 44 distinct classes. It delivers a mapping baseline required for large scale environmental and spatial analysis [41]. The method was evaluated against a past fire in Portugal, where the authors observed a delayed behavior in fire spread when running a typical CA implementation during the exploding stage of the actual wildfire. In contrast, the modified implementation reproduced this stage more accurately, although it exhibited unrealistically rapid spread during the initial phase of the simulation. This discrepancy may be due to a possible overfitting of the wind-threshold parameters that govern the extent of the neighborhood cells considered for ignition, since the researchers had access to meteorological data from the real wildfire event.

2.6.3 Accessible Data-Driven CA

The implementation proposed in [42] addresses the challenge of manually assigning fuel classes to vegetation cells, which typically involves examining large remote-sensing datasets. Unlike other implementations that utilize more complex and less ready-to-use data sources (e.g., raster files that must be downloaded and then cropped to the simulation area), they directly query the widely accessible Google Maps API. Even though Google Maps was not designed for remote sensing applications, particularly vegetation classification, it offers a fast and effective means of analysis and serves as a valuable baseline metric for such applications. In contrast to satellite photoproducts specifically designed for remote sensing applications, Google tiles contain only RGB channels, which limits their applicability for calculating indices such as the *Normalized Difference Vegetation Index* (NDVI) [43] and the *Soil-Adjusted Vegetation Index* (SAVI) [44] which are commonly used to assess vegetation type, condition, and health. However, similar metrics can be estimated using visible RGB wavelengths like the *Triangular Greenness Index* (TGI) [45] and the *Visible Atmospherically Resistant Index* (VARI) [46]. While the proposed method can operate in scenarios with limited user-provided geospatial data, its performance is inherently constrained by the low spatial resolution and outdated imagery available from Google Maps, as well as by the lack of a clearly defined methodology for mapping vegetation types to the corresponding CA vegetation classes.

2.6.4 Operation-Ready Implementation

The PROPAGATOR model [47], considered the most mature in terms of operational readiness, has been refined through eleven years of implementation and deployment in operational environment by civil protection authorities. The latest release of PROPAGATOR has been implemented as a web-based platform that enables the integration of user-specified fire suppression strategies (classified as water-based operations and mechanical or structural operations) and data acquisition techniques for obtaining terrain elevation and fuel cover characteristics. The system also utilizes meteorological parameters for the simulation of more realistic

fuel behavior processes, including an implementation for estimating the moisture content in fine dead fuels, that is, small, highly combustible dead vegetation (e.g., needles, twigs, and leaf litter) that responds rapidly to environmental conditions.

An innovative aspect of the method is the incorporation of an adjacency matrix that quantifies the likelihood of fire spread among six vegetation classes. This matrix defines a nominal spread probability p_n as a function of the vegetation type of the burning cell and that of its adjacent neighboring cell. These classes (Broadleaves, Shrubs, Grassland, Fire-prone conifers, Agro-forestry areas and Not-fire-prone forest) provide an abstract yet easily interpretable classification of forestry fuel types, with their corresponding nominal spread probabilities obtained via calibration against observed fire behavior data.

2.6.5 Continuous-State CA Implementation

Karafyllidis [48] introduced a novel framework to overcome the limitations of traditional methods, which rely on a finite set of constrained states to represent the condition of a burning cell across continuous burning stages, from partially burned to completely consumed by fire. At time step t , the state $S_{i,j}^t$ of the cell with coordinates (i, j) represents the burned area within the cell as a ratio of the total cell area, and hence its value lies in $[0, 1]$, where 1 means that it is completely burned. In contrast with the other methods, which decide probabilistically which cell to ignite at the next simulation step, the method will ignite (at least partially) any cell that either shares a side or is diagonally adjacent to a burning cell. The state $S_{i,j}^{t+1}$ of the cell (i, j) at time step $t + 1$ is given by Equation 2.7, where, for simplicity, wind and elevation factors are omitted:

$$S_{i,j}^{t+1} = S_{i,j}^t + (S_{i-1,j}^t + S_{i,j-1}^t + S_{i+1,j}^t + S_{i,j+1}^t) + 0.83(S_{i-1,j-1}^t + S_{i-1,j+1}^t + S_{i+1,j-1}^t + S_{i+1,j+1}^t) \quad (2.7)$$

This equation implies that the state of a cell will become the sum of the states of its adjacent ones and the sum of the states of its diagonally adjacent cells multiplied by the term 0.83. Obviously, at any simulation step, the value of $S_{i,j}^{t+1}$

varies from 0 to 1, but may also exceed 1 and therefore must be appropriately constrained to this value. The duration of the simulation time step t_a is derived as the ratio of cell length to the ROS. Due to the unique manner in which the method utilizes the ROS, neither tuning coefficients nor predefined vegetation-type classes are required. Instead, the ROS itself governs the fuel characterization and fire propagation dynamics. It is assumed that users employ an appropriate model to estimate the ROS [39] prior to the simulation.

2.6.6 Variable Simulation Time Step CA

The improved CA method introduced by Sun et al. [49] utilizes the *Daily Burning Index* (DBI) [50] to quantify the influence of current environmental conditions on wildfire behavior. The equivalent of ROS denoted as R (forest fire spread speed) is obtained by Equation 2.8, where the parameter R_0 (initial speed of the forest fire) is derived from the DBI index (Equation 2.9). K_ϕ , K_θ and K_s model the influence of terrain, wind, and fuel respectively, while K_r is a time correction coefficient. T is the maximum air temperature, V is the noon mean wind scale, H is the lowest daily relative humidity and a, b, c, D are constant coefficients.

$$R = R_0 K_\phi K_\theta K_s K_r, \quad (2.8)$$

$$R_0 = aT + bV + c(100 - H) - D, \quad (2.9)$$

The simulation time step Δt is calculated dynamically at each time step according to Equation 2.10:

$$\Delta t = \frac{\Delta t_0}{e^{a_{i,j}^t}}, \quad (2.10)$$

where Δt_0 is the time step selected during the initialization of the method and $a_{i,j}^t$ is the speed change rate index, which is calculated by subtracting the expected speed of spread at the next step from the speed in the current one. The cell state at the next time step is determined by its current state, spreading speed of neighboring cells, cell length, and dynamic time step Δt .

2.6.7 Synergistic Vector-Based and CA Approach

The approach of Zhuang et al. [51] also utilizes Wang’s DBI index [50] to derive the fire spread index and a similar dynamic time step selection. A key novelty of the proposed approach is that fire propagation is not constrained by cell boundaries and may extend beyond the boundaries of the originating cell (Figure 2.4), depending on the ROS and cell dimensions L_{cell} .

The propagation dynamics are governed not only by each cell’s state (which is used here in a different way), but also by an additional variable, the “critical condition judgment” C_f . Both variables are continuous, and they are interpreted as follows. The C_f value is used to characterize the state of a cell based on three conditions: $C_f = 0$ indicates an unburned cell, $0 < C_f < 1$ is a cell in burning state, and values above 1 signify that the cell is completely burned. On the other hand, the variable “state” $S_{(i,j)}^t$ describes the state of the adjacent cell (according to the direction of spread) of the cell with coordinates (i, j) . A value of 0 or 1 indicates that the adjacent cell is unburned, $1 < S_{(i,j)}^t < 2$ means that the current cell caused its adjacent cell to start burning, and values above 2 means that the adjacent cell is completely burned. The condition judgment factor is calculated according to Equation 2.11:

$$C_f = \frac{ROS_{i,j}^t \Delta_t}{L_{\text{cell}}} \quad (2.11)$$

2.6.8 Enhancing Spatial CA Accuracy

Ghisu et al. [52] highlight that raster-based approaches, including CA, are prone to spatial distortions, often producing triangular-like artifacts rather than smooth curvature lines, and thus failing to accurately represent a realistic wildfire perimeter. Extensive research has been conducted to characterize and mathematically represent [53] the geometry of wildfire perimeters. Although one of the earliest and simplest formulations—the elliptical model [54]—remains the standard adopted by most researchers, alternative shapes, such as the double ellipse and the ovoid, have also been proposed, particularly for more specialized scenarios.

The method consists of two main components: a conventional CA with four states and an optimization procedure which optimizes six CA coefficients (e.g., rate of spread, wind impact), intended to mitigate the shape distortions arising from the discrete nature of the CA. The time step is held constant and chosen such that the fire front cannot extend beyond the boundaries of a single cell. The optimization module is based on Tabu Search [55] with the problem formulated as a multi-objective optimization problem based on three metrics. This optimization stage yields a set of correction coefficients, which keep the estimated wildfire perimeter close to the theoretical one. Although this is a promising method, combining the spatial-resolution advantages of vector-based approaches for theoretical perimeter construction with the computational efficiency of raster-based methods, the authors limited their benchmark evaluation to simulations generated using the FARSITE software [56, 57], rather than against real-world test cases. In addition, since the model relies on a theoretically prescribed fire shape, more complex phenomena such as backfire spread are estimated using approximations rather than being represented by dedicated physical models.

Trunfio et al. [58] also contributed to improving the spatial accuracy of the simulated burned area by adopting a *Macroscopic Cellular Automata* (MCA) [59] approach. Unlike typical CA-based models, MCA account for external effects and are not limited to purely local rules in order to capture macroscopic effects. In this implementation, these effects are modeled by a set of global functions that are updated before the transition function. Every cell in the grid contains variables that dictate both the environment and the fire’s behavior. The key properties include environmental data (altitude z , fuel model μ , and a combustion state σ) and characteristics of an ellipse. In order to track the expanding fire, each ignited cell is associated with a local fire-spread ellipse γ , which is defined by its rear focus f (the fire’s source point), the accumulated forward spread ρ (the length of the ellipse along its major axis), the spread direction r , and the ellipse’s eccentricity ϵ . During each time step, the cell’s transition function expands its local fire ellipse γ by increasing the accumulated forward spread ρ by the product of the spread rate and the time step ($r\Delta t$). Then it evaluates outward propagation by computing the

intersection set between the expanding ellipse and the geometric boundary χ_i of an unignited neighboring cell c_i . If the ellipse invades the neighboring cell, ignition is triggered. To model the new fire source, the algorithm generates a new ellipse γ_i such that it is tangent to the parent ellipse γ .

2.6.9 Lightweight Vegetation-Aware CA Model

In contrast to most other advances, which make CA models more complex, often requiring extensive coefficient fine-tuning and detailed vegetation-class and vegetation density classification, Weinhouse [60] proposes a lightweight and easily implementable approach that relies primarily on the NDVI index, which can be readily computed from commonly available satellite photoproducts, wind conditions and terrain slope obtainable from a *Digital Elevation Model* (DEM). A p_{burn} probability term (Equation 2.12) is computed similar to [36, 37] as follows:

$$p_{burn} = p_0 f_{NDVI} f_{slope} f_{wind} \quad (2.12)$$

The term f_{NDVI} (vegetation modifier) is determined by the NDVI value of the corresponding cell according to predefined threshold values. Its purpose is to replace the ambiguous vegetation density probabilities used in previous studies by providing a more explicit and physically interpretable representation.

The primary strengths of the method lie in its simplicity, minimal input requirements, and computational efficiency. However, tuning the p_0 probability coefficient still requires access to near-real-time data, such as current weather conditions and updated satellite photomaps.

2.6.10 CA for Wildland–Urban Interface Areas

Traditional CA models fail to accurately model urban environments, because houses and human-made infrastructure do not burn like continuous biomass. WUI-CA models solve this by introducing distinct cell categories (e.g., “Forest”, “Structure”, “Road”) and category dependent transition rules.

The WUI-CA model of Jiang et al. [61] integrates forest fire dynamics and

structural fire dynamics within the same CA framework, by labeling each cell as either “forest areas” or “urban areas”, each associated with different states. Cells in forest areas can evolve in five states: “Not burned,” “Catching burning,” “Full-developing,” “Turn to extinguish” and “Extinguished.” The permitted states of cells in urban areas are extended to six: “Original State,” “Ignition,” “Slow Development,” “Full Development,” “Flashover” and “Extinguished.” Additionally, fire-resistant or non-combustible elements, such as roads and wastelands, can be modeled. These elements are represented as barriers that violate the continuity of the flammable materials. For cells in “forest areas” a rate of spread R and state $C_{i,j}^{t+1}$ is calculated in a similar way as in [49]. For cells in “urban areas,” the model switches to a different semi-empirical transition rule based on thermal radiation and contact. Instead of the rate of spread R , the model adopts the concept of ignition probability I_n based on thermal transfer principles. A building ignites with probability I_n if the total heat received exceeds a threshold. The probability I_n of igniting the unburned n^{th} building is derived by the influence of M nearby burning buildings according to Equation 2.13 (in simplified form).

$$I_n = 1 - \prod_{i=0}^M (1 - P_{M_{i,n}}) \quad (2.13)$$

The term $P_{m,n}$ refers to the probability of igniting the n^{th} building due to the heat received from the m^{th} building within an influence range. The calculation of $P_{m,n}$ is derived from [62] (Equation 2.14) where P_{T_n} is a probability coefficient (from a lookup table) related with building’s materials, P_W is a weather related probability coefficient, P_{S_n} is a coefficient dependent on the state of the building and $P_{A_{m,n}}$ represents the ratio of the area within the influence range that is occupied by the n^{th} building.

$$P_{m,n} = P_{T_n} P_w P_{S_n} P_{A_{m,n}} \quad (2.14)$$

Fire propagation from forest to urban areas and vice versa is governed by heat flux emitted from burning cells, represented by influence ellipses whose size depends on environmental conditions. A forest fire can ignite nearby buildings if

they fall within its thermal influence range, while burning buildings can similarly ignite surrounding forest fuels, enabling bidirectional fire spread between wildland and urban areas despite their different internal CA representations.

Ju et al. [13] presented a similar WUI environment modeling approach coupled with a CA simulation component, focusing primarily on semantic representation by introducing ontologies and explicitly defining properties for entities involved in the fire process. In a different context, [21, 63, 64] also explored the mapping of fire components into ontologies, highlighting the importance of formalizing fire-related domain knowledge.

2.6.11 CA Comparative Matrix

A comparative table (Table 2.1) of the previously examined CA models is presented in this subsection, enabling a systematic comparison of their key characteristics and differences. For a detailed description of the methods and their limitations, the reader is referred to our original publication in [15].

Table 2.1: A comparative overview of State of the Art Cellular Automata models. Adapted from [15].

Method	Neighborhood	Fire Suppression	Spotting	States	Time Step	Complexity	Cell Size	Test Case
[36]	Moore	No	Yes	Finite	Static	Low	5m	Past fire
[37]	Moore	Yes	Yes	Finite	Static	Medium	5m	Past fire
[38]	Dynamic	No	Yes	Finite	Static	Medium	100m	Past fire
[42]	Moore	No	No	Finite	Static	Low	-	Synthetic
[47]	Moore	Yes	No	Finite	Dynamic	Medium	20m	Past fire
[48]	Moore	No	No	Continuous	Static	Medium	-	Synthetic
[49]	Moore	No	No	Continuous	Dynamic	High	30m	Past fire
[51]	Moore	No	No	Continuous	Dynamic	High	30,50,100m	Past fire
[52]	Moore	No	No	Finite	Dynamic	High	20m	Synthetic
[60]	Moore	No	No	Finite	Static	Low	10m	Past fire
[58]	Moore	No	No	Finite	Dynamic	High	20m	Synthetic
[61]	Moore	No	No	Finite, 2 classes	Dynamic	High	-	Past fire

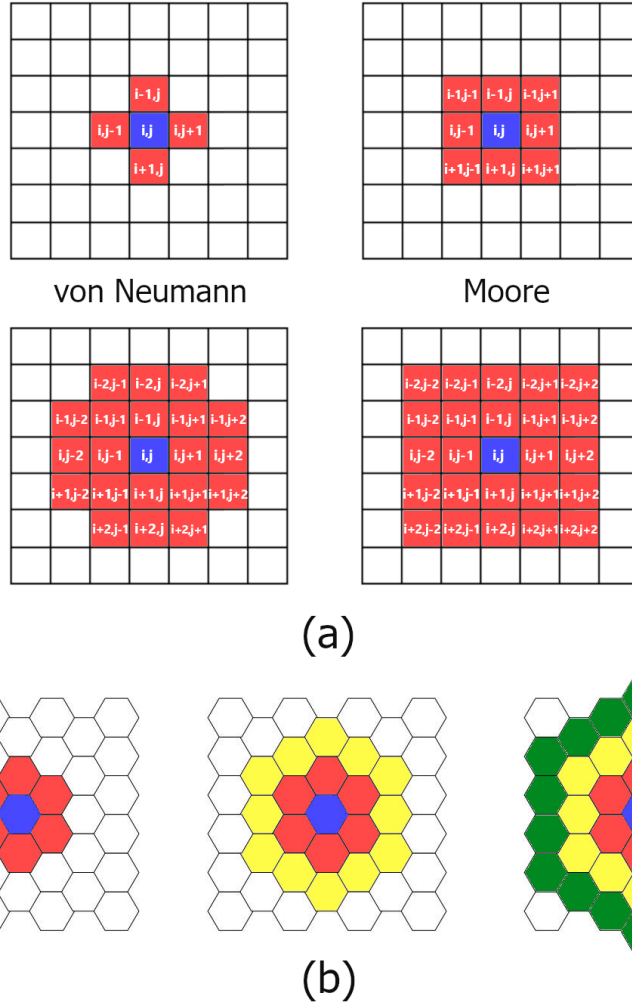


Figure 2.2: A two dimensional grid with (a) rectangular and (b) hexagonal cells. Possible neighborhood configurations for the current (in blue color) cell are shown, for different types of cell shape and arrangement. Adapted from [15].

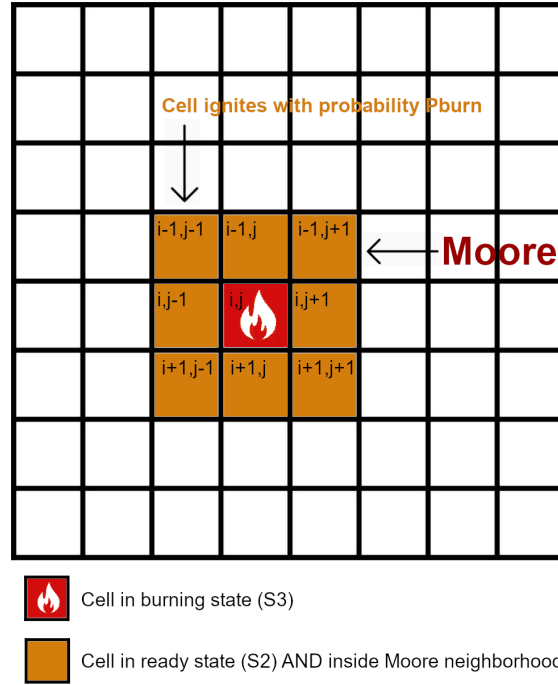


Figure 2.3: Illustration of P_{burn} within a Moore neighborhood configuration. Adapted from [16].

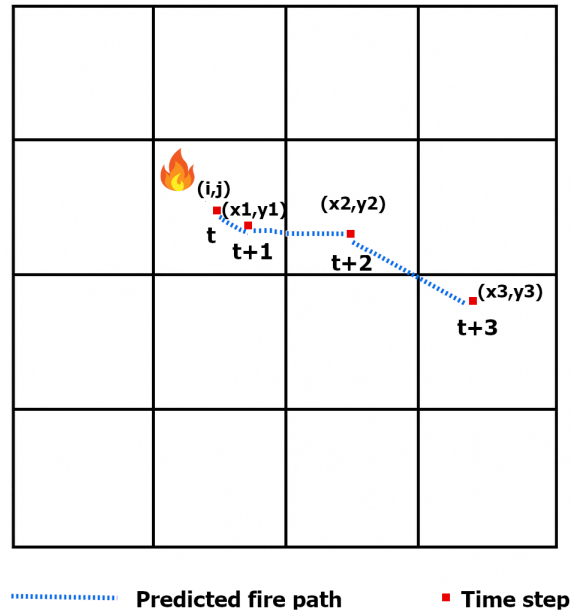


Figure 2.4: Initial ignition point (indicated by the fire symbol) and subsequent fire propagation across consecutive simulation time steps. The fire point may or may not lie inside the same cell during the next time step. Adapted from [15].

Chapter 3

Enhancing Wildfire Simulation Through Volunteered Geographic Information

3.1 Crowdsourced Wildfire Spread and Status

Wildfires, particularly those occurring in the *Wildland–Urban Interface* (WUI), pose growing threats to civilian populations by causing loss of life, property damage, and disruptions to communication infrastructure. In these moments, every second matters, and so does every observation of the evolving phenomenon. Today, nearly every person carries a powerful sensing device in their pocket, capable of capturing and sharing what they see and feel in almost real time, i.e., their smartphone. This convergence of risk and capability motivates a new direction in wildfire research and response: leveraging crowdsourced, user-submitted data to track fire propagation and evolving conditions [65]. By transforming in-field stakeholders who are familiar with wildfire dynamics into active contributors of geospatial, fire-related data, such approaches aim to enhance situational awareness and help communities navigate danger more safely.

Building on this motivation, a growing body of research has begun to explore how such voluntary, user-driven reporting can be systematically harnessed

for wildfire monitoring and simulation. Prior work spans a range of approaches, from mobile applications that collect geotagged photos and incident reports, to social media mining techniques that extract real-time signals of fire activity, to hybrid systems that fuse crowdsourced observations with satellite imagery and fire spread models. The proposed approaches differ in their assumptions, data quality controls, and integration strategies, yet they share a common goal: to transform dispersed, user-generated inputs into reliable, actionable estimates of wildfire behavior and propagation. The remainder of this section provides an overview of various research approaches in this area.

3.2 Crowdsourcing vs. Remote Sensing

A fundamental requirement of the proposed framework is the ability to perform *progressive wildfire simulation*, whereby the simulation is continuously updated as new information regarding the evolving fire environment becomes available. Unlike conventional wildfire modelling approaches, which are typically initialized using static datasets, the proposed framework is designed to assimilate observations throughout the duration of the incident reported by firefighting personnel. This enables the simulation to incorporate changes in the fire perimeter, newly burned areas, local fuel conditions, and weather variations, thereby maintaining an up-to-date representation of the evolving wildfire.

Relying exclusively on remote sensing products or standard weather APIs is insufficient to meet these requirements. Although satellite and aerial observations constitute the standard source of geospatial wildfire information, they are generally characterized by acquisition delays, limited revisit frequencies, and spatial resolutions that may be insufficient for supporting operational decision-making during an active wildfire. Furthermore, several fire related variables required by the simulation, such as local fuel characteristics or rapidly evolving fire conditions, cannot always be inferred reliably from remotely sensed imagery, particularly under adverse atmospheric conditions or dense smoke [66, 67]. Rather than depending on periodically updated remote sensing products, the framework leverages observa-

tions collected directly by firefighters operating in the field. As the fire evolves, firefighters can continuously report newly burned areas, fire behaviour, fuel characteristics, and other locally observed conditions, allowing the simulation to be progressively updated with information that reflects the current state of the incident. Since data are collected at the point of observation, their spatial granularity is not constrained by the resolution of a sensing platform but can instead match the level of detail required by the simulation.

While crowdsourced observations are inherently subject to uncertainty, inconsistencies, and varying levels of completeness [68], they offer a critical advantage for operational wildfire simulation: *timeliness*. For near-real-time applications, the availability of current field observations is often more valuable than highly accurate but delayed datasets. Consequently, the proposed framework prioritizes the continuous availability of operational information over the absolute accuracy of individual observations and is designed to accommodate the uncertainty associated with crowdsourced data.

Unlike conventional *Volunteered Geographic Information* (VGI) systems, which are typically open to the general public, the proposed approach restricts data submission to firefighters who possess operational experience and domain-specific knowledge of wildfire behaviour, fuel characteristics, and fire suppression activities. This assumption substantially reduces the likelihood of erroneous or malicious submissions while ensuring that observations originate from individuals capable of reliably identifying fire related phenomena that may not be readily discernible by non-experts.

Furthermore, the reliability of individual contributors can be explicitly incorporated into the data integration process. Rather than assuming that all observations are equally trustworthy or relying solely on majority voting, the VGI literature has proposed numerous approaches for estimating the credibility of individual contributors and deriving a distributed consensus from multiple, potentially conflicting observations [69]. Such methods typically assign confidence scores to contributors based on their historical performance, employ statistical truth-discovery models [70, 71] that weigh observations according to source reliability, or itera-

tively estimate both contributor credibility and the most probable state of the observed phenomenon. Consequently, observations submitted by highly reliable contributors exert a greater influence on the final integrated dataset than those originating from less reliable sources. The detailed investigation of contributor credibility assessment and truth-discovery algorithms lies beyond the scope of this work. However, Section 5.3.4 discusses how the proposed probabilistic ACO-based algorithm derives operational knowledge from redundant user inputs.

3.3 Crowdsourcing vs. Local Sensing

Beyond the limitations associated with offline remote sensing datasets (Section 3.2), the choice of a data acquisition strategy must also consider the feasibility of deploying and maintaining automated local sensing infrastructures. Although local sensors can provide accurate and continuous measurements, their widespread deployment across wildfire-prone areas remains challenging.

Wildfire-prone regions often cover extensive and remote areas, making the installation of sufficient numbers of sensors difficult and economically expensive. A dense sensor deployment would require significant initial investment, careful planning, and continuous maintenance to ensure reliable operation. Furthermore, sensors located in natural environments are exposed to harsh conditions, including extreme temperatures, smoke, weather events, and physical damage caused by wildlife or human activity. The possibility of sensor malfunction, degradation, or theft introduces additional uncertainty and reduces the reliability of a fully automated sensing approach. Additionally, achieving comprehensive coverage through static sensors would require a large number of devices, increasing both deployment complexity and operational costs. In contrast, human contributors can provide observations from locations where fixed infrastructure is unavailable, allowing information to be collected dynamically and adapted to the specific requirements of an ongoing wildfire event.

Moreover, sensors are designed to measure predefined physical parameters, but they may fail to correctly represent unusual situations due to malfunction, cali-

bration errors, or unexpected environmental conditions. Human observers, through experience and situational awareness, can recognize patterns, validate sensor measurements, and provide qualitative information such as visible fire patterns caused by fire induced phenomena, area accessibility information, smoke characteristics, or the possible impact on nearby communities. This human expertise can serve as an additional validation layer, improving the robustness of wildfire simulation inputs.

UAV-based data acquisition represents another promising approach for wildfire monitoring. Unlike fixed sensors, drones provide mobility and can collect high-resolution information from targeted areas, offering detailed spatial observations of active fire regions and fuel coverage. However, their deployment requires specialized equipment, trained operators, operational planning, and significant initial investment. These requirements may limit their availability, particularly when rapid deployment is required.

Future wildfire simulation and monitoring systems should therefore consider hybrid approaches that combine the strengths of different data sources. UAVs can offer broad spatial coverage, while VGI can offer localized observations, rapid updates, and human interpretation of environmental conditions. A future integration of crowdsourced observations, drone imagery, and sensor measurements within an edge-based simulation framework could provide a more complete and resilient representation of wildfire dynamics. Hence, in the proposed near-real-time simulation system, we adopt an approach that moves the intelligence closer to the wildfire event, reduce dependence on fragile infrastructure, and use diverse, human-validated real-world observations rather than relying only on centralized automated data acquisition systems.

3.4 Existing Crowdsourcing Approaches

3.4.1 CITISENS

The CITISENS [72] platform, proposes a dual layer participatory sensing framework in which commodity Android smartphones function as georeferencing

instruments for wildfire hotspot reporting. User-derived observations are dynamically assimilated into a probabilistic wildfire spread simulator (FLogA) [73] through a crowd-sourced *Dynamic Data-Driven Assimilation System* (DDDAS). The principal novelty of the system lies in this closed-loop paradigm: citizen observations generate georeferenced ignition hotspots, which are used to conduct multiple fire spread simulation scenarios, producing updated burn probability maps that, in turn, trigger refined alerts to participating citizens.

The application’s fundamental data primitive is the georeferenced hotspot. Users cannot contribute fire perimeter lines, burned area polygons, fuel type observations, or similar reports. This is a significant architectural constraint, as wildfire spread models operating in developed fire scenarios require fire perimeters as boundary conditions rather than isolated point observations. In the current version of the software, no filtering or noise reduction is applied to user inputs, although the authors indicate that this limitation will be addressed in the future. Moreover, every component of the pipeline – hotspot transmission from reporter to CITISENS server, FLogA simulation server, burn map delivery to the viewer, and planned push-notification alerts – requires continuous bidirectional internet connectivity. These architectural decisions are incompatible with the system’s target deployment context; wildfires can occur in remote, mountainous terrain with sparse cellular infrastructure. Also, active fires frequently destroy telecommunication infrastructure as they spread. The system provides no offline fallback mode, or peer-to-peer communication layer, so loss of connectivity – a likely condition during a major fire event – renders the entire system inoperative. This is arguably the most critical operational limitation of the proposed design.

3.4.2 FireLoc

An alternative to CITISENS, proposed by Fu [74] is FireLoc. FireLoc is a multi-modal wildfire geolocation and mapping system that combines stationary ground cameras (plus Android smartphones mounted on WUI residential rooftops/backyards) with *Digital Elevation Models* (DEM) to achieve low-latency, wildfire localization. Its primary technical novelty lies in three localization algorithms:

TopoMono (DEM-based view ray intersection), DMT (Deep Monocular Topography, recalibrating MonoDepth2 via TopoMono), and FireTri (object-based triangulation across camera pairs).

In that particular crowdsourcing model, the participants contribute stationary camera nodes, rather than active user-originated observations. FireLoc’s crowdsourcing paradigm is therefore fundamentally different from CITISENS, since users do not actively observe, describe, or report anything. Their contribution is a one-time hardware opt-in: mounting a smartphone, pointing it at a vegetated fire-prone area, and leaving it powered. Authors presuppose the availability of grid power for device charging and WiFi or cellular connectivity for scene upload. In addition, the method strongly relies on static historical DEM data. The authors acknowledge that vegetation growth causes the actual terrain surface to differ from the DEM, causing view-ray non-intersection (overshoot). Finally, its detection model is insufficient for nighttime and near-realistic smartphone deployment conditions (e.g., high ISO induced sensor noise under low light conditions).

3.4.3 FDWithoutFire

FDWithoutFire [75] is a crowdsourced emergency management system targeting forest fires. Its architectural premise is a three-tier pipeline: (1) citizens submit free text/audio fire reports via a mobile app; (2) a server side *Data and Information Fusion* (DIF) engine — implementing the Quantify model — processes, qualifies, and fuses reports with heterogeneous external sources (topology, weather, satellite imagery, historical databases); (3) fused situational knowledge is presented to commanders and field firefighters through dedicated interfaces. Incident data are used to construct an ontology, which is formed as an initial semantic instance derived from decoded user inputs describing the active situation. The system’s primary novelty is applying an explicit DIF model to unstructured citizen generated text reports in a fire emergency context.

The system is best characterized as a situational awareness aggregation and alerting platform which is a valuable but incomplete component of a fire emergency management stack, rather than a standalone decision support system, as it lacks

the capability to simulate wildfire dynamics of these scenarios.

3.4.4 Early Wildfire Detection via Vapor Pressure Deficit

Shachaf et al. [76] propose a method which uses passively crowdsourced smartphone sensor data, specifically ambient temperature and relative humidity, to compute *Vapor Pressure Deficit* (VPD), a well-established proxy for vegetation dryness and fire fuel availability [77]. In their publication, the authors demonstrate that calibrated smartphone derived VPD anomalies were statistically detectable days before major wildfire events in Israel and Portugal. It should be noted that this particular crowdsourcing model is entirely passive, with users contributing sensor data without any deliberate action.

The primary limitation of this approach is that it yields a single scalar output per grid cell per day, i.e., a VPD index. This is a fire weather indicator, not a metric that can be used as input for fire spread simulation. It cannot be directly assimilated into physics-based fire behavior models (e.g., Rothermel’s model [39]) which require spatially distributed wind fields, fuel maps, and ignition geometry as primary inputs, none of which this system provides. Additionally, raw smartphone temperature and relative humidity readings are systematically biased since temperature sensors are contaminated by internal heat sources (battery, CPU) and, in any case, these sensors are not common across modern devices.

3.4.5 A Multi-Source Data Fusion Framework for WUI Fire Detection

Zhong et al. [78] proposed a multi-source data fusion framework integrating social media data, remote sensing, cellphone signaling data (from cellular network operators), terrain data, and meteorological data to track WUI fires. Using semantic web technology, the framework is implemented as a *Knowledge Base Service* (KBS) [79] running on top of the *WUIFire* ontology, which formalizes domain knowledge and provides a basis for tracking and analyzing WUI fires by fusing multi-source heterogeneous data. The main crowdsourced component is social me-

dia posts from Weibo (China’s dominant microblogging platform), from which fire situation information is extracted and fused with authoritative data sources. The KBS layer sits on top of this ontology, consuming heterogeneous inputs and enabling SPARQL-based semantic query and reasoning. The pipeline converts raw multi-source data (social media posts, satellite imagery, cellphone location signals, weather APIs, and DEMs) into ontology defined entities (fire objects, affected areas, population exposure, infrastructure status) that can be jointly queried. Finally, a fire spread simulation of the affected area can be conducted.

The conducted fire spread estimation does not dynamically assimilate crowd-sourced data as real-time input to update the simulation. Rather, the simulation is parameterized using terrain, meteorological, remote sensing data, and preexisting sources and its output (fire perimeter predictions) is then stored in the WUIFire knowledge graph. Regarding connectivity, the entire architecture is explicitly server-side as the KBS, ontology store, SPARQL query engine, NLP extraction pipeline, and remote sensing data all operate on centralized infrastructure. There is no edge or offline capability described, meaning continuous network connectivity is a hard requirement for the framework to function, which is a significant operational constraint in the WUI environments, where telecommunications infrastructure cannot be taken for granted.

3.4.6 Social Media Driven Fire Detection

Florath et al. [80] used a *Volunteered Geographic Information* (VGI) system as a dynamic crowdsourcing mechanism for real-time disaster management. By continuously harvesting and spatially analyzing Twitter (now X) crowdsourced data, authorities can rapidly delineate wildfire impacted areas and monitor event progression. User generated tweet posts are processed through a pipeline incorporating semantic, spatial, and temporal filtering, followed by noise reduction procedures, including deduplication and relevance classification. Subsequently, the algorithm performs geolocation and spatial entity extraction, coupled with spatiotemporal clustering.

The aim of the method is to make an approximate assessment of the region

requiring attention from first responders, rather than mapping an exact, definitive fire perimeter. Moreover, additional geospatial data that could potentially be used to simulate or estimate wildfire spread are not incorporated (e.g., imagery of vegetated areas for fuel composition extraction and hotspot identification). Furthermore, the method assumes that users have continuous access to network connectivity in order to generate and publish social media posts, which may be an issue during wildfires.

3.4.7 Multimodal Fire Spread Prediction Framework

The *Multimodal Fire Spread Prediction Framework* (MFiSP) [81] proposes a fire spread prediction framework that integrates social media data (Twitter/X) and NASA FIRMS [82] remote sensing observations as dual crowdsourced/satellite inputs into a Monte Carlo ensemble simulation pipeline built on FlamMap’s Minimal Travel Time fire growth model. Its primary novelty is a Rate of Spread manipulation mechanism that dynamically recalibrates fuel load maps between hourly assimilation cycles, by selecting the perturbed fuel map whose simulated perimeter best matches the observed combined perimeter, effectively using incoming social media and remote sensing data to steer the simulation’s physical parameters proactively. Social media posts are periodically processed to generate fire behavior polygons encoding four behavioral states (spot fire, suppression, intensification, abatement). These polygons directly modify the fuel load maps input to FlamMap software.

The deployment architecture is not explicitly specified. However, given that FlamMap is a server side simulation tool and that the Monte Carlo ensemble approach requires executing multiple parallel instances per cycle, it is reasonable to assume that the method requires continuous network connectivity and substantial computational resources. The retrieval of social media data relies on API access to Twitter/X, which depends on third party services. Furthermore, the system does not account for edge processing, offline functionality, or operation in power-constrained environments.

3.5 A Web-Based Edge Simulator for Limited Connectivity Scenarios

The approaches presented in Section 3.4 for the utilization of crowdsourced data all have their limitations, especially in limited connectivity and power constrained scenarios, which are common when mobile devices are used close to a wildfire event. They also do not exploit detailed user-provided geospatial inputs (e.g., fuel polygons and fireline position) which are required for accurate spread simulation. Therefore, we argue that a more field-deployable approach is required, that can incorporate the specialist knowledge of firefighting personnel and operate even when connectivity is limited or disrupted. Our goal is to enable firefighters operating in the field to act as both the primary end users of the system and the providers of observational data, allowing them to continuously enrich the simulation with up-to-date information collected during ongoing suppression activities. The proposed framework is therefore designed around a progressive simulation paradigm, whereby newly acquired field observations are incrementally incorporated into the model as they become available, triggering successive updates of the simulation state. This design choice is further supported by the selection of a CA-based modelling approach for the core simulator (Section 2), as it enables computationally efficient re-execution of the model following each data update, while providing a valid system state at every simulation time step. This is in contrast to more computationally intensive approaches (e.g., MTT-based formulations [83]), which may require longer convergence times or are less suitable for frequent simulation executions. This enables firefighters to locally perform near-real-time simulations that reflect the evolving conditions of the incident, even in scenarios where network connectivity is intermittent or unavailable, ensuring that decision support remains operational under constrained communication environments.

The proposed platform utilizes a lightweight, browser-based hybrid architecture designed to facilitate VGI collection and edge-computed simulations in WUI zones. To address the computational constraints of mobile edge devices while ensuring high-resolution data fidelity in complex WUI areas, the platform adopts

a hybrid distributed architecture. Implemented as a cross-platform *Progressive Web Application* (PWA), the system leverages a thick-client model where the core CA-based simulation logic is executed locally within the client’s browser environment using high-performance web technologies such as JavaScript and, in future implementations, WebAssembly.

The selection of a thick-client model allows the computational load of the CA-based wildfire propagation algorithm to be fully offloaded to the client device, accessing as much as possible data that is stored in the device’s local storage to perform the simulation. This is achieved using optimized JavaScript to execute the simulation locally within the web browser, thereby minimizing and even eliminating the need for a central server, reducing latency, and enabling offline functionality in remote areas. This is especially important when the communication infrastructure is limited (in remote areas), damaged (from the fire) or overloaded (from potential evacuees). The edge computing paradigm adopted in this work is inspired by a similar *Information-Centric Networking* (ICN) [84] approach presented in [85].

Unlike traditional centralized systems, the proposed architecture integrates a *Peer-to-Peer* (P2P) communication layer to facilitate collaborative sensing and computational offloading. The user interface integrates an interactive mapping component based on Leaflet [86] that allows users to manually digitize homogeneous fuel zones as vector polygons overlaid on satellite-based tiles, if network connectivity is available or cached tiles exists (Figure 3.1). Each polygon is stored as a GeoJSON object, encapsulating its spatial geometry, alongside user defined attributes from predefined categories, such as fuel density, canopy height, and moisture content. This allows firefighters or other stakeholders to enter their own up-to-date observations from the field.

User contributed data (that is, VGI) may be propagated directly between peers to fill data voids in the local simulation environment (Figure 3.2), where users are firefighters actively engaged in field suppression activities. For example, a user may be unsure about the fuel composition in a vegetation strip or be unable to identify it due to smoke cover. Once network connectivity is restored, even partially and by any individual user, the data can be uploaded to a backend server

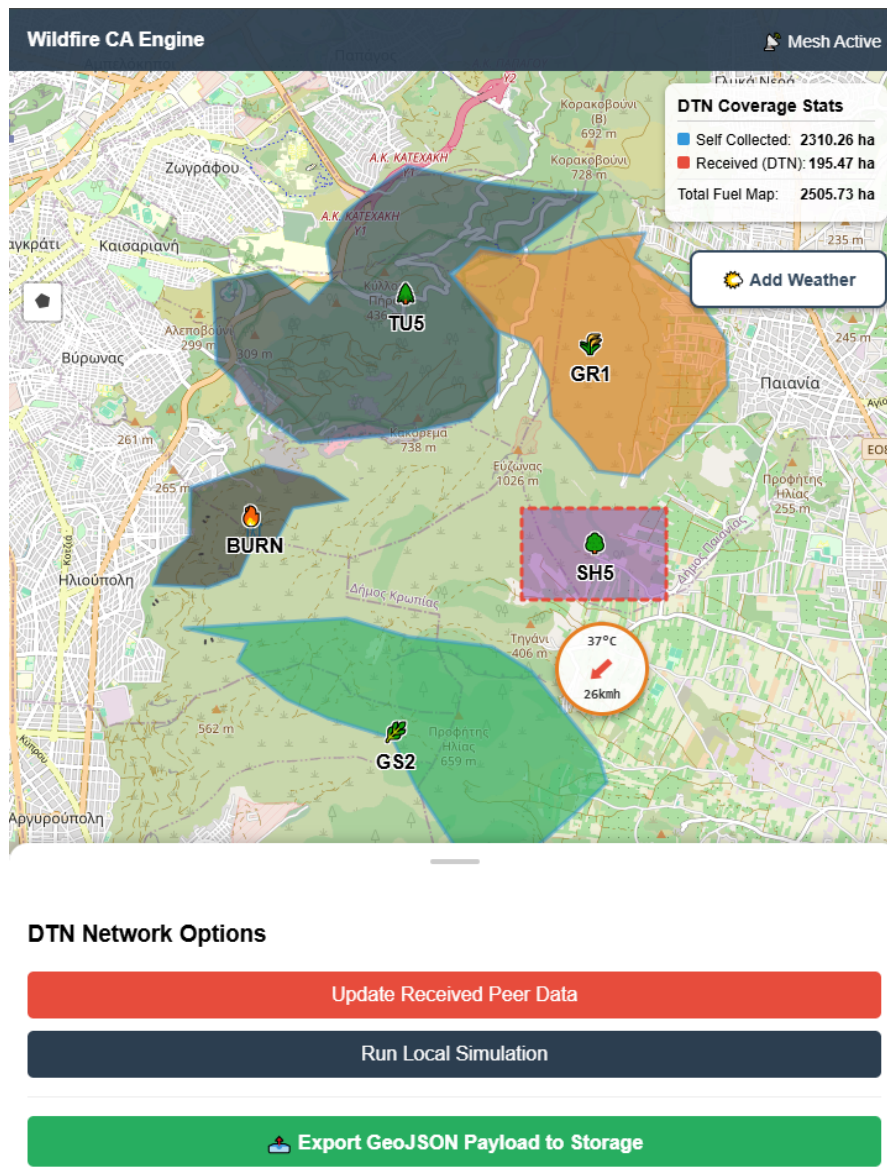


Figure 3.1: Overview of the VGI interface as viewed from a user's mobile device, showing weather inputs and fuel data collected from the area. Adapted from [20].

to disseminate the evolving situational knowledge to stakeholders outside the field.

To optimize energy consumption on battery constrained mobile devices, simulation generated outputs can also be shared across peers. This allows peers to retrieve pre-calculated propagation results, rather than repeatedly executing the CA algorithm, thus distributing the computational burden across the connected edge cluster. A central backend server, accessible via a RESTful API, functions primarily as a synchronization hub rather than as a computing engine. It can aggregate the received crowdsourced fuel configurations and distribute corresponding updates to users that have access to the cellular network, allowing them to patch missing data in the WUI zone and share simulation outcomes in near-real-time.

Given the volatile connectivity users exhibit in wildfire environments, the platform implements a *Delay Tolerant Networking* (DTN) application layer [87, 88, 89] to ensure data persistence and support offline operation (Figure 3.3). Unlike traditional synchronous web applications that require continuous connectivity, a DTN-based system adopts a store-and-forward paradigm, allowing edge devices to operate offline, cache data locally, and forward messages (called DTN bundles) opportunistically whenever connectivity becomes available.

All spatial data updates (e.g., fuel polygon modifications) and simulation progress are encapsulated into lightweight GeoJSON FeatureCollections in each application’s local storage. The shared features conform to, and are structured according to, the framework of a lightweight ontology for wildfire simulation [21] proposed in Chapter 4, which defines the core concepts, relationships, and attributes necessary to support consistent representation and analysis across simulation components. These messages can be stored in the browser’s IndexedDB until a new transmission path is established. In remote areas where the cellular infrastructure is unavailable or unreliable, the platform transitions into an opportunistic *Device to Device* (D2D) operating mode [90]. Local peer discovery can be achieved with mechanisms such as *multicast DNS* (mDNS) or *Bluetooth Low Energy* (BLE) advertisements, accessed via native wrappers where required, to identify nearby devices within physical proximity. Upon discovery, the DTN agent initiates a synchronization handshake during which peers exchange their current

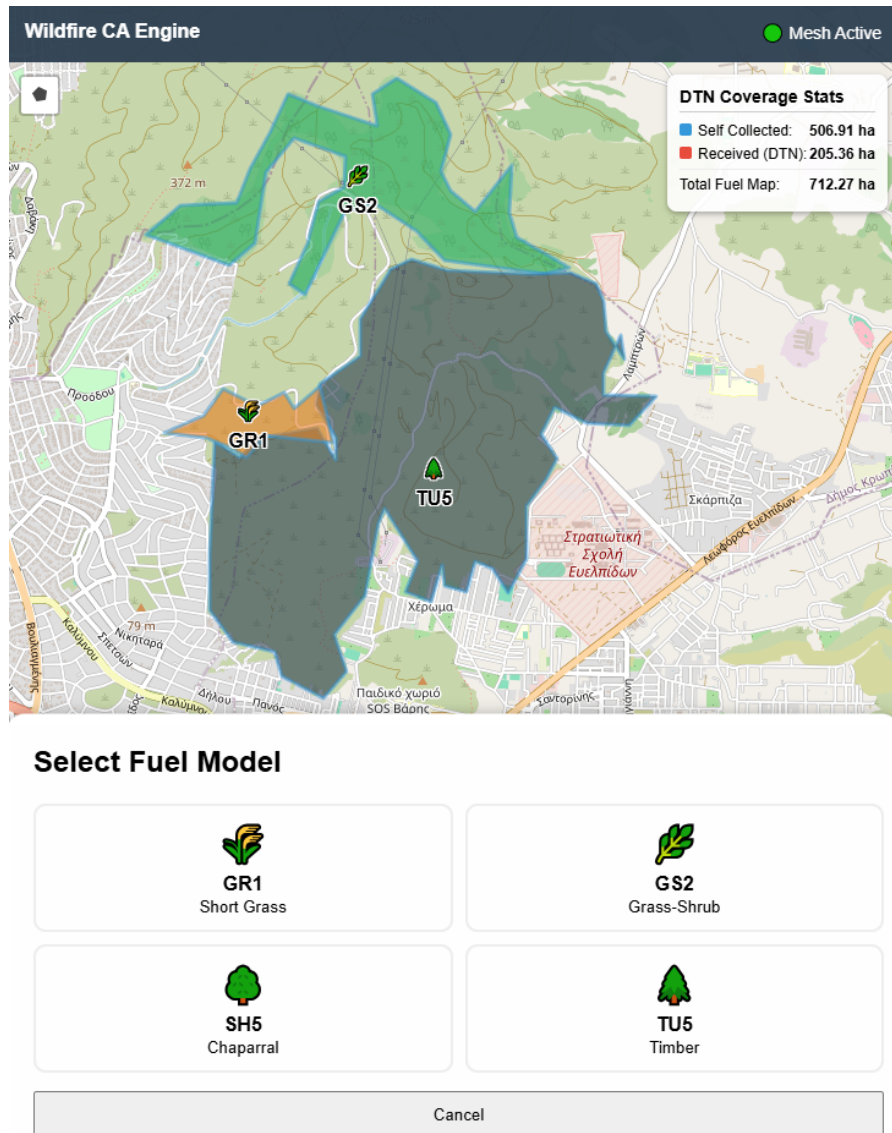


Figure 3.2: User inserted fuel data and wildfire updates propagate through the DTN (*application screenshot*). Adapted from [20].

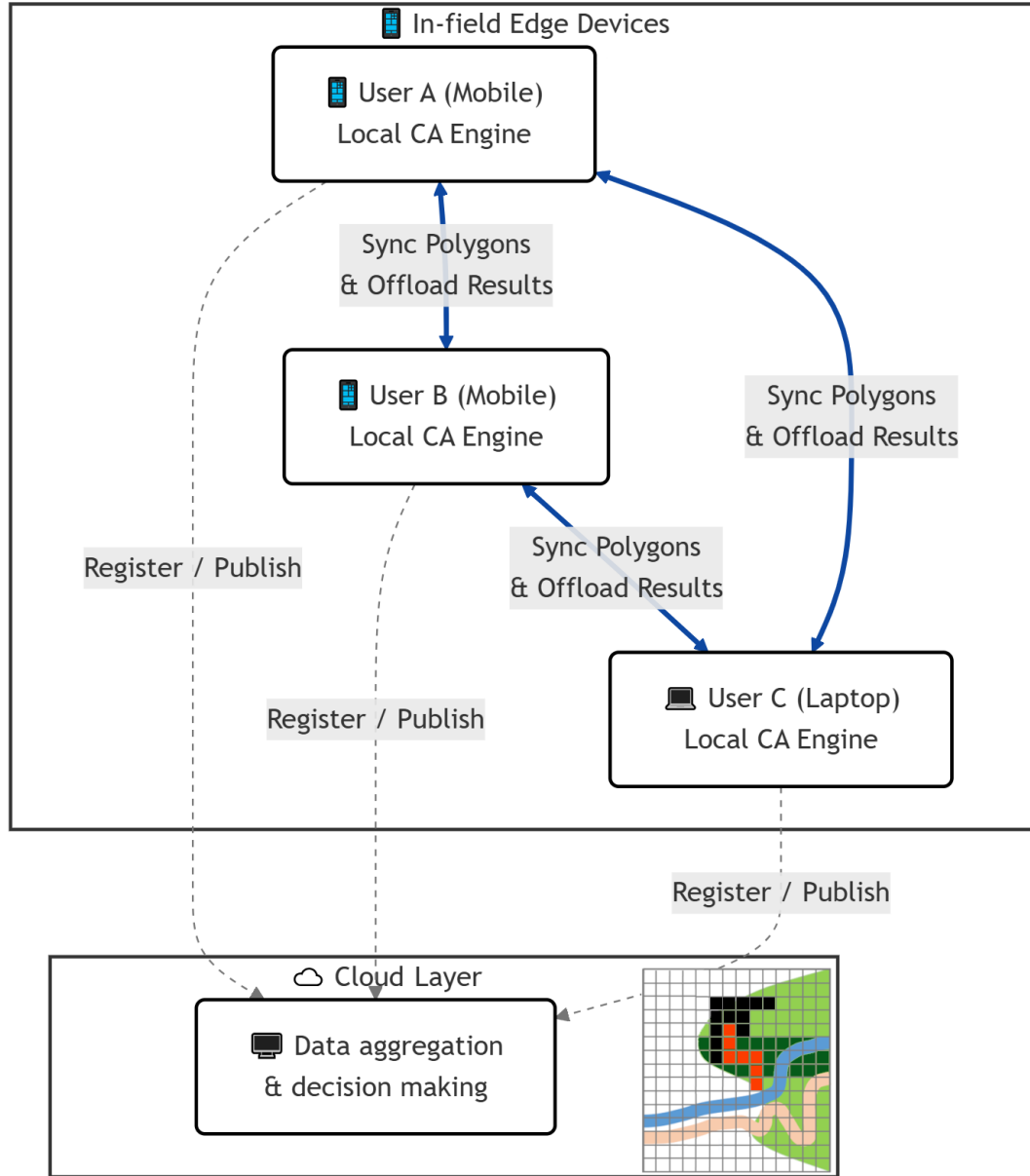


Figure 3.3: High-level representation of the proposed *VGI / DTN* distributed architecture. Adapted from [20].

knowledge about the state of the simulation environment. As a result, WUI fuel data propagates virally across the loosely connected mesh cluster, maximizing data dissemination and supporting eventual consistency whenever disconnected network partitions are reconnected.

Chapter 4

A Lightweight Ontology for Real-Time Wildfire Simulation

4.1 A Semantic Model for Wildfire Simulation

Effective wildfire simulation increasingly relies on timely and context-rich information originating from in-field observations and *Volunteered Geographic Information* (VGI). Especially in the case of CA-based wildfire simulation, even slight deviations from ground-truth input data can substantially propagate into significant discrepancies within the simulated fire perimeter [16]. However, such information is often heterogeneous, semi-structured, and described using non-standard terminology, limiting its direct integration into simulation environments. To facilitate interoperability and consistent interpretation of reported data from human, in-field observers, we introduce a lightweight semantic ontology to formally describe key wildfire simulation components and events reported through a VGI submission interface. The proposed approach aims to provide sufficient semantic structure for information exchange and automated processing, while remaining flexible enough to accommodate the dynamic and imperfect nature of field observations reported by human participants.

4.2 Architectural Decisions

The conventional approach to express an ontology is using formal semantic representation languages such as the *Resource Description Framework* (RDF), *RDF Schema* (RDFS), and *Web Ontology Language* (OWL), typically serialized in formats including RDF/XML, Turtle, or JSON-LD [91, 92]. These representations are designed for processing by semantic reasoning engines and ontology management frameworks such as Apache Jena, Protégé, Pellet, HermiT, or FaCT++, which support inference mechanisms including description logic reasoning, class subsumption, consistency checking, semantic rule evaluation, and SPARQL-based knowledge querying.

Although such technologies provide strong semantic expressiveness and reasoning capabilities, their deployment is often impractical in edge-based operational environments characterized by constrained computational resources, intermittent connectivity, and strict energy limitations. In particular, mobile and edge platforms operating on Android devices typically exhibit limited CPU performance, memory availability, battery capacity, and storage resources, especially when executing real time geospatial processing pipelines alongside sensor acquisition, communication, and graphical interface rendering tasks. Even if such software could theoretically be deployed on modern mobile devices, its practical integration into mobile edge environments remains highly challenging. The deployment and, more importantly, its long-term maintenance across heterogeneous Android and iOS ecosystems would require substantial engineering effort due to platform fragmentation, evolving operating system constraints, and frequent changes in mobile APIs and runtime environments. Moreover, ensuring continued compatibility and stable operation following major operating system updates would likely depend on extensive third-party library support and continuous software adaptation. These limitations significantly reduce the practicality of heavyweight semantic reasoning infrastructures for the proposed lightweight, field-deployable wildfire VGI application described in Section 3.5.

For this reason, the proposed ontology is represented using GeoJSON rather than conventional Semantic Web encodings. GeoJSON provides a lightweight,

human-readable, and natively geospatial-aware data representation that can be directly consumed by ad hoc VGI collection and wildfire monitoring applications operating at the edge, as well as by third-party conventional web applications. This design choice aims to provide straightforward interoperability with mobile GIS frameworks, web mapping libraries, and geospatial processing pipelines commonly used in Android / Web based environments, including Leaflet, OpenLayers, Mapbox, and Cesium JS. Instead of relying on formal semantic representation, the proposed approach adopts a lightweight and relaxed semantic structure embedded within GeoJSON feature properties and metadata fields. This strategy preserves sufficient semantic consistency for data exchange, interpretation, and integration, while avoiding the deployment complexity and runtime costs associated with full ontology reasoning infrastructures. Consequently, the framework remains suitable for real time wildfire field operations, where robustness, portability, low latency, and minimal computational overhead are critical requirements.

4.3 Knowledge Representation Requirements

The term ontology denotes the systematic and explicit modeling of domain concepts and their semantic relationships [93, 94]. To ensure that the concepts are clearly and correctly interpreted, a set of ontological commitments (agreements to use a common vocabulary consistently) are specified in [94]. From the perspective of an in-field wildfire observer, the adoption of a strict formalism and the associated structures required to accurately represent entities and their relationships may introduce additional complexity and increase the likelihood of misinterpretation. For example, in the case of SoKNOS [95] (see Section 4.4.3), the authors mentioned that domain experts (firefighters) found the terminology *quite irritating*, as it did not match their own perspective and colloquial language. This implies that the existence of a formalized ontological hierarchy and a reasoning framework do not guarantee usability or interpretability, particularly when the adopted terminology and conceptual structures diverge from the users' domain-specific understanding.

To illustrate the difficulties arising when designing an ontology for wildfires,

consider the concept of a *forest*. While the forest is a key entity in wildfires, the interpretation of the term is not universally consistent, as its meaning may vary significantly depending on the scientific, administrative, cultural, or operational perspective of the individual or organization using it. This semantic variability, commonly referred to as *polysemy*, constitutes a major challenge in the development of ontologies and knowledge representation systems for wildfire applications [94]. For example, a forest ecologist may define the forest as a complex ecological system composed of interacting biotic and abiotic components, characterized by tree coverage exceeding a specified threshold. In contrast, administrative or legal definitions often emphasize jurisdictional boundaries rather than ecological characteristics, describing a forest as any land area managed by a forestry authority, regardless of its actual vegetation density. Such definitions are more likely to be used by local forest or fire services, whose responsibilities are tied to clearly defined spatial jurisdictions. A GIS engineer may interpret the same term using remote sensing or land cover analysis, where a forest is identified simply as an area exhibiting dense vegetation or tree cover based on observational data.

These alternative definitions illustrate that the same term may correspond to substantially different conceptual representations depending on the application context. Such semantic discrepancies become particularly critical in wildfire simulation systems, where the interpretation of *forest* directly affects fuel characterization, fire spread dynamics, risk assessment, and operational decision making. For example, an ecological definition of a forest may emphasize vegetation composition and biomass properties relevant to combustion behavior and selecting appropriate fuel models [96, 97], whereas an administrative definition may be more suitable for resource management and emergency coordination purposes.

A further challenge emerges when attempting to construct hierarchical representations for concepts whose relationships are not strictly defined or unambiguous [94]. In many real-world domains, including wildfire modeling, conceptual classifications may follow multiple, partially overlapping criteria, leading to alternative and sometimes incompatible hierarchical structures. For instance, a hierarchy may be constructed based on land-cover classifications, while an alternative hierarchy

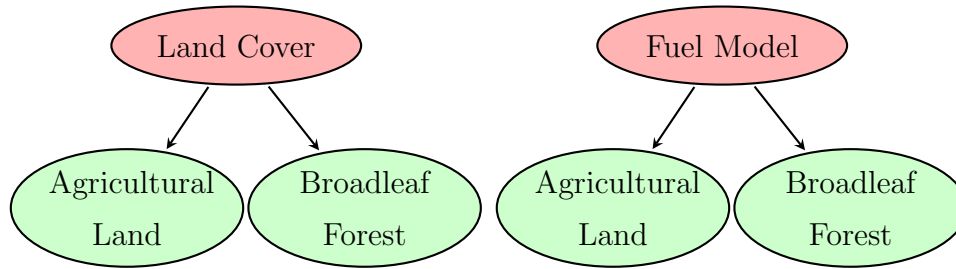


Figure 4.1: Overlapping hierarchical structures, in which the same conceptual entity may occur as a subclass within different ontology classes. Adapted from [21].

that focuses on implementing the wildfire spread simulator itself may instead be derived from fuel-type models, each reflecting different modeling assumptions and application objectives (Figure 4.1). Another common source of ambiguity in such structures arises from the unintended mixing of distinct semantic relationship types within a single hierarchy. In particular, subtype–supertype (is-a) relationships are often conflated with part–whole (has-a or part-of) relationships, despite their fundamentally different ontological meanings. This confusion can result in inconsistent or semantically incorrect models, where hierarchical links do not preserve a single, coherent interpretation criterion. Figure 4.2 depicts a high-level representation of wildfire-related concepts in which different types of semantic relationships are intentionally combined within a single structure. Specifically, it shows the mixed modeling of parts of a wildfire, with subtypes (e.g., crown fire) of it. In the context of wildfire simulation software these issues are especially critical, as hierarchical inconsistencies can propagate through reasoning mechanisms, leading to incorrect inference of fuel properties, spatial dependencies, or fire spread behavior. Therefore, careful separation of taxonomic relationships is essential when designing ontological structures, ensuring that each hierarchy preserves a single semantic principle and supports both human interpretability and machine reasoning [98].

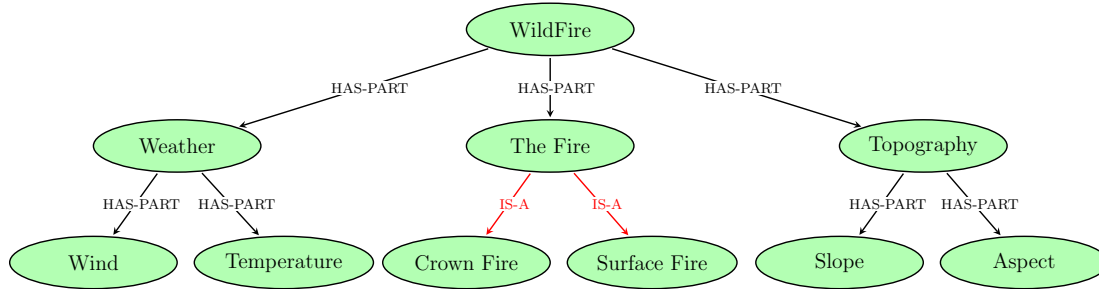


Figure 4.2: A hierarchical structure with relationships of different types. Adapted from [21].

4.3.1 Ontology Modeling Challenges

Representing the multifaceted and highly dynamic environment of wildfires through a unified ontological framework constitutes a non-trivial challenge. It is expected that conceptual and taxonomic misunderstandings arising from differences in stakeholder perspectives, variations in training and methodological background, as well as cultural and social influences will affect the design and deployment process [94]. In this context, the key difficulties that must be addressed in order to develop a standardized ontology for wildfire simulation can be categorized as follows [94, 99]:

Ontology-related challenges

- **Verifying an ontology-based model:** verification requires deep understanding and technical awareness of the knowledge domain modeled. Poor modeling can result in model inconsistencies and incoherent reasoning.
- **Expanding and updating a model:** trying to update or expand the model to incorporate new features may lead to constraint violations or misleading information.
- **Unrealistic perspectives of the environment:** a model tries to represent the environment, but it is limited by its creator’s knowledge barriers and cultural biases. As a result, some elements of the real-world environment may only be partially or inadequately captured.

Human behavior-related challenges

- **Terminology and polysemy:** stakeholders may interpret the same term with different meanings, or misunderstand a key concept leading to conceptual ambiguity.
- **Ambiguous data:** when users provide observational inputs required by the model, the submitted data may be ambiguous, incorrect, conflicting, or noisy when collected under uncertain or heterogeneous conditions.

The effectiveness of the proposed VGI (Section 3.5) for wildfire simulation, depends heavily on its ability to accommodate multiple semantic perspectives while maintaining conceptual consistency across data sources and user groups. In this context, ontology design should prioritize semantic interoperability, adaptability, and user-oriented conceptualization, ensuring that both domain experts and operational stakeholders can interact with the system using terminology and abstractions that are meaningful within their respective contexts.

4.4 Existing Ontological Approaches

4.4.1 OntoFire

OntoFire [63] is a geoportal service that enables users to retrieve and publish geospatial information related to, among others, past and active wildfires via a web interface. Access to the service for publishing is limited to authorized users with a strong scientific background. It is structured around the relationship between Natural Risk, Infrastructure, and Physical Environment, formalizing the interaction between biophysical hazards and anthropogenic vulnerabilities. The system is built using RDF Schema to define its vocabulary and adheres to the ISO 19115 metadata standard to describe various geospatial information resources. Architecturally, the system relies on a centralized three-tier model (Apache Tomcat and PostgreSQL, serving also for RQL [100] queries), which is optimized for the discovery of existing metadata rather than the decentralized, real-time data synchronization and aggregation required for edge-based simulation environments.

4.4.2 Onto-Safe

Onto-Safe [101] is a semantic framework designed to facilitate wildfire emergency management across all phases of the disaster risk management cycle. It is constructed using an OWL 2 formalism and follows a modular architecture that reuses and combines established ontologies, such as SoKNOS [95] and BeAWARE, to promote interoperability. It gathers information from heterogeneous geospatial data collected from diverse sources, such as social media, sensors and relationships between responding services. A key novelty is the use of SHACL [102] compliant rules to infuse expert knowledge and generate automated alerts (e.g., when the *Fire Weather Index* (FWI) exceeds a value). The framework serves more as an event linking and mission assignment tool, while the reasoning module is sophisticated and its practical performance in the field will be challenging.

4.4.3 SoKNOS

SoKNOS [95, 103] is a prototype for disaster management that utilizes semantic technologies to facilitate mutual understanding between diverse organizations during large-scale incidents. Its architectural core is a domain ontology on emergency management aligned with the foundational *Descriptive Ontology for Linguistic and Cognitive Engineering* (DOLCE), which provides a semantically unambiguous basis for shared information. A central component of its modular architecture is the *Joint Query Engine* (JQE), which integrates heterogeneous, distributed databases at run-time by mapping their schemas to the resources ontology using F-Logic rules.

The reliance on *strict formalisms* like DOLCE introduces abstract concepts that often confuse or irritate professional end users like firefighters who find such terminology removed from their colloquial language. This complexity creates a granularity gap that necessitates a dedicated ontology engineer, as untrained domain experts cannot easily formalize terminology for reasoning tasks or navigate through them.

4.4.4 BeAWARE

The BeAWARE project [104] implements an ontology [105] aiming to support a decision support system during climate related crises, such as forest fires, floods, and heatwaves. It is constructed using OWL 2 and strongly emphasizes interoperability by reusing and adapting established vocabularies. Specifically, it adopts MOAC for disaster impacts, SoKNOS [95] for resource and damage categorization, and OASIS [106] for mission assignments. The development was strictly driven by end-user requirements, which were formalized into *Competency Questions* (CQs) and translated into SPARQL queries to validate the model’s structure. However, since it functions as a semantic integration layer for external data streams, its operational effectiveness is heavily dependent on the accuracy of upstream analysis components.

4.4.5 SIADEX / BACAREX

SIADEX [107] is developed around BACAREX, an ontology server, implementing an ontology developed especially for forest firefighting. Unlike pure metadata portals or observational models, SIADEX functions as an active scheduling engine. It utilizes a specialized gateway to translate the knowledge stored within the ontology to the *Planning Domain Description Language* (PDDL). This allows an underlying *Hierarchical Task Network* (HTN) planner [108] to automatically generate, monitor, and dynamically patch temporal attack plans in a distributed, service-oriented web environment.

A highly distinguishing feature of SIADEX is how the ontology is engineered to explicitly model real-world uncertainty across three specific dimensions. First, temporal uncertainty is handled by representing uncertain event times, such as deadlines and action completion times, as time intervals (e.g., [t1, t2]) rather than as precise timestamps. Second, resource uncertainty is handled by leveraging the ontology’s class hierarchy to abstract resource usage assigning an abstract variable to missing resources. Finally, location uncertainty is managed by allowing a resource’s current position to be mapped to a broader GIS area within the ontology if a precise location cannot be perfectly known.

4.4.6 WUI Fire System Ontology

Ju [13] introduces a knowledge-driven framework designed specifically for dynamic risk assessment of WUI fires. At its core is the WUI fire system ontology, built by extending standard W3C standards (such as OWL, *Semantic Sensor Network Ontology* (SSN), TIME, and GeoSPARQL).

The system employs the FireSTS framework, which utilizes a dual database strategy. Textual attributes and semantic relationships are stored in a Neo4j graph database to enable rule-based reasoning, while heavy geographic processing is delegated to a PostgreSQL spatial database. This architecture allows an embedded inference engine to continuously update the fire’s hazard, exposure, and vulnerability metrics dynamically based on incoming sensor data and simulation outputs. Authors state that conventional ontology reasoners (such as those integrated into Protégé) were insufficient for the complex spatiotemporal data processing required for fire spread simulation and risk calculation. Instead, they use a custom, rule-based inference engine encapsulated within their FireSTS framework. The reasoning engine integrates the fire simulation program with GraST [109], which is a specialized geospatial temporal semantic query optimization framework.

4.4.7 Fire Detection Ontology

Developed within the scope of the European SILVANUS project, it focuses on establishing a dynamic semantic knowledge base to support early warning systems and operational decision-making [64]. Its core aim is to seamlessly fuse highly diverse, heterogeneous data streams, such as environmental IoT sensors, meteorological forecasts, and crowdsourced social media, into a unified ontological structure. By applying semantic reasoning rules over this integrated knowledge graph, the system automatically extracts actionable insights and triggers real-time alerts. Instead of focusing on the mechanical physics of fire propagation, this ontology acts as an intelligent aggregation layer that standardizes multi-source intelligence to feed AI-assisted decision support models.

4.4.8 NOA Ontology

The system’s semantic foundation is built upon the NOA Ontology [110], which models raw earth observation data (satellites, sensors) and generated products (hotspots, shapefiles) and is interoperably aligned with the foundational *Semantic Web for Earth and Environmental Terminology* (SWEET) ontology [111]. To handle dynamic spatial geometries, the authors introduced stRDF, an extension of the standard W3C RDF that utilizes OGC standards (Well Known Text and Geography Markup Language) [112, 113] to represent geospatial data that changes over time. The system continuously consumes raw satellite imagery via an acquisition station into MonetDB, where primary fire detection algorithms are executed. These initial hotspot detections are then semantically refined and integrated with auxiliary spatial data within the Strabon RDF store before being visualized on a real-time web interface for crisis management purposes. Rather than employing a traditional description logic reasoner, the system achieves spatial and temporal reasoning through the execution of complex stSPARQL update statements within the Strabon [113] database.

4.4.9 Comparison and Limitations

Table 4.1 presents some of the most well known ontology-based approaches that model natural disasters and fire related scenarios. They demonstrate significant similarities, but none of them can be used unmodified when nearly real-time data from a fire incident are used to conduct a *mostly accurate* wildfire simulation, especially when stakeholders suffer from limited / poor network coverage. These approaches are designed to tackle problems related to natural disasters and the management of rescue and suppressing missions. Stakeholders, natural risks, meteorological conditions, topological data, and so on, are mapped to classes with corresponding attributes. Semantic information is represented as linked class instances. Authoring of such classes can be done using RDF and OWL. This has the advantage that we can query RDF data initially stored in flat files using SPARQL [114] in a SQL-like manner.

However, these solutions were not developed to support simulation environ-

Table 4.1: A comparative overview of ontology-based approaches modeling wild-fires and natural disaster incidents. Adapted from [21].

Publication	Main Purpose	Retrieval / Reasoning	Ontology Language	Architecture
OntoFire [63]	Geo portal	RQL [100]	RDF / ISO 19115	Client-server (WEB)
Onto-Safe [101]	Forest fire fighting	SHACL [102]	OWL 2	Generic Framework
SoKNOS [95]	Incident management	JQE / F-Logic	DOLCE / OWL	Generic Framework
BeAWARE [104, 105]	General purpose crisis management	SPARQL	OWL 2	Generic Framework
SIADEx / BACAREX [107]	Forest fire fighting	HTN planning	Protégé (OWL)	Client-server (WEB)
WUI Fire System Ontology [13]	WUI fire risk	GeoSPARQL	Protégé (OWL)	Generic Framework
Fire Detection Ontology [64]	Wildfire detection	GraphDB / SPARQL	RDF	Generic Framework
NOA Ontology [110]	Wildfire monitoring	stSPARQL	stRDF	Client-server (WEB)

ments, therefore they do not meet some simulation-specific requirements, especially for dynamic simulations. One important problem in a wildfire context is that the data gathered in the field are subject to a user’s judgment and personal knowledge about the evolving phenomenon. Poor judgment, fragmentary knowledge, and lack of experience may lead to datasets of poor quality, with limited value to simulation software. Therefore, the ontology must be capable of handling *contradictory data entered in the field*.

Another problem is that a wildfire is a natural phenomenon which evolves dramatically over time. Even minor changes to parameters (such as wind direction) may have a huge impact in the final outcome [16]. Due to the phenomenon’s fast-changing nature, any information gathered is subject to regular updates over time. For example, areas with vegetation (combustible fuels) are displaced by burned areas, fuel humidity declines due to radiation from flames, fuel discontinuities may be formed from firefighting activities, etc. Therefore, the ontology must take into account *the time of each observation*.

Finally, the *client-server* architecture adopted by most solutions is a limiting factor when we need to exploit the collective knowledge of users deployed in the field. Not only the connectivity in the wildfire area is questionable, but each individual stakeholder cannot interpret and model the incident and its surroundings at their full extent. Firefighters deployed in the field have the experience, the knowledge and the proximity to identify, describe and model the required parameters, unlike in previous solutions where information is spread top-down by a small group of experts or stakeholders. This implies that updated information must

propagate rapidly among nearby users. However, due to the dynamic and intermittent connectivity between users, *each participant may possess only a partial view of the overall situation*; consequently, the available data at a given time may be insufficient for effective inference or reasoning.

4.5 A Wildfire Ontology for Near Real-Time Simulation

A common problem with ontology-based approaches to wildfire prediction is that both users and software designers often assume that the model and the acquired data constitute an accurate and complete representation of the wildfire environment, overlooking the fact that they remain subject to uncertainty, evolution, refinement, and improvement. This is clearly untrue in a dynamic environment with imperfect knowledge. Our proposed model addresses this challenge by explicitly acknowledging that user observations may be conflicting and temporally dependent, thereby requiring the integration and fusion of diverse individual perceptions into a continuously updated collective representation of the model. Storing multiple independent representations of the same entity allows the use of interpolation techniques (e.g., weighted averaging [115], consensus modeling [116], metaheuristic algorithms [117]), for merging user inputs in a statistically robust way. The aggregated geometry is more likely to converge toward the true spatial ground truth, as it integrates the most probable components of each input, rather than prematurely forcing agreement. The process of spatial aggregation for the entities will be presented explicitly in Chapter 5. Additionally, it requires less resources, since new inputs (due to continuous data evolution) can be compared directly to all prior representations, without requiring a costly re-topologizing of the dataset. In the context of a mobile, decentralized wildfire simulation platform that relies on crowdsourced user data – where uncertainty is inherently present – prioritizing accuracy and replayability over minimal storage constitutes a robust methodological choice and defensible design principle.

4.5.1 Handling of Duplicate and Ambiguous Inputs

In contrast to existing approaches that assume only one or a limited number of stakeholders are responsible for mapping the real world to corresponding ontologies, we instead introduce a decentralized architecture in which any user can contribute to, and disseminate, new knowledge across the system. As a consequence, we expect that the same ontology (eg., a forestry area with a given type of vegetation) will be modeled by different users with different experience, knowledge and even under different time frames. In some approaches, this is not even possible, and inevitably will introduce ambiguous, but not necessarily incorrect input data. Moreover, our approach introduces an additional challenge: it is not feasible to map a single ontology in a one-to-one manner across all users involved. Factors such as human error, limited GPS accuracy, and rapidly changing environmental conditions can lead to divergent representations of the same area across different users' stored data.

As an example, consider the case in which two users (e.g., firefighters) deployed in the field submit, via a mobile mapping interface, two forested areas with different fuel types by drawing polygons on a map, which are subsequently used as inputs to the simulation engine. It is likely that the final submitted polygons will differ, resulting in inconsistencies and potential conflicts. Figure 4.3 illustrates this scenario, in which two observers independently submit representations of the same geographic area as two distinct forested polygons. The intermix zone between the two vegetation types is judged differently among them and noted with red (first observer) and blue (second observer) lines. This design choice results in ambiguous regions (silver polygons), where the final representation of the area is derived through the aggregation of multiple user inputs. This approach enables a more accurate modeling of transitional or border regions that lack clear classifications or well-defined boundaries. As additional user observations are incorporated, the resulting model progressively improves in accuracy and reliability. Assuming that the majority of users provide accurate observations, high quality inputs are expected to outweigh erroneous or low quality contributions through the aggregation and interpolation of the progressively constructed dataset.

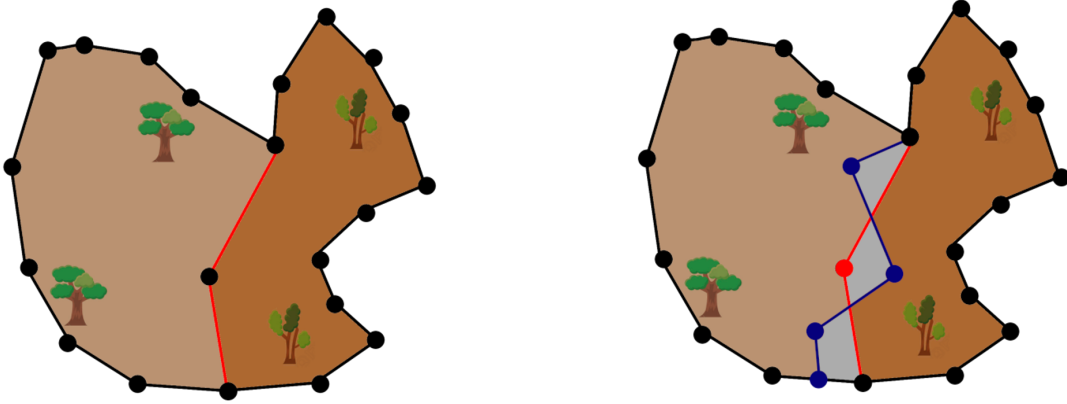


Figure 4.3: Digitization of the same forested area by different users via a mobile mapping interface, leading to ambiguities. Adapted from [21].

With this challenge addressed, a related issue must also be considered, namely the fact that users submit independent observations without being able to reliably determine whether the ontological entity they submitted corresponds to an already existing entity. This limitation arises from the decentralized and intermittently disconnected nature of the system, in which globally consistent identifiers (such as ids) cannot be reliably assigned, synchronized or maintained across devices at the edge. If the observations refer to the same real-world object, a mechanism is required to determine which object parameters will ultimately be used by the simulation. Consequently, an efficient and lightweight interpolation method is necessary, particularly given the computational constraints imposed by mobile devices. In Chapter 5, we describe an ant colony optimization-based algorithm capable of normalizing the uncertainty and fuzziness associated with user provided inputs and determining shape similarities while preserving the geospatial primitives (polygons, polylines) required by the simulation engine.

4.5.2 Representation Scheme

The final technical aspect to be addressed concerns the data representation scheme. Since the ontology must support both quantitative and spatiotemporal data, adopting the stRDF (spatiotemporal RDF) scheme constitutes the most obvious solution. The stRDF scheme is an extension of the RDF model designed

to support the representation and querying of spatiotemporal data through semantically enriched geospatial and temporal constructs [112, 113]. It natively uses widely adopted formats like *Well-Known Text* (WKT) and *Geography Markup Language* (GML) to represent geometries as literals of datatypes. Also, it introduces a datatype named *period*, allowing intervals to be represented as literals of the datatype *strdf:period*. Correspondingly, stSPARQL extension provides support for spatial queries involving geometric representations, while also facilitating temporal operations over discrete time instants and temporal intervals.

While semantic web standards such as stRDF or GeoSPARQL offer robust theoretical frameworks for spatiotemporal knowledge representation, their practical adoption in resource constrained or decentralized architectures introduces prohibitive computational overhead. Implementing a comprehensive RDF stack necessitates the integration of complex, third party dependencies, including dedicated RDF parsers, heavy-weight triplestores, and specialized SPARQL query engines [118]. In execution environments that demand localized processing, offline capabilities, or P2P data synchronization at the edge, relying on such an intensive infrastructure rapidly becomes a systemic bottleneck. The processing cost, memory footprint, and battery consumption required to parse and query serialized RDF graphs detract significantly from core application performance, dictating the need for a radically less resource-intensive approach.

Consequently, we propose a *GeoJSON-based representation* to encapsulate the necessary spatial ontologies and field entities. GeoJSON provides a highly lightweight, native serialization format that is inherently parseable by any application without requiring external semantic querying engines. This is the simplest and most resource inexpensive approach, drastically reducing computational latency while facilitating seamless integration with standard web-based geospatial interfaces and applications for real-time field data acquisition.

Furthermore, the structural representation of this spatial data must carefully account for the challenges described in Section 4.5.1. The most conventional optimization in spatial modeling is the adoption of a strict topological vector structure, where adjacent polygons (e.g., neighboring forestry zones) perfectly share

common boundary edges in the database to eliminate redundancy. However, we argue that a “spaghetti” vector model – where adjacent polygons maintain distinct, independent edges that are essentially listed twice – is fundamentally more appropriate for modeling environments reliant on human observation. In a dynamic context, relying on strict topology artificially enforces consensus where none may exist, silently stripping away the inherent uncertainty introduced by human inputs.

By intentionally adopting a spaghetti model, we preserve the diverging, potentially overlapping boundary inputs provided by different users (Figure 4.3) for the exact same geographical area. Although this structural redundancy marginally increases localized storage and processing requirements, it transforms human input discrepancies from a data topology error into a valuable stochastic asset that will be used by the algorithm presented in Chapter 5. These overlapping, non-merged boundaries can subsequently be aggregated and analyzed probabilistically, allowing the underlying framework to explicitly quantify and leverage this spatial uncertainty rather than discarding it through premature topological smoothing.

4.5.3 Formulation of the Minimal Ontology

In order to determine the fundamental components and semantic entities of the proposed ontology, an extensive study of the relevant literature was conducted to investigate the mechanisms and environmental conditions governing wildfire behavior and propagation. This analysis focused on identifying the critical factors that influence wildfire ignition, spread dynamics, intensity, and evolution, with the objective of formally representing them within the ontological model ultimately used to be consumed by the simulation software. The wildfire behavior is determined by the complex interaction of multiple environmental, geographical, meteorological, and anthropogenic parameters. These include, among others, vegetation and fuel characteristics, terrain morphology, atmospheric conditions, and human activities and interaction. The identification and categorization of these factors are essential for defining the corresponding ontological entities, relationships, attributes, and geospatial primitives required by the simulation framework.

Figure 4.4 illustrates the principal factors involved in the wildfire propa-

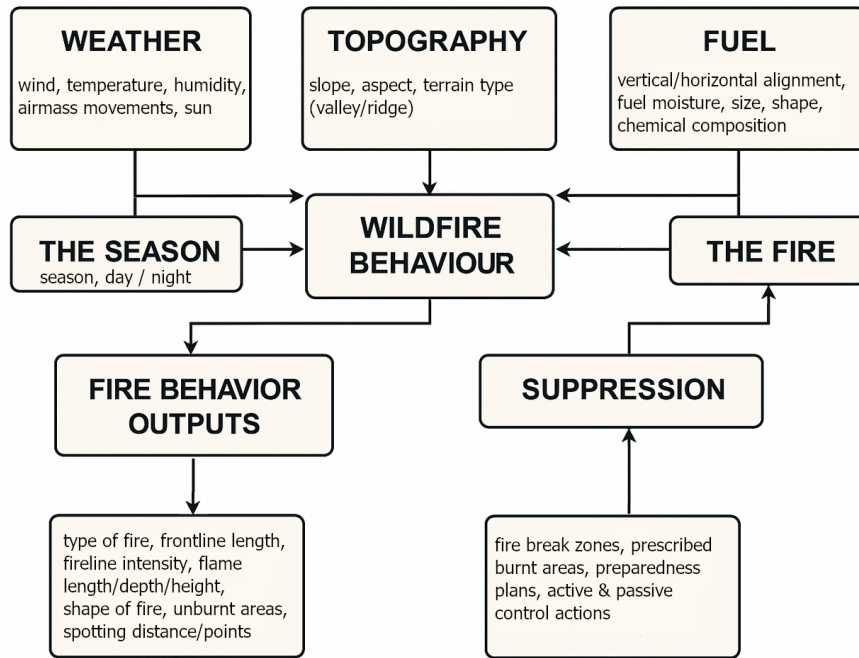


Figure 4.4: High-level representation of factors contributing to wildfire behavior. Adapted from [119].

gation mechanism, and their relationship [119]. These factors are used to derive the minimum set of classes and properties to formulate the proposed *WildFire Simulation Ontology* (WSO) (Figure 4.5).

However, determining the minimum set of parameters, measurements and observations required is heavily dependent on the wildfire simulation model used, with each model being applicable to certain conditions (e.g., wind tolerance thresholds, applicable fuel types, crown fire behavior, etc.). That mutual dependency between input measurements and applicable models raises the need to collect as much data as possible, starting with a limited subset and re-running the simulation when more data are available.

Table 4.2 presents a structured overview of the ontology’s principal classes, their associated datatype and object properties, and the geometry primitives required to spatially ground each class within the simulation domain. For each property, two additional dimensions are reported in the context of near-real-time wildfire simulation: the data acquisition modality – denoting whether the value is

obtained through direct field observation, derived from third-party data sources or computed indirectly – and the temporal variability, which characterizes the rate at which the property value changes and, consequently, the frequency with which it must be updated to maintain simulation fidelity.

Table 4.2: WildFire Simulation Ontology (WSO): properties, geometry primitives, acquisition modality, and temporal variability. Adapted from [21].

Class	Properties	Geometry	Data acquisition	Units	Time variable
Weather	Wind direction	point	In field	degrees	extremely
	Wind speed	point and elevation	In field	m/s	extremely
	Temperature	point	In field / forecast	degrees	very
	Humidity	point	In field / forecast	%	very
	Precipitation	point	Historic data	mm	moderate
	Day/Night	-	-	-	moderate
	Season	-	-	-	no
	Air mass stability	point	In field	stable / unstable	very
Forestry fuel	Fuel type	polygon	In field / Satellite data	-	no
	Fuel moisture	polygon	In field / simulation	%	very to moderate
	Dead/live fuel	polygon	In field	tn/ha	moderate
	Understory / overstory composition	polygon	In field	-	moderate
	Fuel shape (diameter, height)	polygon	In field / Satellite data	-	moderate
	Vertical alignment	polygon	In field / Satellite data	-	moderate
	Horizontal alignment	polygon	In field / Satellite data	-	moderate
	Compactness	polygon	In field / Satellite data	tn/ha	moderate
	Chemical composition	polygon	In field / Satellite data	-	moderate
Topography	Slope	points, elevation	Satellite data	%	no
	Aspect	polygon	Satellite data	degrees	no
	Terrain type	polygon	Satellite data	valley/ridge/open	no
Fire behavior	Ignition point(s)	point	In field	-	no
	Type of fire	polygon	In field	crown/ground/surface	very
	Speed and direction of spread	point	In field / simulation	m/s, degrees	extremely
	Frontline length	line	In field / simulation	m	extremely
	Frontline intensity	line	Simulation	kW/m	extremely
	Flame length	point	In field / simulation	m	extremely
	Flame height	point	In field / simulation	m	extremely
	Flame depth	point	In field / simulation	m	very
	Shape of fire	polygon	In field / simulation	-	very
	Spotting	point point	In field / simulation	-	very
	Spotting distance	point	In field / simulation	m	very
	Unburned areas	polygon	In field / simulation	km ²	moderate
Historic knowledge	Burned areas (past fire)	polygon	In field / historic data	km ²	moderate
	Prescribed burned areas	polygon	historic data	km ²	moderate
	Firebreak zones	line	historic data	m	moderate
Suppression activities	fireline	line	In field	m	very
	attack points (land crews)	point	In field	-	very
	attack points (aerial)	point	In field	-	very

Chapter 5

Geometric Primitive Analysis in Noisy Crowdsourced Data

5.1 Geospatial Uncertainty in Crowdsourced Inputs

A critical issue in simulating wildfires is that we often have to rely on incomplete or outdated information. For example, our knowledge about a remote wild forest at a specific time may be based on satellite observations or earlier aerial surveys that do not provide detailed information on the types and state of vegetation; even this information may become outdated as the landscape evolves over time. When a wildfire is in progress, more detailed and up-to-date information is gathered by eyewitnesses, firefighters, drones and aircraft converging in the area. Although such observations can help simulations to be updated with the most recent data, they are in themselves often inaccurate. For example, different firefighting crews may provide conflicting estimates of the fire front or where one type of vegetation stops and another starts, due to their different vantage points and the uncertainty surrounding such an evolving phenomenon.

Additionally, these crews may experience limited connectivity in forestry or other remote areas; consequently, they cannot rely on unique class instance identifiers (e.g., for a specific vegetated area). As a result, the same area may be

modeled multiple times across different time windows by different users, with no straightforward mechanism to determine whether these observations refer to the same area or how to merge them into a plausible representation of reality.

5.2 From Crowdsourced Observations to Simulation Inputs

The near real time simulation paradigm pursued in this dissertation depends critically on the continuous ingestion of field observations submitted by users with diverse backgrounds and expertise, who are geographically distributed across different areas of the wildfire event. Each observation, serialized as a GeoJSON instance conforming to our *WildFire Simulation Ontology* (WSO) (Chapter 4), constitutes an independent, potentially partial, and inherently noisy description of a shared physical reality (e.g., a burning fireline, a fuel composition of a forestry area, a topographic feature). Before such observations can be consumed by a simulation engine, they must pass through a structured data integration pipeline that resolves the heterogeneity introduced by multiple user sources, links records describing the same entity, and produces a unified, noise-reduced representation of the monitored environment. The integration pipeline adopted in this work [99, 121] comprises three sequential components: Schema Mapping, Record Linkage and Data Fusion (Figure 5.1).

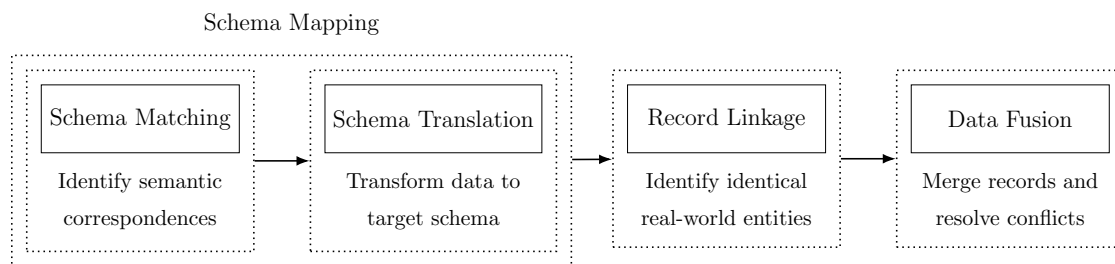


Figure 5.1: Data integration process for heterogeneous and noisy user observations.

1. **Schema Mapping:** This is a two-stage process that can be further divided to Schema Matching and Schema Translation. *Schema matching* is the pro-

cess of identifying semantic correspondences between elements of the schemas being integrated, while *Schema translation* refers to the process of deriving a transformation function between one schema and another [99]. A distinguishing characteristic of the proposed framework is that, by mandating WSO compliant GeoJSON as the accepted submission format, the schema matching and schema translation stages described above are resolved by architectural design rather than at runtime.

2. **Record Linkage:** This component addresses the problem of identifying, across a set of observations, those records that describe the same real-world entity. In the wildfire simulation context, the most critical linkage targets are co-referent observations of the same physical feature: two field crews independently reporting the same forestry fuel polygon, two observers reporting the same fireline, or two measurements capturing the same polygonal feature using slightly different vertex positions and resolution. Without record linkage, each such pair of observations would be treated as two distinct simulation inputs, introducing noise, conflicts, and a systematic overestimation of the number of distinct fire features.
3. **Data Fusion:** This component is the final integration step, in which the co-referent records identified by the linkage phase are combined into a single, accurate representation of each real-world entity. In a crowdsourced setting, fusion must simultaneously address two distinct challenges: noise elimination, arising from individual observations that are erroneous, imprecise, or deliberately falsified; and aggregation, the computation of a representative value from a set of valid but mutually inconsistent observations of the same property (e.g., multiple field crews reporting slightly different boundaries for the same forestry area).

5.2.1 Conflict Handling

The heterogeneity of crowdsourced field data, inevitably results in three distinct categories of conflict within the integration pipeline [121] (Figure 5.2). A

schematic conflict arises when multiple data sources describe the same real-world entity using structurally or terminologically incompatible schemas – for instance, two field submissions recording relative humidity under different property names or value encodings. This category of conflict is inherently mitigated through the adoption of our WSO. An *identity conflict*, occurs when the same physical entity – a forestry fuel polygon, an active fireline – is independently observed and recorded by different field crews without a shared identifier, producing multiple disjoint representations of a single real-world object. This is unavoidable, as the VGI application is designed to operate in offline mode through DTN connectivity; consequently, unique identifiers cannot be assigned to every potential spatial feature. This category of conflict is addressed by the proposed ACO-based algorithm through the novel methodology presented in Section 5.3. Lastly, *data conflict*, emerges after schematic and identity correspondences have been established: even when two records are confirmed to describe the same entity, their measured values may disagree, regardless of whether the measurements correspond to scalar quantities (e.g., a temperature reading) or spatial features such as georeferenced polygons, as a consequence of instrument imprecision, observer error, or the natural spatiotemporal variability of the fire phenomenon.

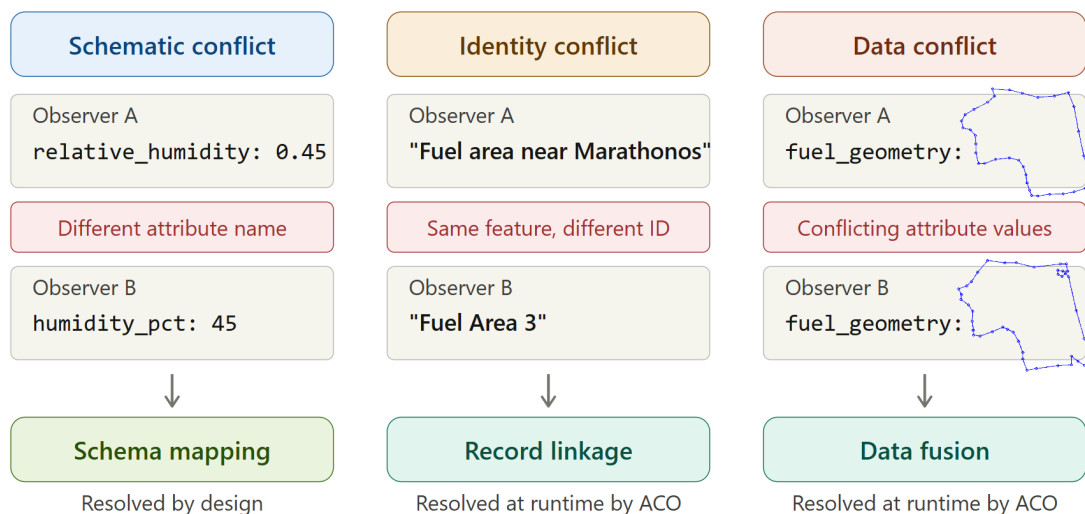


Figure 5.2: Categories of input conflicts addressed by the WSO and ACO.

The three categories of input conflict identified above may be addressed through one of three broad strategies, ordered by increasing computational commitment [122, 121]. The first, *conflict ignorance*, requires no explicit resolution mechanism: all incoming feature observations are overlaid directly onto the simulation grid and the engine selects, for each computational cell, whichever observation it ingests first – either randomly or according to the traversal order of the ingestion algorithm. While trivially implementable, this strategy sacrifices data quality entirely and introduces systematic and unpredictable errors. The second strategy, *conflict avoidance*, operates proactively by eliminating the conditions under which conflicts arise. As discussed, the adoption of WSO results in a unified shared ontological schema prior to transmission, thereby neutralizing schematic conflicts at the architectural level and rendering runtime schema matching unnecessary. The third strategy, *conflict resolution*, actively resolves conflicting observations and is further subdivided, following Bleiholder and Naumann [122], into *instance-based* and *metadata-based* approaches. Instance-based resolution derives decisions exclusively from the data values and geometries of the conflicting records themselves. Metadata-based resolution additionally exploits auxiliary information such as source reliability scores, sensor accuracy ratings, user authority levels, or observation timestamps to weigh competing values during fusion. The ACO-driven pipeline presented in this dissertation operates at both levels: in the record linkage phase, the algorithm co-references conflicting feature observations by evaluating geometric and semantic similarity between pairs, identifying those that describe the same real-world entity; in the subsequent data fusion phase, pheromone-derived confidence weights, based on metadata timestamps, are applied to aggregate co-referenced observations into a single, noise-reduced simulation input. Figure 5.3 presents a taxonomy of conflict handling mechanisms within the proposed WSO-ACO methodology.

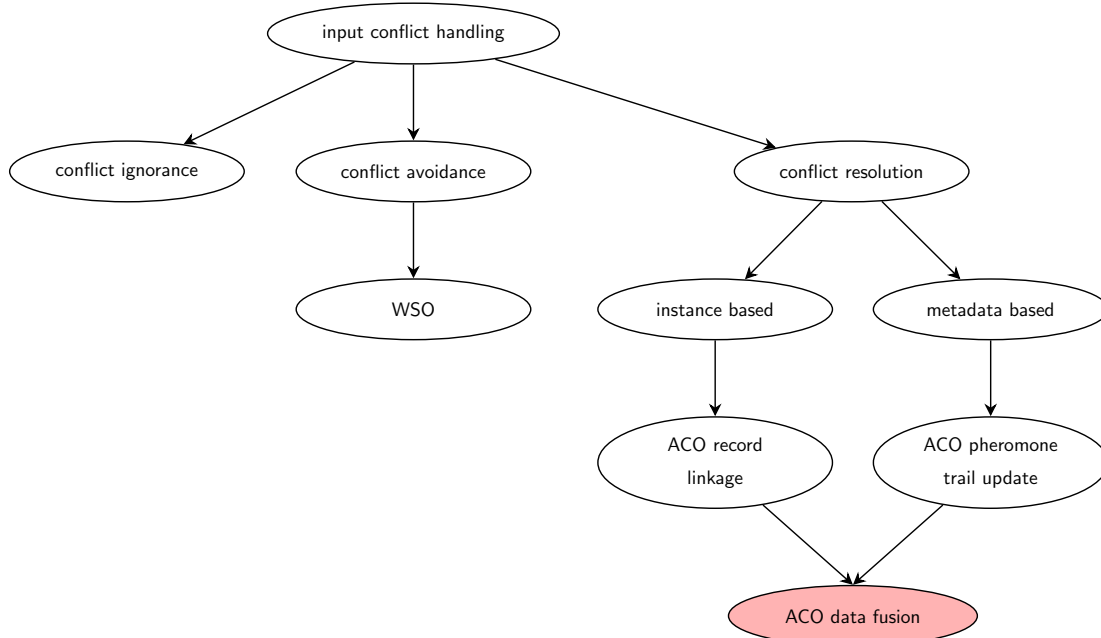


Figure 5.3: User input conflict taxonomy handling within the proposed WSO-ACO methodology. Inspired by [122].

5.3 An Ant Colony Optimization Algorithm for Integrating User Inputs

The ACO family of algorithms are inspired by the indirect communication mechanism utilized by ants to coordinate collective foraging behavior [123]. When a promising source of food is discovered by an ant, it lays a pheromone trail during its return trip to its nest. That pheromone trail is detectable by other members of the colony, guiding them towards the food source. If the food source is preferable by the ants, the trail between it and their nest will soon be reinforced by deposition of additional pheromone trails left by other members of the colony. Conversely, if the food source is exhausted or superior alternatives emerge, the existing pheromone trails will be left to gradually evaporate.

ACO algorithms implement digital ants which explore the search space of complex optimization problems. Typically, when a promising solution is found, a “pheromone trail” is deposited to guide the rest of the “ants” to converge on that

point of the search space. To escape local optima, each ant probabilistically selects its next move based on pheromone levels and other heuristic parameters, thereby ensuring that every region of the search space has a nonzero probability of being explored. The robustness and efficiency of ACO methods is well documented in the literature, and as a result, numerous ACO variants have been developed for a broad spectrum of optimization problems, both discrete and continuous [124, 125].

5.3.1 A Baseline ACO Approach

In this work, we extend an existing ACO algorithm for solving the contour correspondence problem [126] to guide artificial ants to probabilistic classify and reconstruct geometric primitives (e.g., polygons and polylines), mimicking collective human judgment over crowdsourced geometric features within the simulation space. Formally, given two contours represented as ordered point sets I and J ($|I| \leq |J|$), the goal is to find a mapping $\pi : I \rightarrow J$ minimizing the *Quadratic Assignment Problem* (QAP) [127] objective (Equation 5.1):

$$QAP(\pi, R, X, I, J) = \underbrace{(1 - \nu) \cdot S(\pi, R, I, J)}_{\text{shape descriptor term}} + \underbrace{\nu \cdot X(\pi, I, J)}_{\text{proximity term}} \quad (5.1)$$

The fitness of a mapping π is evaluated by minimizing a weighted objective function combining two terms. The shape descriptor term S measures the average dissimilarity between descriptors at matched point pairs $(i, \pi(i))$, where each point i carries a *shape context descriptor* [128] R_i — a histogram of surrounding contour point positions binned by distance and angle relative to i . The proximity term X measures the distortion of pairwise geodesic distances [129]: for every pair of points i, i' on contour I , it penalizes assignments where the geodesic distance $D_I(i, i')$ is not preserved by their matched counterparts $D_J(\pi(i), \pi(i'))$ on contour J , with Gaussian weights suppressing violations between points that are already far apart on I . The parameter $\nu \in [0, 1]$ controls the tradeoff between the two terms.

Following the typical ACO graph paradigm [124], the algorithm formulates a complete directed bipartite graph $G = (V, E)$, where the vertex set $V = I \cup J$ is composed of the two point sets to be matched (Figure 5.4). The directed edge set

E fully connects the two point sets in both directions, yielding two conceptually distinct subsets: edges directed from I to J , which carry pheromone levels τ_{ij} initialized uniformly to τ_0 , and edges directed from J to I carry no pheromone and serve only as structural transition paths (black dotted arrow).

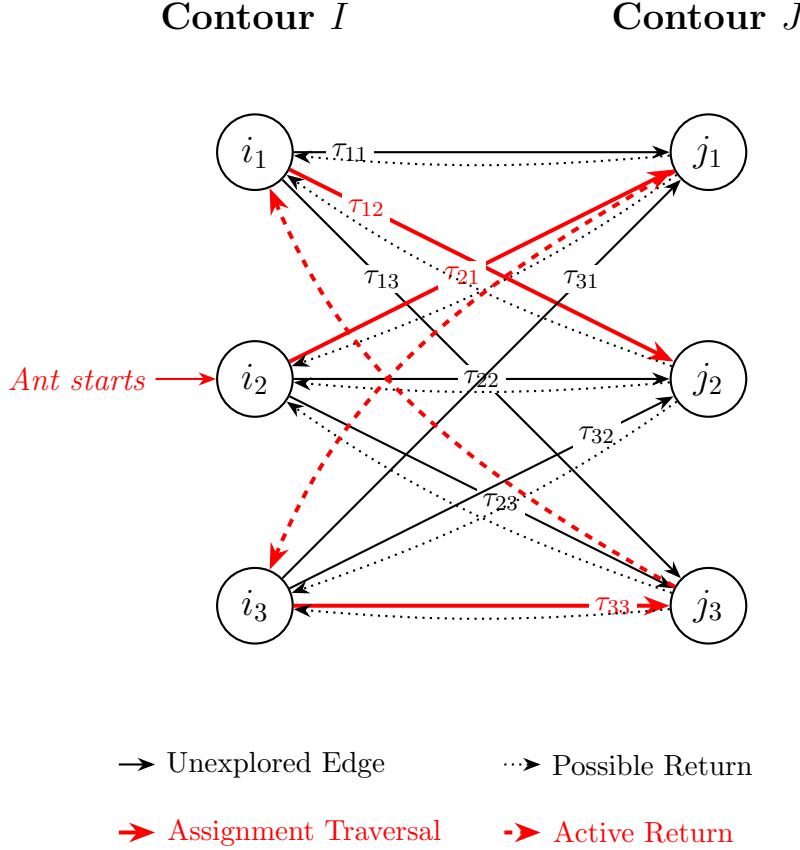


Figure 5.4: Bipartite graph $G = \{I \cup J, E\}$ topology for the QAP-formulated contour-matching ACO algorithm. Inspired by [126].

Each ant traversal over G constitutes a candidate correspondence. An ant begins from a randomly selected vertex in I and proceeds by alternating between the two parts of the graph G . When traversing an $I \rightarrow J$ edge, the ant commits to an assignment $\pi(i) = j$, adding that pair to the solution (red solid arrow). When traversing a $J \rightarrow I$ edge, the ant selects the next unassigned vertex in I to process from the remaining candidates (red dotted arrow). The traversal continues until every vertex in I has been assigned, at which point the full correspondence π has been extracted during the traversal of G .

The k^{th} artificial ant selects a target vertex $j \in J$ when departing from vertex $i \in I$ in a stochastic manner, governed by the same probability p_{ij}^k formulated in the original ACO framework (Equation 5.2):

$$p_{ij}^k = \frac{\alpha\tau_{ij} + (1 - \alpha)\eta_{ij}}{\sum_{l \in N_i} [\alpha\tau_{il} + (1 - \alpha)\eta_{il}]}, \quad (5.2)$$

where τ_{ij} denotes the pheromone deposited in edge (i, j) , η_{ij} is the heuristic information (how desirable is the solution component) expressed similarly to the QAP cost function (Equation 5.1), $N_i = \{l \in J : (i, l) \in E\}$ and $0 \leq \alpha \leq 1$ regulates the relative influence of learned pheromone versus immediate heuristic information.

After all ants complete their traversals in one iteration, pheromones are updated in two steps. First, a pheromone evaporation is conducted at all edges $I \rightarrow J$ according to Equation 5.3:

$$\tau_{ij} = (1 - \rho)\tau_{ij}, \quad (5.3)$$

where $0 \leq \rho \leq 1$ is a constant named *evaporation rate*. This prevents indefinite accumulation and allows the colony to forget poor early solutions. Then, a pheromone deposition is applied only to edges used in this iteration's matchings according to Equation 5.4:

$$\tau_{ij} = \tau_{ij} + \sum_{k=1}^m \Delta\tau_{ij}^k, \Delta\tau_{ij}^k = \frac{\delta}{C_k}, \quad (5.4)$$

where δ is the *deposition parameter* and C_k is the cost of the candidate solution. A minimum pheromone level τ_{min} is maintained on all edges to prevent complete elimination of any traversal, ensuring the exploration of the search space.

5.3.2 VGI Fusion via Modified Ant Colony Optimization

Let $\mathcal{D} = \{(g_k, t_k)\}_{k=1}^N$ denote the set of crowdsourced submissions, where each entry consists of a georeferenced geometric primitive g_k , represented as a GeoJSON polygon, and an associated timestamp t_k . We have already assumed that multiple contributors may independently submit the (almost) same environmental feature at overlapping or proximate times during the fire spread, so $\mathcal{D}$ may contain

redundant observations that must be identified and reconciled before any temporal analysis can be conducted.

The process begins with a *spatio-temporal clustering step* applied to $\mathcal{D}$. To extract the time windows during which coherent changes to the wildfire geometric primitives occurred, a density-based clustering algorithm, specifically *Density-Based Spatial Clustering of Applications with Noise* (DBSCAN) [130], is applied to the timestamp sequence $\{t_k\}_{k=1}^N$. DBSCAN is specifically selected for this task as it does not require a prior specification of the number of clusters and is robust to noise, naturally discarding isolated submissions that do not correspond to any collective observation event. The output is a partition of $\mathcal{D}$ into $\mathcal{C}$ temporal clusters $\{W_c\}_{c=1}^C$ where each cluster W_c defines a time window $[t_c^{\text{start}}, t_c^{\text{end}}]$ within which a set of submissions is assumed to describe (roughly) the same state of the wildfire environment.

Then, DBSCAN is applied again, but this time to the spatial domain g_k inside cluster c , yielding a two-dimensional cluster structure W_{ij} , where the outer index $i = 1, \dots, C$ enumerates distinct temporal windows and the inner index $j = 1, \dots, S_i$ enumerates spatially coherent sub-clusters within each such window. The spatial sub-clustering within each time window therefore ensures that the ACO correspondence algorithm is applied only to geometrically proximate *and* physically coherent observations. As with standard DBSCAN, submissions that do not satisfy the density reachability criterion in either dimension are treated as noise and discarded.

For each spatio-temporal sub-cluster W_{ij} , the submissions are partitioned into the two sets of points required by the ACO correspondence framework. A single GeoJSON file input is probabilistically selected from W_{ij} to form the reference set I_{ij} . The remaining submissions within the same W_{ij} sub-cluster collectively form the target set J_{ij} , whose contour points are aggregated into a unified point set. This probabilistic selection of I_{ij} avoids introducing a systematic bias toward any particular contributor’s delineation, treating all submissions as equally credible. The constructed sets I and J , are supplied to the ACO-based contour correspondence algorithm. The algorithm constructs a mapping $\pi^* : I_{ij} \rightarrow J_{ij}$ by

minimizing the QAP objective (Equation 5.5):

$$\pi^* = \arg \min_{\pi} [(1 - \nu)S(\pi, R, I_{ij}, J_{ij}) + \nu X(\pi, I_{ij}, J_{ij})] \quad (5.5)$$

The resulting assignment π^* identifies, which geometric primitive in J corresponds to the one in I . It should be noted that the optimal assignment π^* will map points from *different* user originated primitives inside J to the randomly selected I point set. This serves as the final data fusion stage, in which inputs from multiple submitters are aggregated into a single geometric primitive. The manner in which human behavior influences component selection is incorporated into the modified pheromone update function, described in Algorithm 4. A graphical outline of the fusion process in the context of the near-real-time simulator is given in Figure 5.5. For a more comprehensive description of the proposed algorithm, refer to Algorithm 1. The *speed_metric* variable is specific to the domain of the geometric primitive and may represent, for example, changes in the area or length of two shapes. Its purpose is to extract additional information regarding fire spread, such as the rate of spread. An example formulation of the `CALCULATEDIFF()` function is presented in Algorithm 2. The technical outline of the multi-stage fusion pipeline is depicted in Figure 5.6.

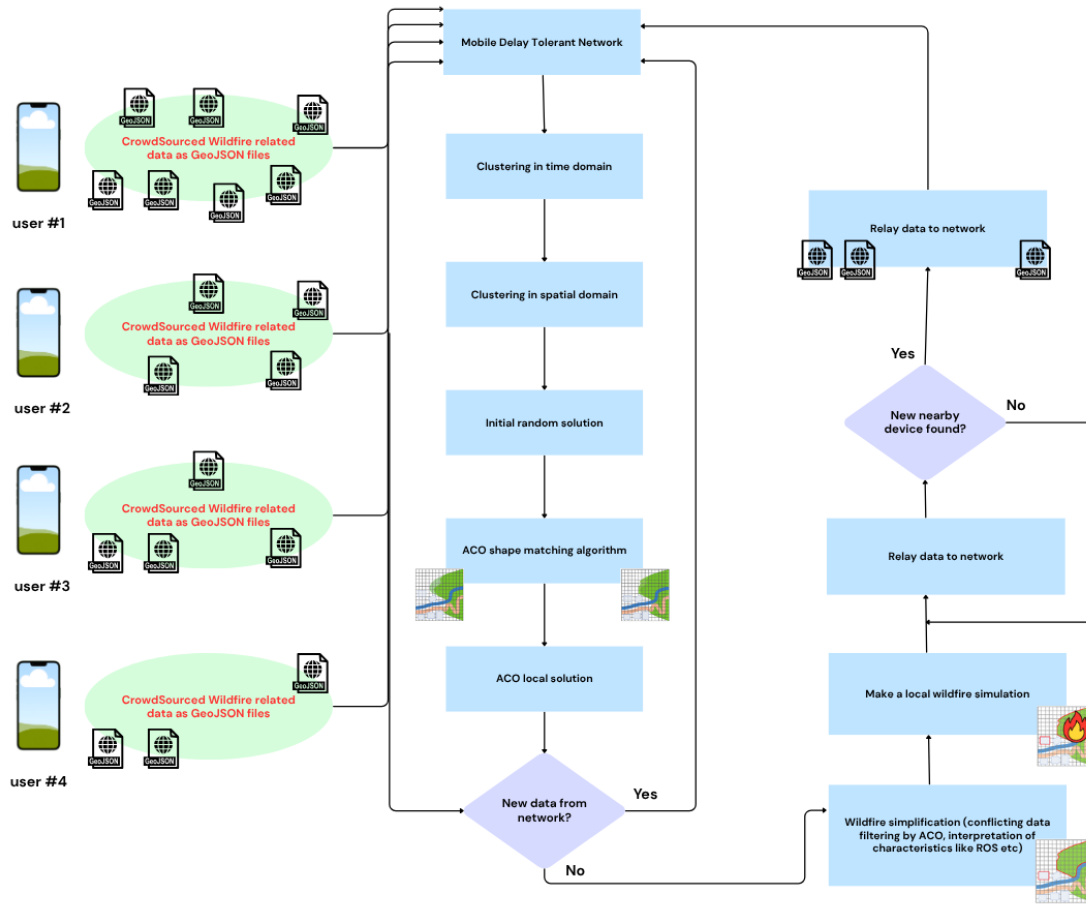


Figure 5.5: Flowchart of a mobile wildfire simulator exchanging crowdsourced geospatial data for progressive simulation in environments lacking input data. Adapted from [22].

Finally, this approach introduces a limitation worth noting, namely that the two level clustering process creates additional challenges associated with the separate density and distance thresholds in the temporal and spatial dimensions. These thresholds can lead to over segmentation, particularly in scenarios where a single continuous fire front generates submissions that are spatially dispersed yet physically connected.

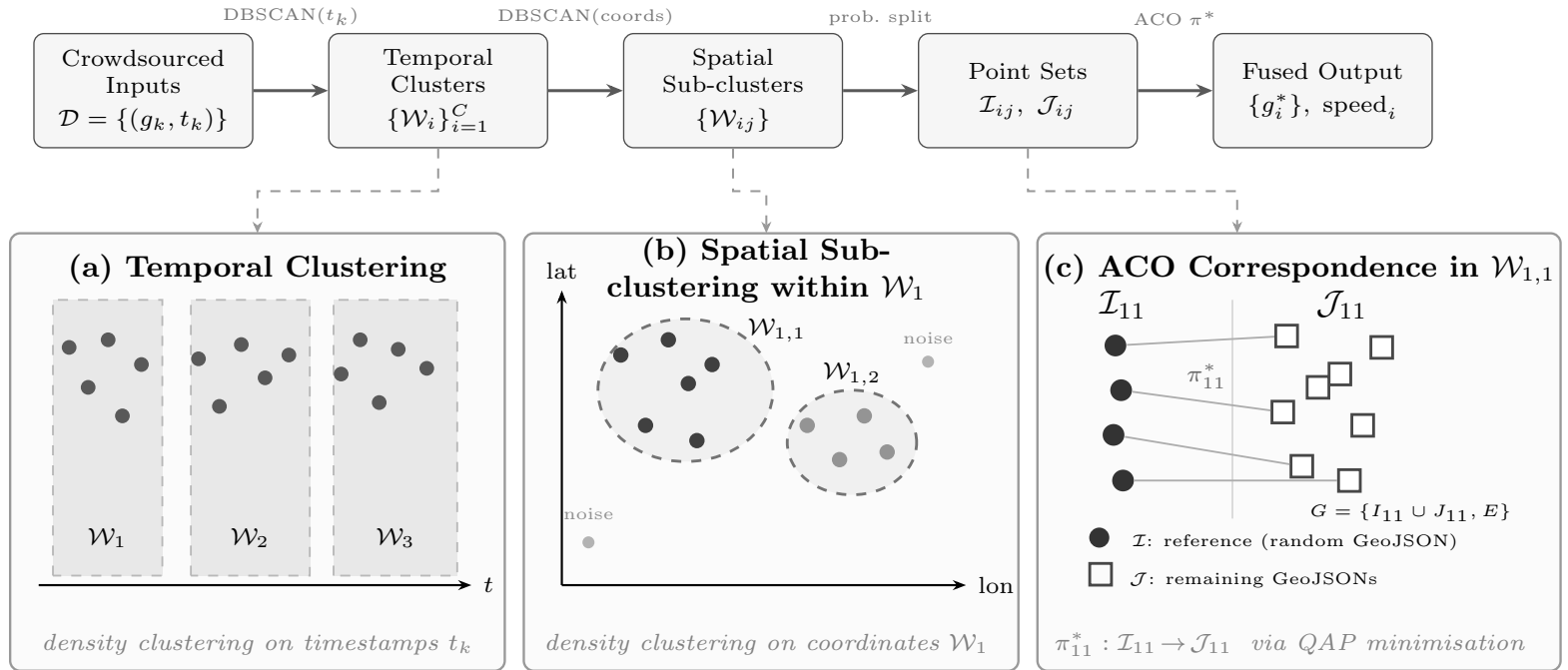


Figure 5.6: Technical overview of the ACO-based fusion pipeline for crowdsourced simulation inputs.

Algorithm 1 Spatiotemporal clustering algorithm

Require: user inputs **jsons**[]

```

1:                                     ▷ Time and spatial clustering of jsons
2: cluster[][]  $\leftarrow$  CLUSTER(jsons.timestamps, jsons.geometry.coordinates)
3: for i = 1 to SIZE(cluster) do                                     ▷ Time domain
4:   for j = 1 to SIZE(cluster[i]) do                               ▷ Spatial domain
5:     I  $\leftarrow$   $\emptyset$ 
6:     J  $\leftarrow$   $\emptyset$ 
7:     I  $\leftarrow$  RANDOM(jsons[cluster[i][j]])
8:     J  $\leftarrow$  J  $\cup$  jsons[cluster[i][j]]  $\setminus$  I
9:     ▷ matching[i][j] is the match that minimizes cost for cluster i, j
10:    (cost, matching)[i][j]  $\leftarrow$  PERFORM_ACO(I, J)
11:    simplified_coordinates[i]  $\leftarrow$  SIMPLIFY(matching[i][j])    ▷ SIMPLIFY()
    returns interpolated points over the matching
12:  end for
13:  speed_metric[i]  $\leftarrow$  CALCULATEDIFF(matching[i], matching[i + 1], (cluster[i].timestamp, cluster[i + 1].timestamp)
14: end for
15: return simplified_coordinates, speed_metric

```

Algorithm 2 Outline of CALCULATEDIFF() function

```

1: function CALCULATEDIFF(matching1, matching2, time1, time2)
2:   speed  $\leftarrow$   $\frac{\text{EVALUATE\_SHAPE}(\textit{matching}_1, \textit{matching}_2)}{\textit{time}_2 - \textit{time}_1}$ 
3:   ▷ EVALUATE_SHAPE() may return the area difference, or length difference
   between two shapes
4:   return speed
5:   ▷ A speed close to zero implies a stationary object submitted by users
6: end function

```

The PERFORM_ACO() function is adapted from [126] and its outline is given in Algorithm 3.

Algorithm 3 Outline of PERFORM_ACO() function

```

1: function PERFORM_ACO( $I, J$ )
2:    $pheromone\_matrix \leftarrow \text{INIT\_PHEROMONE\_MATRIX}(\mu)$ 
3:    $best\_solution \leftarrow \emptyset$ 
4:    $cost \leftarrow \infty$ 
5:    $G \leftarrow \text{INIT\_SOLUTION}(I, J)$        $\triangleright$  Construct a feasible matching between
      points in  $I$  and  $J$ 
6:   for  $i = 1$  to  $N_{iterations}$  do
7:     for  $j = 1$  to  $N_{ants}$  do
8:        $temp \leftarrow \text{CONSTRUCTMATCHING}(G, pheromone\_matrix)$ 
9:        $\triangleright temp$  is a  $1 \times \text{SIZE}(I)$  array
10:       $\triangleright temp[k] = m$  means the  $k$ -th point of  $I$  matches with the  $m$ -th
        point of  $J$ 
11:    end for
12:     $\text{UPDATEPHEROMONE}(pheromone\_matrix, temp)$ 
13:    if  $\text{COST}(temp) < cost$  then
14:       $cost \leftarrow \text{COST}(temp)$ 
15:       $best\_solution \leftarrow temp$ 
16:    end if
17:  end for
18:  return ( $cost, best\_solution$ )
19: end function

```

5.3.3 Why an ACO-Based Correspondence Solver?

As described in Section 5.3, the fusion stage of the ingestion pipeline resolves a noisy set of clustered, crowdsourced polygon submissions describing the same fire-related segment of the simulation space into a single consensus polygon. One clustered submission is selected as the reference vertex set $I = \{i_1, \dots, i_n\}$; the vertices of all remaining submissions in the cluster are pooled, without regard to which submission or contour position they originated from, into a candidate

set $J = \{j_1, \dots, j_m\}$, typically with $m \gg n$ (Figure 5.7). The fusion step then requires a correspondence $\pi : I \rightarrow J$ from which the consensus polygon is obtained by averaging each $i \in I$ with its matched vertex in J . This section justifies why π is computed with an ACO metaheuristic [22, 126] rather than with an exact combinatorial solver.

A correspondence that only minimizes a per-vertex descriptor dissimilarity,

$$S(\pi) = \frac{1}{|I|} \sum_{i \in I} d(R_i, R_{\pi(i)}), \quad (5.6)$$

is separable: the cost of assigning $i \rightarrow \pi(i)$ does not depend on how any other vertex is assigned. An objective of this form is a *Linear Assignment Problem* (LAP) and is solvable exactly in polynomial time, e.g. with the Hungarian algorithm in $O(n^3)$ [131]. However, $S(\pi)$ alone is not an adequate fusion objective for our setting. Because J is pooled across multiple independently traced submissions – each contributed at its own starting vertex, tracing direction, and digitization resolution – J carries none of the contour ordering that would otherwise let us bound the search to order-preserving assignments; and because submitters vary in tracing precision, vertices are frequently missing or unevenly spaced along the true fire-front boundary. Under these conditions, a purely descriptor-driven LAP solution routinely produces geometrically incoherent correspondences: vertices matched out of spatial sequence, producing self-intersecting or locally inverted fused segments, which is unacceptable input to a downstream fire-spread simulator that assumes a simple, non-self-intersecting perimeter. Avoiding this requires the addition of an explicit proximity-preservation term that penalizes changes in the relative distances between matched vertices, combined with $S(\pi)$ into a single weighted objective (Equation 5.5). That formulation allows for coupling every pair of assignment decisions $\pi(i)$ and $\pi(i')$: the cost contribution of matching i to a candidate j can no longer be evaluated independently of where i' is matched. This coupling is precisely what makes $\text{cost}(\pi)$ a QAP rather than a LAP – an NP-hard combinatorial optimization problem, with no known polynomial-time exact algorithm.

Exact QAP solvers (branch-and-bound, cutting-plane) are, in practice, limited to instances on the order of a few dozen points [132]; a pooled J aggregated

from even a modest number of crowdsourced submitters routinely exceeds this. Restricting the search to order-preserving assignments would recover polynomial-time exact solvability via cyclic sequence alignment [133], but only if contour order in J can be assumed – which, as argued above, pooling precludes by construction. We considered, and rejected, avoiding the pooled formulation altogether by matching I against each submission in the cluster separately and averaging the $N - 1$ resulting pairwise correspondences. This does not eliminate the underlying difficulty; it relocates it, replacing one correspondence problem with $N - 1$ smaller ones that must then be reconciled into a single consensus vertex, and it scales the number of solver invocations linearly with a cluster size that is itself unpredictable and event-dependent. The pooled formulation performs a single fusion pass per cluster regardless of how many submitters contributed to it, which better matches the throughput requirements of a near-real-time ingestion pipeline.

The second consideration is specific to our deployment target assumption: correspondence is computed on-device, on commodity mobile hardware, under a power budget that varies with battery state and thermal headroom. ACO exposes this trade-off directly through two parameters, the number of ants N_{ants} and the number of iterations $N_{iterations}$. After every iteration the pheromone matrix encodes a complete, valid candidate assignment, so the search can be interrupted at any point and still return a usable correspondence. Because the best assignment found so far is retained across iterations, solution quality is monotonically non-decreasing by construction, even though the underlying pheromone updates are themselves stochastic [134]. This lets us select the value of $N_{iterations}$ at runtime. Setting $N_{iterations}$ to lower values allows for an immediate simulation update, while a higher $N_{iterations}$ value can be selected when the device is idle or charging from a battery pack rather than fixing a single, worst-case compute budget for every fusion event. For QAP specifically, exact solvers offer no comparable guarantee: their solution-quality evolution over time is worst-case exponential and instance-dependent, with no explicit mechanism for trading solution quality against a bounded time or energy budget. It should also be noted, that ACO’s graceful degradation effect is an empirically observed property of the pheromone-reinforcement mechanism rather

than a formally proven approximation bound.

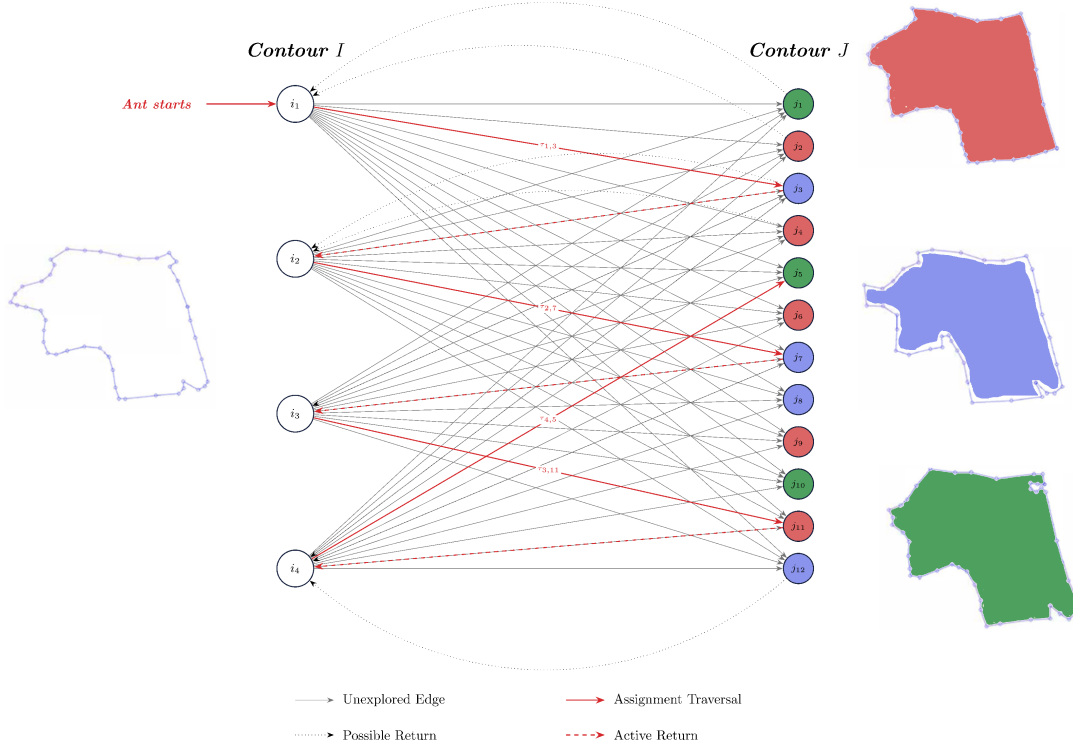


Figure 5.7: Stochastic consensus shape fusion using ACO-based algorithm. The mapping of crowdsourced polygon vertices to graph vertices is indicated by the corresponding colors. To improve graph readability, only a subset of the edges directed from J to I is shown.

Together, these two considerations – the objective’s NP-hardness under the pooled, unordered formulation our *pipeline* requires, and the need for a tunable, interruptible solver under a mobile *power budget* – justify the selection of ACO as a purposeful algorithmic design choice rather than a generic optimization approach.

5.3.4 A Modified Pheromone Update Model

In our context, one source of conflict between user-generated inputs is the evolution of the wildfire itself, which keeps changing the state of the burning area. This implies that more recent observations may better reflect the current state in the field. To incorporate this into our matching scheme, an additional amount

of pheromone is introduced based on the number of solution components associated with more recent timestamps. This encourages the ACO-based algorithm to construct solutions using newer data, which are more likely to reflect the current state of the evolving phenomenon. The proposed modification is not intended to produce matchings with lower cost; rather, it biases the algorithm toward selecting more solution components containing the most up-to-date information.

This is achieved by the addition of the term λ in the pheromone deposition process described in [126]. The term λ boosts the pheromone levels in solution components that contain newly submitted entries following the computation of pheromone deposition presented in the original version of the algorithm. The modified algorithm is presented in Algorithm 4.

Algorithm 4 Outline of UPDATEPHEROMONE() function

```

1: function UPDATEPHEROMONE( $\tau, matching$ )
2:   for  $i = 1$  to SIZE( $I$ ) do
3:     for  $j = 1$  to SIZE( $J$ ) do
4:        $\tau_{ij} \leftarrow \tau_{ij} - evaporation\_rate$ 
5:       if  $\tau_{ij} < \tau_{min}$  then
6:          $\tau_{ij} \leftarrow \tau_{min}$ 
7:       end if
8:        $\Delta \leftarrow \frac{\delta}{COST(matching)}$ 
9:        $\lambda \leftarrow EVALUATE\_RECENT(matching)$ 
10:       $\tau_{ij} \leftarrow \tau_{ij} + \Delta + \lambda$ 
11:    end for
12:  end for
13: end function

```

Function EVALUATE_RECENT() accounts for the number of solution components that are members of the newest submitted entry, multiplied by a coefficient that governs their influence against the traditional computation of pheromone deposition. Its outline is given in Algorithm 5.

Algorithm 5 Outline of EVALUATE_RECENT() function

```

1: function EVALUATE_RECENT(matching)
2:   count  $\leftarrow$  0
3:   for  $i = 1$  to SIZE( $I$ ) do
4:     if matching[ $i$ ]  $\in$  most_recent_timestamped_entry then
5:       count  $\leftarrow$  count + 1
6:     end if
7:   end for
8:   return count * recent_coefficient
9: end function

```

5.4 Experimental Results

To examine the efficiency of the proposed method and its ability to generate a simplified version of the original raw user submissions, we asked a group of users with different knowledge backgrounds to draw several shapes on a map (via Open Street Maps) depicting a wildfire characteristic. The group of users included a participant possessing extensive training and expertise in firefighting methods. The area was located inside a WUI zone in East Attica [135, 136]. An illustration of raw user inputs is given in Figure 5.8 where the same vegetated area is captured by four different contributors. Figure 5.9 depicts the same modeled area (acquired by the first user) submitted through the VGI interface as a map overlay of 36 points defined in the WGS 84 coordinate reference system.

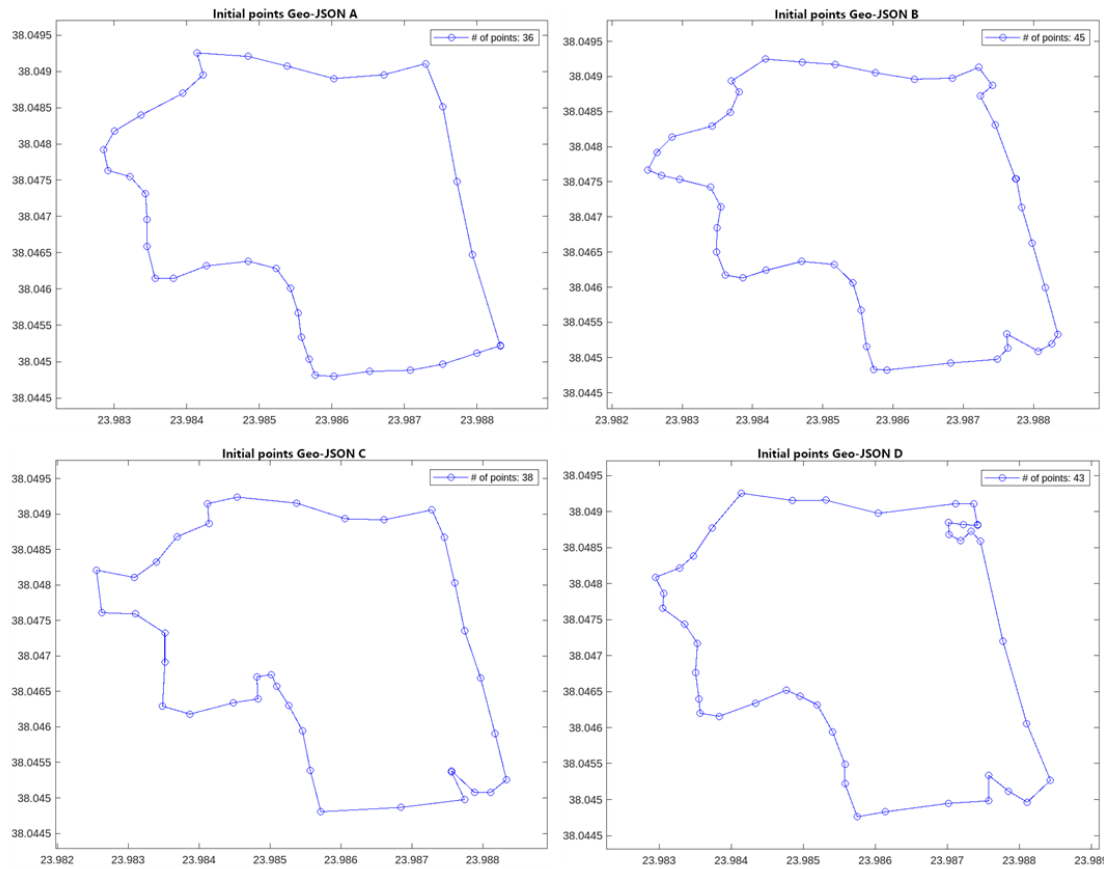


Figure 5.8: The same vegetated area in East Attica as modeled by four users (WGS 84 coordinates). Adapted from [22].

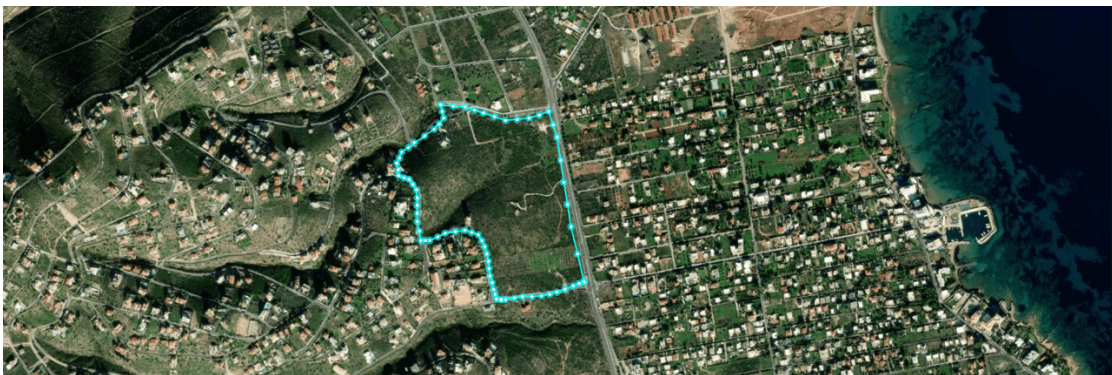


Figure 5.9: A vegetated area in East Attica projected on the map by a volunteer. Adapted from [22].

Given the input provided by four submitters for the same forestry area depicted in Figure 5.8, and assuming for simplicity that all observations belong to

the same c_i temporal cluster, the output of the modified ACO-based data fusion algorithm is the shape illustrated in Figure 5.10. Set I (corresponding to a randomly selected user) is depicted in blue, while set J (representing the remaining users) is shown in red. The ACO algorithm constructs a mapping (illustrated by black lines) between each point in set I and its optimal assignment in set J (Figure 5.10(a)). Subsequently, the user may process only the “compressed” geometric primitive corresponding to the optimal assignment (Figure 5.10(b)). This primitive can be approximated by the midpoint between the corresponding points in sets I and J (or another interpolation method), and may then be redistributed to nearby users through the DTN infrastructure or any other available form of network connectivity. The ACO parameters used are listed in Table 5.1.

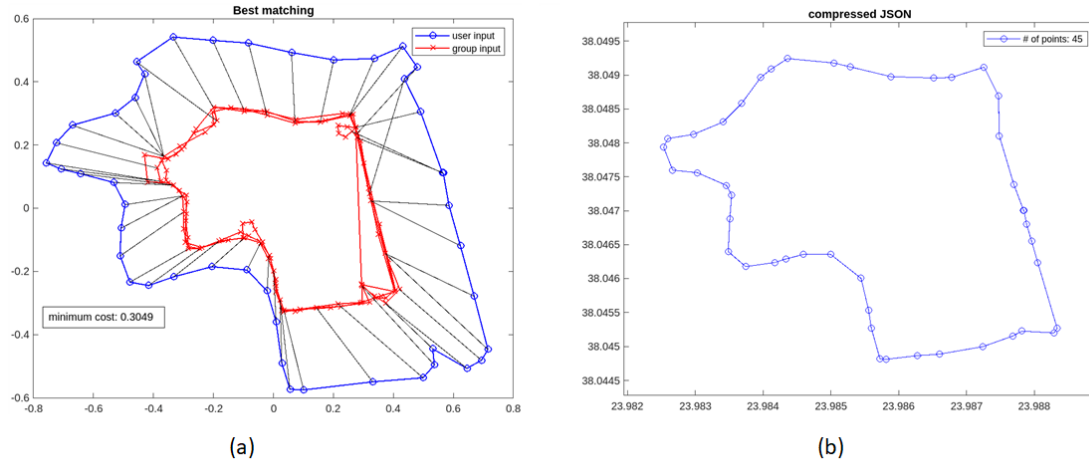


Figure 5.10: (a) Optimal assignment between I (random user) and J (remaining users) sets (b) simplified primitive. Generated by modifying [137] software.

In order to evaluate the performance of the proposed ACO method on additional geometric primitives, such as LineStrings (e.g., representing an active fire front or a burn barrier), the same group of volunteers was also asked to draw LineStrings corresponding to the boundary of a past wildfire event. We interpreted these inputs as the active fireline for a hypothetical wildfire event, since it is not practical to acquire this information in a real-world scenario. The firelines acquired by the users is depicted in Figure 5.11.

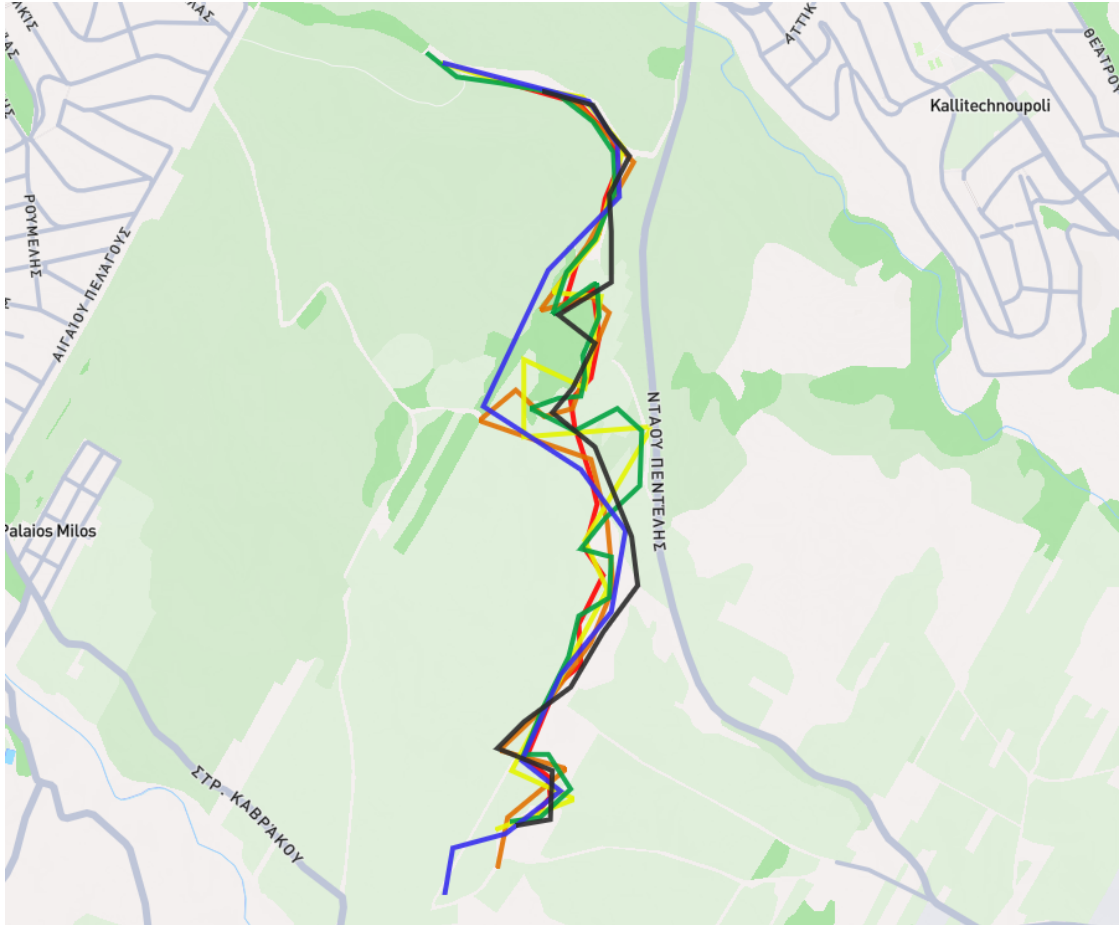


Figure 5.11: A noisy representation of a fireline obtained from the same group of volunteers. Individual submissions are depicted in different colors. Adapted from [22].

An additional test case was conducted, in which the objective was to evaluate the method’s ability to perform *record linkage* (refer to Section 5.2) among instances of user-submitted geometric primitives. We examined different pairwise combinations of polygons and LineStrings to assess the geometric similarity between shapes, using two metrics: the heuristic information used by the proposed ACO algorithm (Equation 5.1) to determine solution cost, and a baseline method consisting of the weighted sum of *Modified Hausdorff Distance* (MHD) [138, 139] and shape area ratio calculation. The parameters used for the ACO algorithm and the baseline method are listed in Table 5.1. Figure 5.12 presents a bar chart of normalized similarity scores obtained by ACO and the baseline method across

polygon pairs of varying vertex count. Although this test is considered a form of benchmarking, the methods are not directly comparable to each other, since the objectives differ, even though they aim to answer the same question: to what extent are two shapes similar? ACO constructs a feasible solution in a non-deterministic way, which is promising according to its objectives (e.g., favoring recent solution components) and tuning parameters. That solution is then processed to simplify the geometry. Therefore, the benchmark serves only as a reference point, illustrating how the baseline method would perform in the absence of the additional objectives considered in this work.

Table 5.1: ACO parameters and baseline weights used in the experiments.

Parameter	Value	Impact
Baseline Method		
MHD similarity weight	0.7	favors shape matching over area matching
Area similarity weight	0.3	favors area matching over shape matching
ACO		
Number of iterations	1000	exploration of search space
Number of ants	1	exploration of search space
Influence of proximity (ν)	0.7	cost objective
Pheromone influence (α)	0.3	exploration of search space
Evaporation rate	0.01	exploration of search space
Minimum pheromone	0.1	exploration of search space
Recent coefficient	0.05	increases pheromone deposition in favor of newer solutions

Although the comparisons were performed across all polygons representing the same area, we also included cases in which a clear drawing bias was observed among volunteer firefighters (e.g., distinctive distortions not present in other participants' annotations). The similarity scores of both methods were found to be quite close. Although ACO generally achieves higher similarity scores than the baseline method in most cases, drawing definitive conclusions remains challeng-

ing due to the level of uncertainty and noise introduced by human submitters. While similarity metrics (such as MHD) are fundamental to addressing our *record linkage* problem, their practical significance is not straightforward to determine. ACO demonstrates a clear efficacy by identifying optimal vertex correspondences between shapes, instead of using a simple approximations between them. Consequently, it derives solution components directly from unprocessed user inputs, yielding results that more faithfully reflect human judgment.

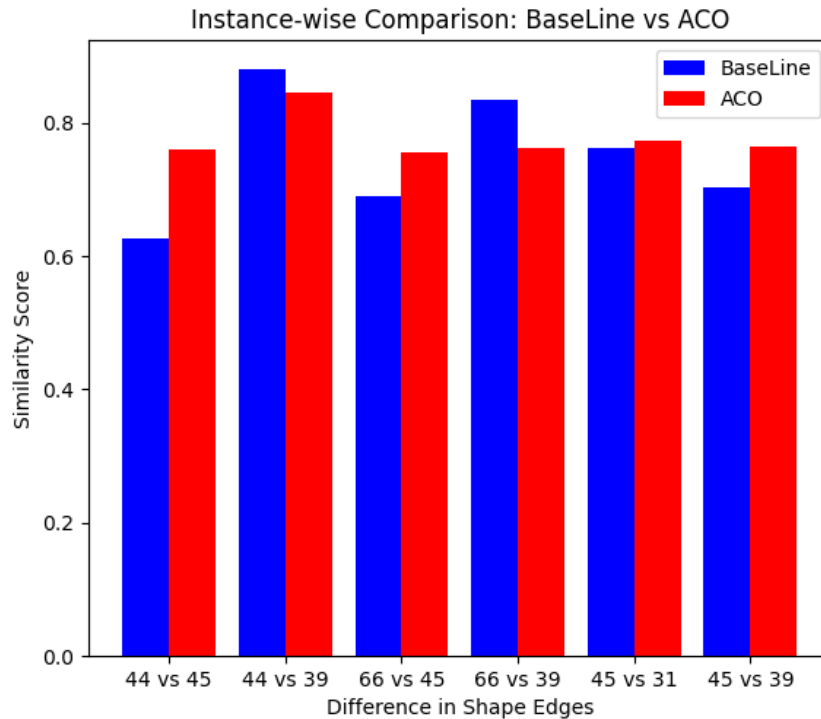


Figure 5.12: Similarity scores of baseline and modified ACO methods for the six instances of polygons with different vertex count. Adapted from [22].

5.5 Result Analysis and Future Extensions

The proposed modified ACO-based algorithm for data fusion in crowd-sourced geospatial data demonstrates that a dynamic, non-deterministic solution to the data fusion and record linkage problem is both feasible and applicable within the pipeline of a near-real-time wildfire simulator. The main advantage of the pro-

posed method lies in its capacity to emulate human decision making through the ant colony analogy, wherein more favorable solutions are preferentially reinforced. Rather than relying on synthetically generated geometric inputs, the algorithm constructs its solution components (geometric primitives) directly from existing, noisy, user submitted raw data.

Further analysis of the results revealed several issues needing investigation. In particular, more effective strategies are needed for handling outliers and accommodating submissions from heterogeneous user groups, including distinguishing between expert and untrained contributors. Moreover, our approach should be evaluated in real-world scenarios with respect to its ability to smooth short-term fluctuations and mitigate temporal noise in crowdsourced inputs. Special attention should be given to the trade-off between recency and reliability, as prioritizing more recent data may inadvertently reduce the influence of older, yet more reliable, submissions by qualified users.

Another important issue is further exploitation of meaningful information related to the nature, behavior, and potential risk exposure of submitters. As user generated data are streamed to mobile nodes, mobility patterns may be inferred not only from the submitted data themselves, but also from the spatial and temporal characteristics of node encounters. Rapid and unpredictable changes in mobility patterns, as well as the emergence of swarms of moving users, may indicate uncontrolled fire outbreaks that trigger evacuation processes or reflect activities associated with modified suppression tactics. Such approaches appear promising and have attracted increasing attention in the literature [140, 141].

Chapter 6

Parameter Fine-Tuning in a CA Wildfire Simulator

6.1 Importance of Parameter Adjustment in CA Wildfire Models

The best argument for CA-based methods is their computational efficiency, which is due to their raster-based architecture that significantly speeds up geometric calculations. As a result, CA are well suited for deployment even on devices with limited computational and power resources, such as portable ones. However, their reliance on empirically tuned local transition rules (refer to Chapter 2) introduces structural and statistical challenges that limit their applicability to general cases of wildfires. In particular, they rely on submodules (such as estimators of wind impact, fuel moisture content, and fuel flammability) that depend on an extensive set of parameters and coefficients, which require user tuning. Moreover, it is extremely difficult for a non-expert user – whether in forest science, wildfire dynamics, or meteorology – to fully comprehend the physical significance and impact of each parameter when conducting a simulation in a real world case, or to distinguish which variables are correlated together. Parameterization and calibration are the Achilles’ heel of these models, making them prone to overfitting, yielding apparently accurate reconstructions of specific fire events, while providing limited

predictive performance outside their calibration domain. In addition, because the effects of the individual parameters are aggregated into a single ignition probability (Equation 2.3), distinct parameter sets may produce similar macroscopic fire patterns, which may lead to ambiguous model interpretations by non-expert users.

One notable limitation of CA-based wildfire methods is their limited ability to accurately reproduce wildfire perimeters [15, 52, 142, 143]. This is mainly due to the finite spatial resolution of the simulation grid, coupled with the inherently discrete representation adopted by the modeling framework. This issue is mainly caused by the assumption that all properties within a single simulation cell are spatially homogeneous (e.g., fuel composition, weather conditions) regardless of their variability inside that cell. While increasing the spatial resolution (i.e., decreasing cell size) can improve spatial consistency, it does not fully address the limitations imposed by discrete cell states. In most CA-based models, each cell is typically restricted to binary conditions (e.g., completely burned or unburned), thereby excluding intermediate states such as partial combustion, re-ignition, or changes in burn status caused by evolving wind direction or fuel moisture conditions.

In the context of the near-real-time simulator examined in this dissertation, the problem is further amplified by both the nature of the task and the characteristics of the input data. We assume that local CA simulation instances executed on users' mobile devices will operate with fragmented and potentially missing inputs, while simultaneously receiving continuously updated data from the DTN layer.

To illustrate this issue, we present an example in Figure 6.1, where we have defined the simulation time step to be 10 min and the cell side to be 100 m. Assuming that there are two types of vegetation with ROS 5m/min and 10m/min, in the simulation time step t , the black cells are completely burned and the red cells are now on fire (probabilistically chosen to catch fire with probability P_{burn}). At the next time step $t + 1$, all the red cells must become completely burned (black), although only for cell 2 it is reasonable to assume that it will be consumed by fire (by oversimplifying and assuming the same front and flank ROS). However, cell 1 will only be partially burned. Even worse, at the same time step $t + 1$, the neighboring cells of cell 1 will be considered as candidates for catching fire,

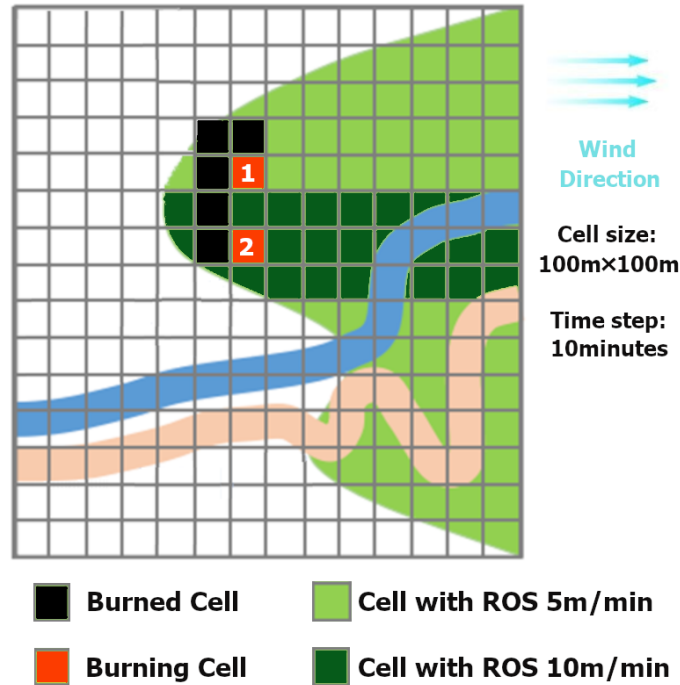


Figure 6.1: Landscape discretization in a CA simulator with two classes of forestry fuels. Adapted from [15].

even if the fire front cannot reach their position yet. A straightforward solution, under the assumption that no dynamic data are fed to the simulator, would be to either decrease the cell size resolution to 50 m or increase the simulation time step to 20 min, although both approaches would introduce additional computational overhead in terms of CPU and memory usage.

This behavior is not unknown to researchers and a wide range of improvements and optimization strategies have been proposed to address it. These approaches span multiple levels of model complexity, ranging from parameter optimization and fine-tuning of existing CA formulations [52, 144], to modified cell shapes [145]. More extensive modifications include the introduction of variable simulation time steps [49, 51, 52, 58], continuous or multi-valued cell states [48, 58], and vector-based propagation schemes [51], all of which aim to reduce discretization artifacts and improve the realism of simulated fire perimeters.

In this chapter we propose our own solution to this problem: fine-tuning the parameters of the CA simulator, to best match the results of the elliptical model

of fire spread. Since wildfire conditions change rapidly, causing the simulator to be executed repeatedly, our goal is to devise a simulator parameter fine-tuning scheme that can also be executed frequently, thus allowing newer simulations to be ran with updated parameters. A secondary goal is for the scheme to work with different CA-based models, which, however, retain the computational advantages of discrete CAs.

6.1.1 Calibration and Parameter Estimation in CA Wildfire Models

In the literature, a variety of methods have been proposed to calibrate the parameters of CA wildfire spread models, highlighting the importance of accurate parameter estimation for reliable fire prediction. Reviewing these approaches provides the necessary context for the calibration methodology presented below.

Ghisu et al. [52] proposed a calibration methodology to optimize the correction factors governing fire spread in a CA model. Their approach modified the spread velocity equations derived from Rothermel's [39] and Richards' [146] fire spread models by introducing five correction factors that compensate for the directional bias inherent in raster-based CA simulations. These factors were estimated through a numerical optimization procedure, formulated as a multi-objective optimization problem. They solve the resulting multi-objective optimization problem using a Tabu Search optimizer [55] paired with the Hooke and Jeeves (H&J) method [147], to estimate the optimal correction factors. The calibrated CA model was then validated by comparison with existing CA simulators and with the vector-based FARSITE simulator [57] under realistic wind and slope conditions.

Couce and Knorr [148] proposed a statistical offline calibration approach for a stochastic CA model using data from satellite-detected wildfires in 2003. Instead of calibrating the model against individual fire perimeters, the authors adjusted the model parameters by comparing the statistical characteristics of the simulated and observed fire-size distributions. The authors noted that, although the obtained results are specific to the selected CA model and the geographical region investigated, the proposed calibration procedure could be adapted to fine-tune and

validate other CA wildfire models. in regions where sufficient fire observations are available.

Alexandridis et al. [36] employed an offline calibration approach based on Nelder–Mead optimization method [149] to estimate the parameters of a CA wildfire spread model using data from a historical wildfire. The calibration was performed prior to operational use by repeatedly simulating the spread of the 1990 Spetses Island fire under known environmental conditions and comparing the simulated burned area with the observed fire perimeter. CA model parameters were adjusted through an iterative optimization procedure to minimize the discrepancy between the simulated and observed burn patterns.

Clarke et al. [150] calibrated their CA wildfire model using thermal infrared imagery acquired during a prescribed burn in the San Dimas Experimental Forest in 1986. Rather than employing an automated optimization procedure, the authors adjusted the empirical model parameters by comparing simulated fire propagation with the observed fire evolution derived from remote sensing data. The calibrated model was then validated against the experimental burn to assess its ability to reproduce the observed spatial and temporal fire behavior.

6.2 CA Simulation Algorithms Considered

6.2.1 Baseline

We adopt the CA formulation proposed in [36], which was presented in Section 2.5.1, as the baseline model in order to prove the effectiveness of the fine-tuning mechanism. A typical Moore neighborhood configuration is used for neighborhood selection, while the burn probability is computed according to Equation 6.1.

$$P_{burn} = P_0(1 + P_{veg})(1 + P_{den})P_wP_s \quad (6.1)$$

Two additional CA variants will also be considered, each incorporating supplementary coefficients that require calibration.

6.2.2 Weather Coupled

The first CA variant is a weather-coupled model that integrates *Fuel Moisture Content* (FMC) as a key determinant of fuel flammability; its formulation is given in Equation 2.3 [37]. For that specific CA variant, the challenge is determining the term P_m (Equation 2.6), which is aligned with the FMC. The FMC is rarely measured directly in practice, as this requires considerable time and a large number of sampling points across the study area, which is unrealistic during a wildfire event. Direct measurements are typically conducted only to validate FMC model estimates in real-world scenarios. In the context of wildfire simulation, FMC is usually obtained indirectly from weather and environmental data. In our implementation, we compute the *Equilibrium Moisture Content* (EMC), a widely used and practical technique for estimating fuel moisture.

We will adopt the EMC estimation proposed by Simard in [151] used in NFDRS [152], which uses H (relative humidity in %) and T (temperature in °F) as input variables (Equation 6.2). Alternatively, the EMC can be obtained using the Nelson model [153]; however, this approach is not adopted, as it introduces additional tunable parameters that may complicate the interpretation of the results of tuning the CA parameters. Nevertheless, both methods do not account for rainfall, which is known to influence fuel moisture content [154].

$$EMC = \begin{cases} 0.03229 + 0.281073 \cdot H \\ -0.000578 \cdot T \cdot H & \text{if } H < 10\% \\ 2.22749 + 0.160107 \cdot H \\ -0.014784 \cdot T & \text{if } 10\% \leq H < 50\% \\ 21.0606 + 0.005565 \cdot H^2 \\ -0.0003505 \cdot H \cdot T - 0.483199 \cdot H & \text{if } H \geq 50\% \end{cases} \quad (6.2)$$

According to the equilibrium moisture time-lag theory [155, 156], fuels dry following a negative exponential curve under constant temperature and relative humidity conditions. Therefore, FMC should, in principle, be updated accordingly

for each fuel class. However, to avoid this complication, we decided to use weather data upscaled to 5-minute intervals in the evaluation of the method (Section 6.4.1). The variables introduced are a and b (Equation 2.6), which require fine-tuning, in addition to the parameters of the baseline CA model.

6.2.3 Dynamic Neighborhood

The third CA variant used in the evaluation is inspired by the approach proposed by Freire [38] (Section 2.6.2). While a typical Moore neighborhood configuration adequately captures local spread dynamics under moderate wind conditions, it fails to represent the long-range spotting and accelerated propagation observed empirically when wind gusts exceed critical thresholds. To address this limitation, we introduce a dynamic neighborhood switching mechanism controlled by a wind gust threshold parameter s (in km/hr).

At each simulation timestep, the wind gust speed $V_{gust}(t)$ is read from the available meteorological data. When $V_{gust}(t) \leq s$, the standard Moore neighborhood is retained, consisting of 8 neighboring cells (Figure 6.2(a)). When $V_{gust}(t) > s$, the neighborhood is expanded to an extended Moore configuration, consisting of a 5×5 window of 24 surrounding cells (Figure 6.2(b)). The parameter s can be treated as a calibration variable within the ACO_R optimization framework alongside the remaining tuning parameters, allowing its value to be inferred from expected fire perimeter data (using the technique described in Section 6.3.3) rather than assumed a priori.

6.3 Ant Colony Optimization for Continuous Optimization Problems

Ant Colony Optimization (ACO) [124] is a robust metaheuristic optimization technique that has been successfully applied to a wide range of problems across numerous scientific and engineering disciplines [125, 158]. Although ACO is primarily known for its effectiveness in discrete combinatorial optimization problems,

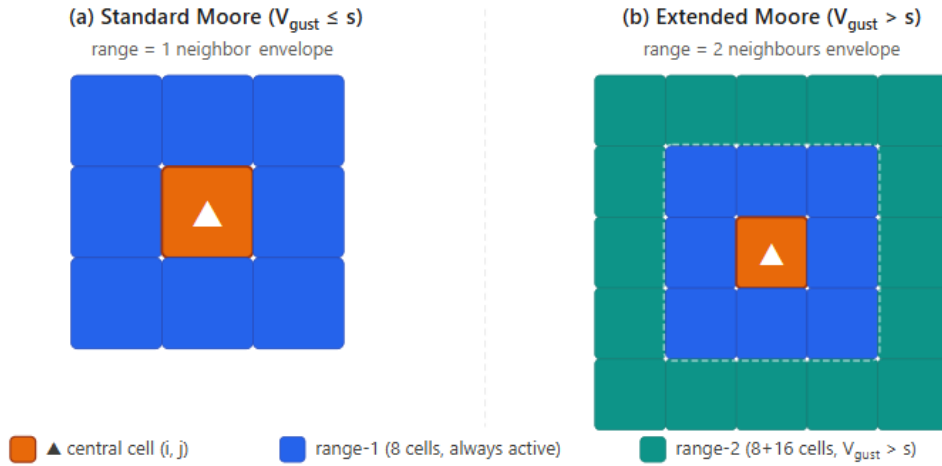


Figure 6.2: (a) Standard Moore neighborhood (b) extended Moore neighborhood when $V_{gust}(t) > s$. Adapted from [157].

several variants have been developed to address continuous optimization tasks.

ACO implementations for continuous solution spaces extend the original framework by adapting the originally discrete, single-dimension pheromone representation to a continuous form. In addition, they introduce new solution construction mechanisms, thereby enabling their application to *Continuous Optimization Problems* (COPs). Among them, Socha and Dorigo proposed the ACO_R [159] algorithm, which is considered the most widely applicable, while maintaining the original formulation of the ACO framework. Previously proposed variants do exist, like the one proposed by Bilchev and Parmee [160] (CACO algorithm), the API algorithm by Monmarché et al. [161] and the hybrid implementation of the CIAC algorithm by Dréo and Siarry [162]. Later approaches include Liu et al. [163] ACO based on a position distribution model, DACO_R [164] and ARACO [165].

6.3.1 ACO_R

We utilize the ACO_R [159] algorithm to optimize the parameters of the dynamic CA simulation, as it provides an effective mechanism for exploring high-dimensional continuous parameter spaces, while remaining consistent to the original ACO framework [124]. The main novelty lies in replacing the incremental

probabilistic construction of solutions used in ACO with a solution archive. Specifically, the ACO_R framework maintains a finite archive T containing k solutions, ranked according to their objective function values. Each solution in the archive is associated with a weight w_i that reflects its quality (according to the objective function) and determines its influence on the sampling process. New candidate solutions are generated by each artificial ant in n (dimension of solution space) incremental steps by (indirectly) sampling a set of Gaussian kernel G_i *Probability Density Functions* (PDFs), where each solution stored in T serves as the mean of a Gaussian kernel (Equation 6.3):

$$T = \{S_1, S_2, \dots, S_k\} \quad (6.3)$$

The objective function $f : S \rightarrow \mathbb{R}$ is optimized over the search space $S \subseteq \mathbb{R}^n$ where n denotes the problem dimensionality. A candidate solution $S = \{x_1, x_2, \dots, x_n\}$ must also be feasible and therefore must not violate the constraints of the problem, denoted by the constraint set Ω .

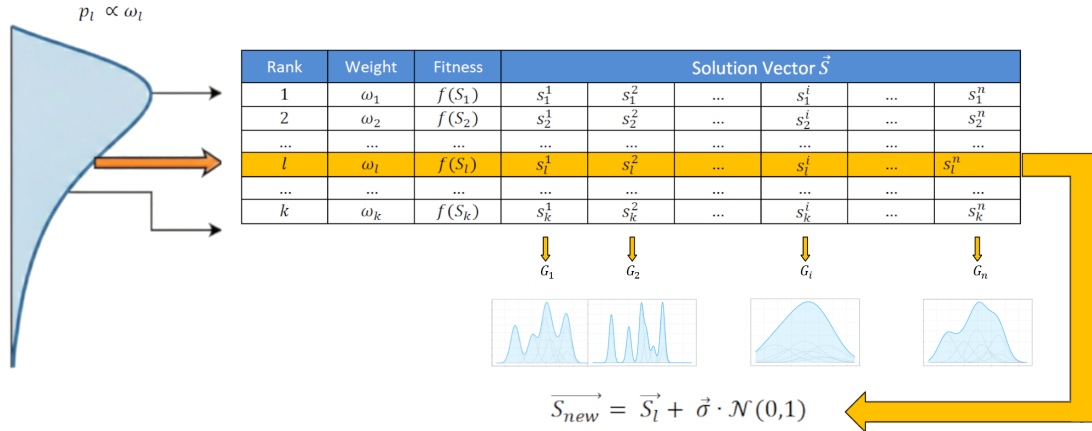


Figure 6.3: Discrete approximation of the continuous search space through the solution archive T . Adapted from [157].

The construction of solution archive T is illustrated in Figure 6.3. The pheromone trail concept of ACO is replaced by the weight factor. An artificial ant selects a stored solution $\vec{S}_l$ with probability p_l and then creates in n steps the new solution $\vec{S}_{new}$ using the selected solution as the mean and a standard

deviation σ derived by the other stored solutions. This emulates the sampling of the corresponding Gaussian kernels G_i for each problem dimension.

Then, it constructs a new candidate solution S_{new} by sampling a Gaussian PDF around the “neighborhood” of the selected solution:

$$S_{new}^i = S_l^i + \sigma_l^i \cdot \mathcal{N}(0, 1) \quad (6.4)$$

where l is the selected solution from T and i the i^{th} dimension of the problem. Each standard deviation is calculated as follows (Equation 6.5):

$$\sigma_l^i = \xi \sum_{e=1}^k \frac{|s_e^i - s_l^i|}{k - 1} \quad (6.5)$$

Where $\xi > 0$ is a user-specified parameter that regulates the balance between exploration and exploitation, analogous to the pheromone evaporation rate in a discrete ACO framework. Higher values of ξ increase the distance between newly generated solutions and the best-known ones, enhancing exploration at the expense of convergence speed. Finally, the effect of pheromone update is accomplished by examining the newly generated solutions and, if they exhibit better fitness than those stored in K , inserting them with the simultaneous removal of the worst-performing solutions, to keep the archive size constant.

6.3.2 Objective Function

In continuous optimization, the search process is driven by an objective function that mathematically describes the optimization goal of the problem. This function evaluates the performance of candidate solutions by assigning a scalar value to each set of decision variables, thereby enabling the identification of optimal or near-optimal solutions. Depending on the nature of the problem, the objective function may be formulated either for minimization or maximization. Its formulation is inherently problem-specific, as it encapsulates the characteristics and optimization criteria associated with a particular application domain. Consequently, its formulation plays a critical role in guiding the search behavior, since it determines how candidate solutions are evaluated and differentiated within the

search space. The characteristics of the objective function shape the optimization capabilities of the method and directly affect the convergence behavior, search efficiency, and solution quality achieved by the optimization algorithm.

Within the proposed dynamic near-real-time wildfire simulation scheme (Section 3.5), we define an objective metric to quantify how closely the simulated fire spread aligns with the expected fire behavior, thereby measuring its deviation from the ideal scenario. This is achieved through the use of a theoretical elliptical representation of the dynamically evolving burned area during the simulation, which constructs an elliptical envelope around the ignition point, as described in Section 6.3.3. Finally, the objective function is defined as the deviation between the expected burned area, represented by the elliptical envelope, and the simulated burned area at the current stage of the CA simulation.

6.3.3 Theoretical Elliptical Envelope Algorithm

At a macroscopic level, extracting the perimeter of a wildfire in a pre-fire event is effectively equivalent to simulating its spatial spread. In this context, the predicted fire perimeter is represented by the geometric evolution of the fire front at specific time points. Comprehensive two or three-dimensional fire behavior models simulate additional physical properties of the fire, such as ROS, fireline intensity, flame length, and fire-induced interactions with the atmosphere. Perimeter estimation can be regarded as a simplified, yet computationally efficient means of assessing the potential spatial extent of wildfire impact, rather than a full characterization of fire behavior or damage severity.

Perimeter estimation is most commonly approached through methods based on Huygens' wavelet principle, whereby each point along an existing fire front is treated as the origin of a secondary wavelet, and the envelope of these wavelets defines the sub-sequent fire perimeter [166]. While the vector-based application of Huygens' principle effectively represents the curvature of the fire perimeter, the *Minimum Travel Time* (MTT) algorithm [83] offers a computationally efficient alternative by modeling fire spread as a network optimization problem over a spatial grid. MTT discretizes the solution space into a grid of nodes (similar to CA) and

calculates the fastest travel path (employing a cost function that examines fuel, terrain, etc.) from the ignition to source. This method preserves the physical analogy of Huygens’ principle, while maintaining a reasonable computational footprint, making it ideal for large-scale estimations.

However, due to computational constraints on edge devices such as mobile phones, a simplified approach based exclusively on the elliptical growth model is employed [167]. This efficiency is critical because this approximation will serve as the objective function for the ACO_R algorithm, therefore it requires extremely fast evaluation times. In the proposed approach, fire propagation is represented by generating spread vectors at discrete time intervals. Rather than tracking a complex perimeter, the aggregate expansion is encapsulated within a single bounding ellipse. The geometry of this ellipse is governed by the Length to Breadth (L/B) ratio, which is expressed as a function of wind speed and direction [168, 54], as illustrated in Figure 6.4.

The theoretical elliptical envelope method approximates fire growth by constructing a single elliptical envelope around a piecewise vector path. The process begins by calculating a sequence of spread vectors at discrete time steps (at a lower frequency than in the CA), where the magnitude and direction of each segment are determined by the weather conditions and fuel combustibility (e.g., using Rothermel’s fire spread model), creating a ‘zig-zag’ trajectory of the theoretically expected spread. Instead of retaining an arbitrary polygon from this complex polyline, the method derives a simplified enclosing ellipse that is mathematically fitted to this vector chain. The geometry of this envelope is defined by a major axis extending from the final vector coordinate (the fire head), backward through the ignition point to a calculated rear boundary, ensuring the ignition source remains collinear with the primary axis of propagation. This construction forces the ignition point to lie along the major axis in the rear focus point, while the shape’s eccentricity is controlled by the L/B ratio derived from empirical models like those presented in [54, 168, 169]. As an example, Figure 6.5 depicts a simulated elliptical perimeter, derived from the aggregation of five consecutive spread vectors.

Although the formulation for estimating the L/B ratio proposed by Simard

Fire Shape Comparison by Length-to-Breadth (L/B) Ratio

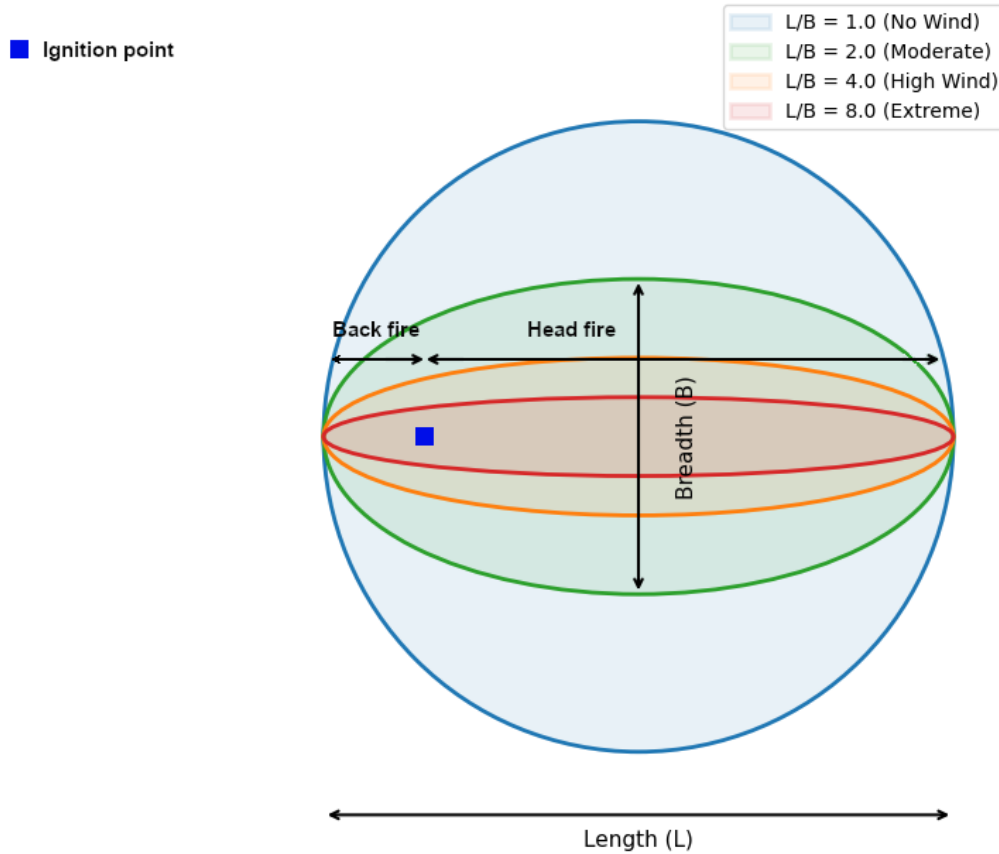


Figure 6.4: Comparison of fire ellipse shapes for different L/B ratios, ranging from no wind (1.0) to extreme conditions (8.0). Adapted from [157].

[170] accounts for the influence of the standard deviation of wind direction, its application requires detailed and accurate in-field wind observations which can be delivered via our proposed crowdsourcing framework; in the absence of such data, this approach would introduce an additional source of uncertainty. Therefore, in order to validate the tuning mechanism, the L/B ratio is derived using the simpler Alexander's expression [54] given in Equation 6.6. An alternative would have been to adopt Anderson's [168] formulation; however, its reliance on a double ellipse representation would substantially increase model complexity and was thus not pursued.

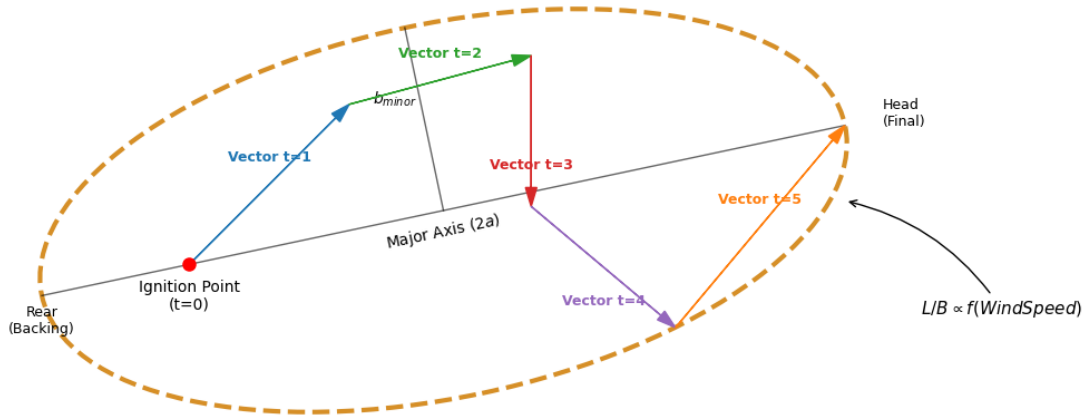


Figure 6.5: Progressive reconstruction of theoretical elliptical envelope of a simulated wildfire in incremental time intervals. Adapted from [157].

$$L/B = 1.0 + 0.00120 \cdot W^{2.154} \quad (6.6)$$

where, W is the 10-m open wind speed in km/hr and it is applicable for wind speeds up to 50 km/hr. The outline of the method is given in Algorithm 6.

Algorithm 6 Theoretical Ellipse Envelope Construction

Require: weather vector **weather**

Require: ignition point **ignition**

```

1:  $point \leftarrow \mathbf{ignition}$ 
2:  $vectors \leftarrow \{\mathbf{point}\}$ 
3: for each sampling interval  $i$  do
4:   find dominant vegetation and calculate  $ROS$  during time  $i$ 
5:    $v_i \leftarrow ROS \cdot dt$  with direction from  $point$  towards  $weather[i].direction$ 
6:    $point \leftarrow v_i$ 
7:    $vectors \leftarrow \{vectors \cup v_i\}$ 
8: end for
9:  $LB \leftarrow \text{CalculateLB}(\text{weather.speed}, vectors)$ 
10:  $envelope \leftarrow \text{CalculateEllipse}(\text{ignition}, vectors, LB)$ 
11: return  $envelope, area(envelope)$ 

```

6.3.4 Formulating CA Parameter Tuning as a COP

Having established the objective function through the progressive envelope construction algorithm (Algorithm 6), the problem of fine tuning specific coefficients of the CA model can now be formulated as a continuous optimization problem (Equation 6.7) applied to the Baseline model presented in Section 6.2.1:

$$\min_X f(X) = |A_{\text{est}}(X) - A_{\text{sim}}(X)|$$

$$X = \{P_0, c_1, c_2, \alpha_s\}$$

$$\text{subject to } 0 \leq P_0 \leq 1, \tag{6.7}$$

$$0 \leq c_1 \leq 1,$$

$$0 \leq c_2 \leq 1,$$

$$0 \leq \alpha_s \leq 1,$$

where $A_{\text{est}}(X)$ is the area obtained from Algorithm 6 and $A_{\text{sim}}(X)$ is the area obtained from the current CA simulation. The same formulation applies for the CA

variants presented in Section 6.2.2 and Section 6.2.3 with the vector X becoming $X = \{P_0, c_1, c_2, \alpha_s, a, b, s\}$.

The parameter calibration process employing the ACO_R metaheuristic is presented in Algorithm 7, while the flowchart of the method (focused on the case where streamed data from crowdsourcing inputs are received from in-field users, in order to improve the wildfire spread simulation) is given in Figure 6.6. First, a preliminary CA simulation (for a limited duration of time) is conducted with default values $X = \{P_0, c_1, c_2, \alpha_s\}$ for the parameters that will be optimized. For the same duration, a theoretical elliptical envelope of the same fire is constructed according to Algorithm 6 (using few sampling points). The objective function is formulated as the area difference between the area of the CA simulation and the area of the envelope. Then, after i iterations of ACO_R metaheuristic, new values for X are obtained that can be used to re-run the full CA simulation for a longer duration.

This parameter calibration process is repeated whenever new crowdsourced data becomes available from in-field users or when the CA simulation yields poor performance. The evaluation of the objective function (during each iteration of ACO_R) with updated X values unavoidably leads to a new execution of a CA simulation. Consequently, these preliminary simulations are conducted over a shorter duration (to reduce computational cost) than the full simulation with the optimized values.

Algorithm 7 CA Parameter Tuning Using the ACO_R Metaheuristic

Require: weather vector **weather**

Require: ignition point **ignition**

Require: time **duration**

```

1: set initial values for parameters  $X = \{P_0, c_1, c_2, \alpha_s, a, b, s\}$ 
2:  $simulated\_grid \leftarrow \text{run\_CA\_simulation}(\text{weather}, \text{ignition}, X, \text{duration})$ 
3:  $simulated\_area \leftarrow \text{area}(\text{simulated\_grid.burned\_cells})$ 
4: while new data are available do
5:    $envelope \leftarrow \text{ELLIPTICALENVELOPE}(\text{weather}, \text{ignition})$ 
6:    $objectiveFunction \leftarrow |\text{area}(envelope) - \text{area}(simulated\_area)|$ 
7:    $T, \omega \leftarrow \text{intializeSolutionArchive}(\text{InitialRandomSolution}, q)$ 
8:   for each iteration  $i$  do
9:     for each ant  $k$  do
10:      select solution  $l$  from  $T$  with probability  $p_l$  according to  $\omega_l$ 
11:       $s_k \leftarrow x_l$ 
12:       $\sigma_k \leftarrow \text{calculate\_}\sigma(\xi, l)$ 
13:       $s_k^{new} \leftarrow s_k + \sigma_k \cdot \mathcal{N}(0, 1)$ 
14:    end for
15:    update  $T$  with new ant solutions according to  $objectiveFunction()$ 
16:  end for
17:   $X_{new} \leftarrow T[0]$  ▷ best solution in  $T$ 
18: end while
19:  $simulated\_grid \leftarrow \text{run\_CA\_simulation}(\text{weather}, \text{ignition}, X, \text{duration})$ 
20: return  $simulated\_grid$ 

```

6.3.5 Parameter Tuning Complexity

To estimate the method's computational complexity, it is necessary to identify the key components of the tuning process: the ACO_R layer executed for I_{ACOR} iterations, and the subsequent evaluation of the objective function, which involves a CA simulation over I_{CA} iterations, for each of the candidate solutions generated

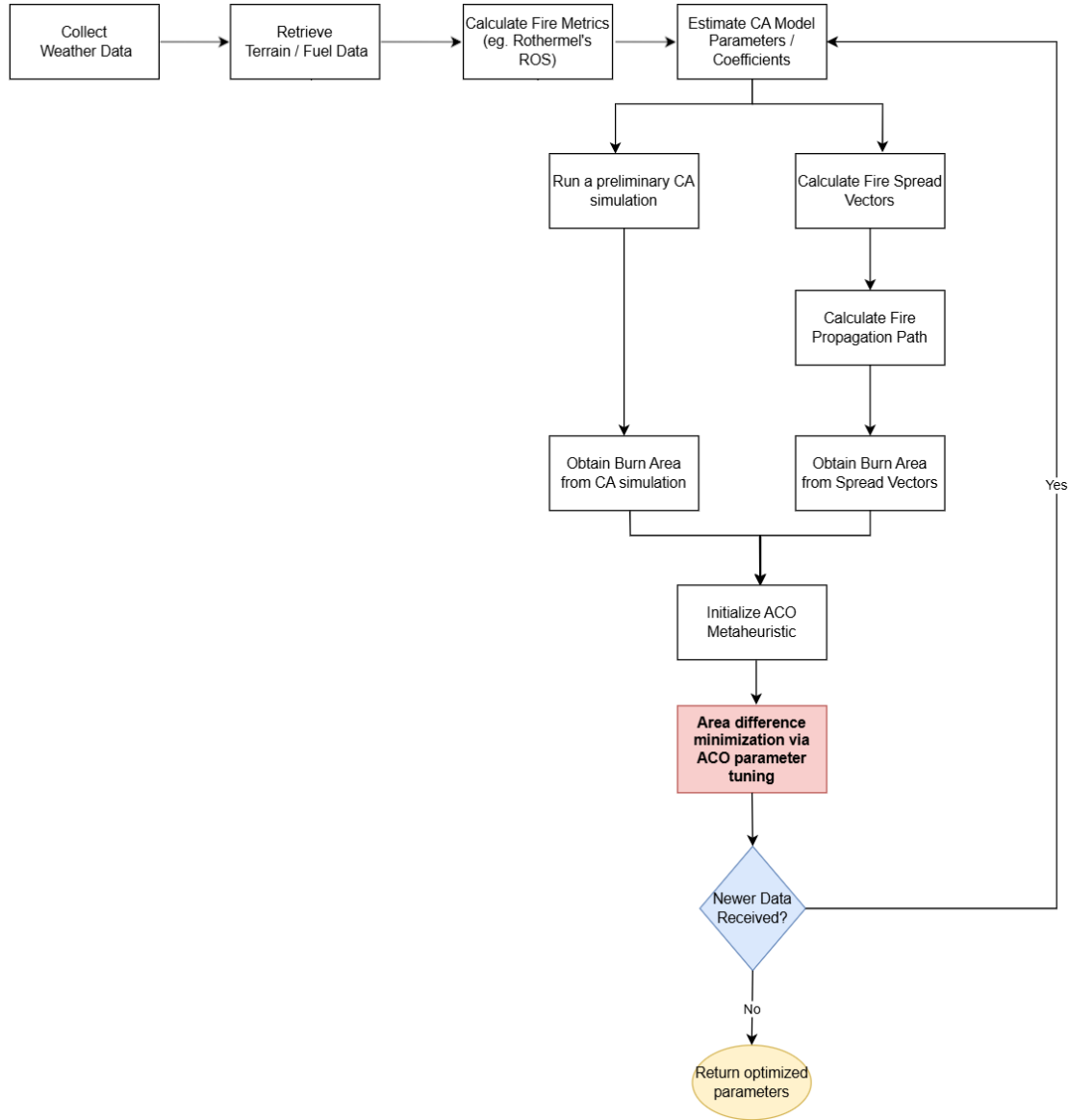


Figure 6.6: Dynamic fine-tuning of CA parameters driven by crowdsourced streaming data from in-field users. Adapted from [157].

by ACO_R . Note that I_{CA} refers to the iterations required for the CA simulation during the tuning process (preliminary run), not the full duration of the simulated event for which the CA is eventually executed.

The ACO_R layer complexity is derived from the mechanism artificial ants use to explore the search space, and how the solution archive is maintained at each iteration. For each I_{ACOR} iteration, m artificial ants generate an equal number of

candidate solutions through sampling n Gaussian kernels (number of decision variables). Then, the solution archive of size k must be merged with the m newly generated solutions. The size of archive T remains k , therefore, the m worst solutions will be evicted from T for the next iteration of ACO_R . Hence, the complexity of the optimization process by ACO_R is $C_{\text{ACOR}} = \mathcal{O}(I_{\text{ACOR}} \cdot (m \cdot n + (m + k) \log(m + k)))$, where the left side of the complexity corresponds to the generation of the candidate solutions and the right side with reforming T in order to contain the k most promising solutions out of the k existing and m candidate solutions; since the k existing solutions are already sorted, faster implementations are possible, but, as we will see, other costs dominate. Specifically, to identify the fitness of the m candidate solutions, we need to evaluate the objective function for them, by running a preliminary CA simulation.

The complexity of the CA simulations is not directly related to the grid size, but to the number of cells transitioning from burning state to burned. In a CA run, we do not process every cell at every step; instead, we only examine the neighbors of currently burning cells. Therefore, the complexity of CA is $C_{\text{CA}} = \mathcal{O}(I_{\text{CA}} \cdot B_{\text{avg}} \cdot N \cdot \text{Estimate}_{P_{\text{burn}}})$ where B_{avg} is the average number of burning cells per step and N the neighborhood configuration (e.g., for a Moore neighborhood, $N = 8$). $\text{Estimate}_{P_{\text{burn}}}$ also introduces computational overhead, since cell burn probabilities must be updated at each iteration (only for neighboring cells) due to changing conditions (e.g., wind direction). Nevertheless, accessing these environmental conditions from external sources may introduce additional I/O overheads that are beyond the scope of this analysis.

The computational complexity of the proposed tuning mechanism is thus dominated by the objective function evaluations. While the ACO_R framework introduces an overhead, the primary cost is the iterative execution of the CA simulator for each candidate solution. In the worst (unrealistic) case, where the fire covers the entire grid, the complexity is $C_{\text{total}} = \mathcal{O}(I_{\text{ACOR}} \cdot m \cdot G \cdot \text{Estimate}_{P_{\text{burn}}})$, where G is the number of cells in the grid, but then, the simulation exits before all I_{CA} iterations have completed. Hence, we can express the complexity of the method solely via the complexity of the CA as follows (Equation 6.8):

$$C_{total} \approx \mathcal{O}[I_{ACOR} \cdot (m \cdot C_{CA})] \quad (6.8)$$

6.4 Experimental Results

6.4.1 Area of Interest

To evaluate the effectiveness of the CA wildfire simulation fine-tuning mechanism, we selected the wildfire incident that occurred on Mount Hymettus near Glyfada, Greece (Figure 6.7). The event took place in June 4th 2022, at 12:50 pm local time. Although the affected area was relatively small (about 432 ha according to the EMSR576 Copernicus Activation Report), the fire spread rapidly, and the incident was considered highly serious due to its proximity to residential areas.

For each CORINE Land Cover (CLC) class within the study area (Figure 6.7), a corresponding pair of P_{den} and P_{veg} coefficients was assigned for use in the CA model, based on experimental analysis and values reported in the literature [36, 37]. The resulting assignment is listed in Table 6.1. Additionally, elevation data of the study area were obtained from the *Shuttle Radar Topography Mission* (SRTM) DEM.

6.4.2 Meteorological Conditions

Meteorological conditions were obtained from a nearby weather station, which was part of the automatic station network of the National Observatory of Athens (Figure 6.8). Peak wind gust intensities and directional variability are observed concurrently with high afternoon temperatures and low atmospheric moisture. Wind speed is plotted together with gust data (dashed line and shaded region) to highlight periods of increased atmospheric turbulence over the 24-hour period. To ensure high temporal granularity for the simulated reconstruction of the incident, the datasets were processed at a 5-minute resolution. It should be noted that while the visualization presents continuous 5-minute data points, these values were derived via linear interpolation from the original, lower-frequency sampling

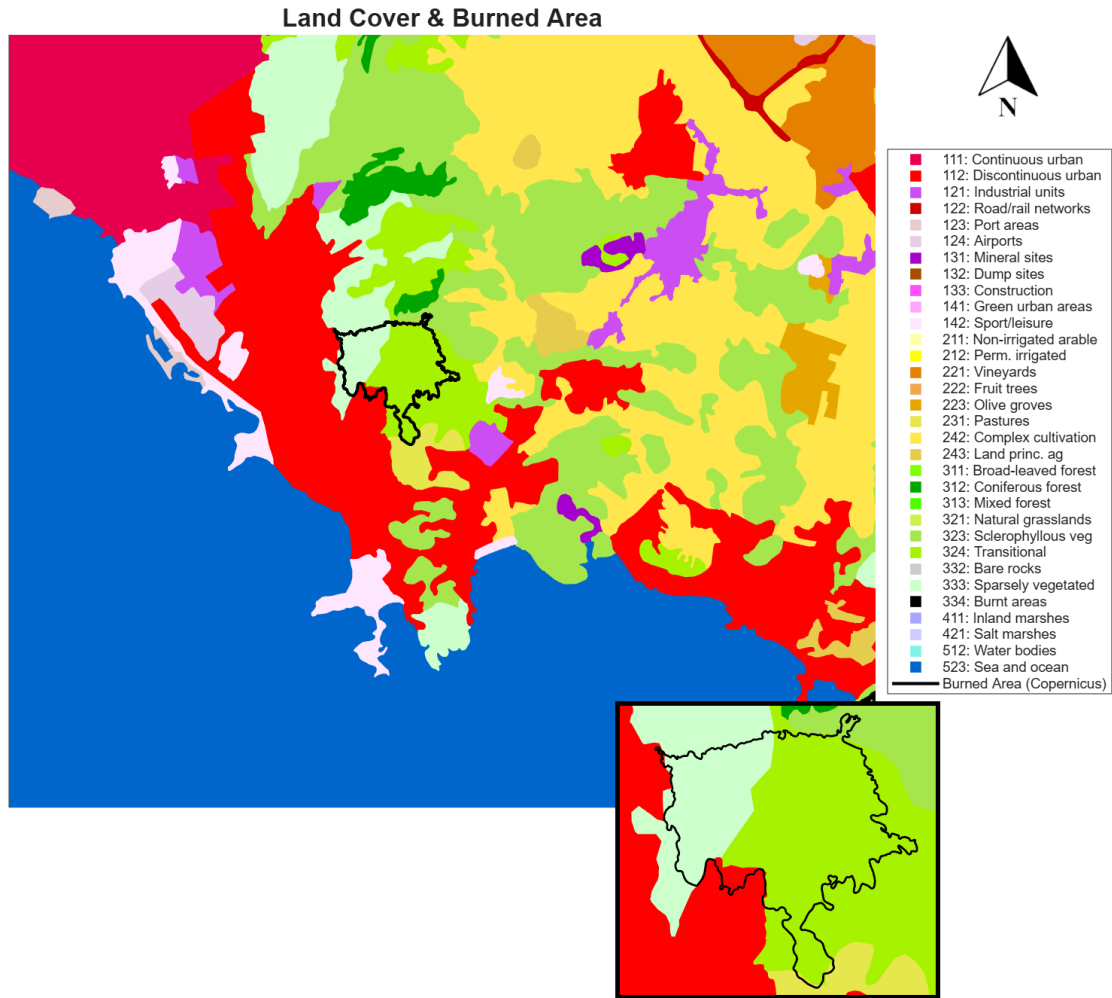


Figure 6.7: Study area near Glyfada, Greece. Land usages acquired from the CLC 2018 dataset, while the burned area of the 2022 wildfire (black outline) was estimated from the EMSR576 Copernicus Activation Report. Adapted from [157].

rate of the weather station. This upsampling choice provides a smoother trend analysis, though the primary peaks and troughs remain anchored to the original observation points. Special caution should be exercised when interpreting wind direction trends. As the data were upsampled from a lower frequency source, the directional values between the original observations are subject to interpolation bias. Consequently, large scale shifts in wind direction are captured effectively, whereas high frequency directional fluctuations should be interpreted as indicative rather than as absolute.

Table 6.1: Assigned P_{den} and P_{veg} coefficients for each CLC class used in the CA model.

CLC Code	Type	P_{den}	P_{veg}
111	Continuous urban fabric	-1.0	-1.0
112	Discontinuous urban fabric ^a	-0.7	-0.8
121	Industrial or commercial units	-0.7	-0.8
123	Port areas	-1.0	-1.0
124	Airports	-1.0	-1.0
131	Mineral extraction sites	-0.9	-0.9
142	Sport and leisure facilities	-0.5	-0.5
223	Olive groves	0.6	0.45
231	Pastures	0.45	0.4
242	Complex cultivation patterns	0.5	0.5
243	Land principally occupied by agriculture, with significant areas of natural vegetation	0.55	0.5
312	Coniferous forest	0.75	0.7
323	Sclerophyllous vegetation	0.65	0.6
324	Transitional woodland/shrub	0.35	0.4
333	Sparsely vegetated areas	0.0	-0.15
523	Sea and the ocean	-1.0	-1.0

^a Includes suburban residential zones with mixed vegetation cover.

6.4.3 Simulation Procedure

To evaluate the efficacy of the proposed parameter tuning mechanism, it was applied to the WUI fire event presented in Section 6.4.1. Initially, a wildfire spread simulation over a 6-hour horizon was conducted using the three CA variants described in Section 6.2, which served as the baseline configuration for subsequent tuning experiments.

The final simulation results for each CA variant were obtained through a multi-stage calibration process aimed at iteratively refining the CA model parameters in order to enhance the predictive accuracy of the wildfire spread simulation. The overall procedure consisted of the following stages:

1. **Preliminary Simulation and Envelope Construction:** A truncated 150-minute (2.5 hours) preliminary CA simulation was executed using the same parameters as in the baseline (no optimization) simulation (Table 6.2). For the same duration, a theoretical elliptical envelope was constructed using Algorithm 6. This involved generating spread vectors and fitting an

Meteorological Conditions: Glyfada 2022-06-04 Incident

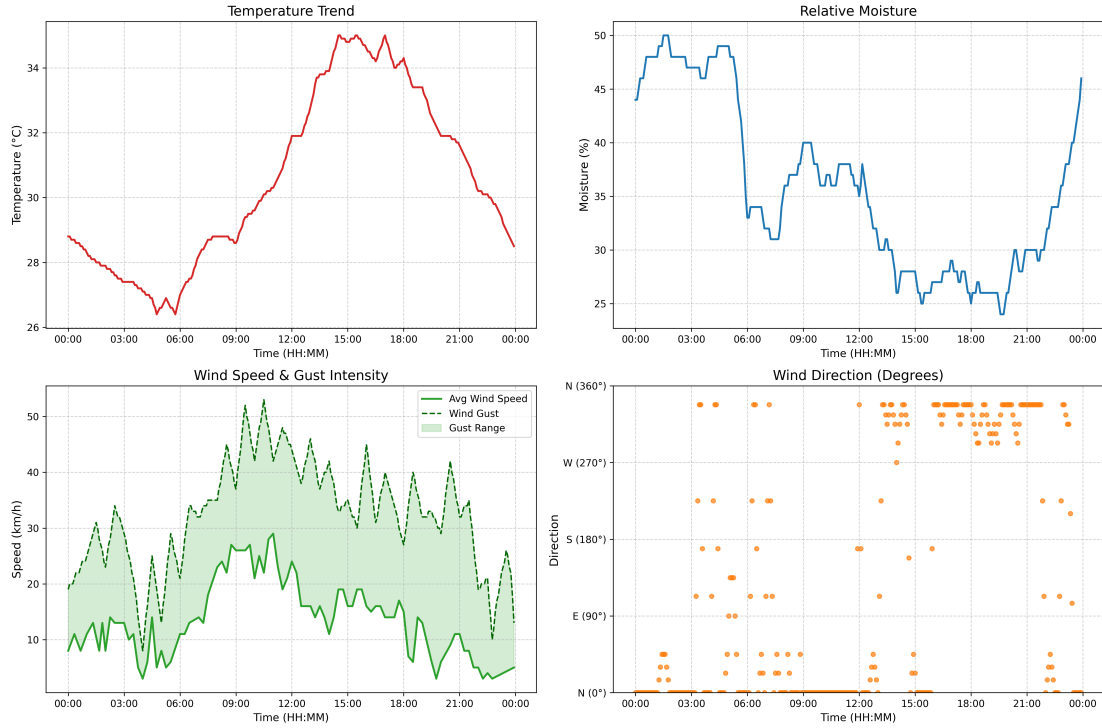


Figure 6.8: Environmental conditions in the Glyfada, Greece area during the June 4th wildfire event. Adapted from [157].

encapsulating ellipse, where the semi-major axis was aligned with the vector extending from the ignition point to the final spread frontier.

2. **Objective Function Definition:** The objective function to be minimized was defined as the absolute difference in the number of burned cells (of the same size) between the preliminary CA simulation output and the theoretical elliptical model output. It should be noted that each objective function evaluation requires both the CA method and the elliptical envelope model to be executed, using the parameters determined during the optimization stage.
3. **ACO_R Metaheuristic Optimization Stage:** The ACO_R algorithm was deployed using the configuration set specified for each test case. During each objective function evaluation, ACO_R initiated a new CA simulation instance, iteratively adjusting the CA input parameters based on the candidate solu-

tions generated.

4. **Convergence and Validation:** After the termination criterion is met (number of iterations) ACO_R algorithm converges to an optimized parameter set for the CA model. Finally, these optimized parameter set is applied to the final 6-hour CA wildfire spread simulation.

6.4.4 Baseline CA

The baseline CA variant (Section 6.2.1) requires fine tuning the vector $X = \{P_0, c_1, c_2, \alpha_s\}$, or 4 decision variables. The initial CA simulation conducted with the parameter set listed in Table 6.2. The ACO_R algorithm performed the fine tuning process using the parameter configuration listed in Table 6.3.

Table 6.2: Parameters used in the initial CA simulation.

Parameter	Value	Parameter	Value
P_0	0.450 ^a	α_s	0.078
c_1	0.045	l_{cell}	30.6m
c_2	0.131	dt	5min

^a P_0 value selected based on the literature findings [36, 37] and proportional to the calculated no-wind, no slope ROS, based on calculations conducted with Behave Plus 6 software [171].

Table 6.3: Parameters used for the ACO_R algorithm.

Parameter	Value	Parameter	Value
Number of ants	5	q	0.05
Solution archive size	30	ξ	0.10
Number of iterations	35		

The ACO_R optimization stage produced an optimal set of input parameters for the CA simulation, as listed in Table 6.4. Using the optimized parameter set, a full-scale 6-hour CA wildfire spread simulation was conducted, yielding a predicted burned area extent of 5753 burned cells. Figure 6.9 illustrates the actual observed burned area (white outline), as reported in the EMSR576 report. The results

of the initial 6-hour simulation without parameter optimization are presented in Figure 6.9(a), while the optimized simulation output obtained after the ACO_R calibration stage is shown in Figure 6.9(b).

Table 6.4: Optimal parameter set for the baseline CA model obtained through ACO_R optimization.

Parameter	Value	Parameter	Value
P_0	0.085240	α_s	0.036826
c_1	0.352091	burned cells in 6-hours:	5753 burned cells
c_2	0.248356		

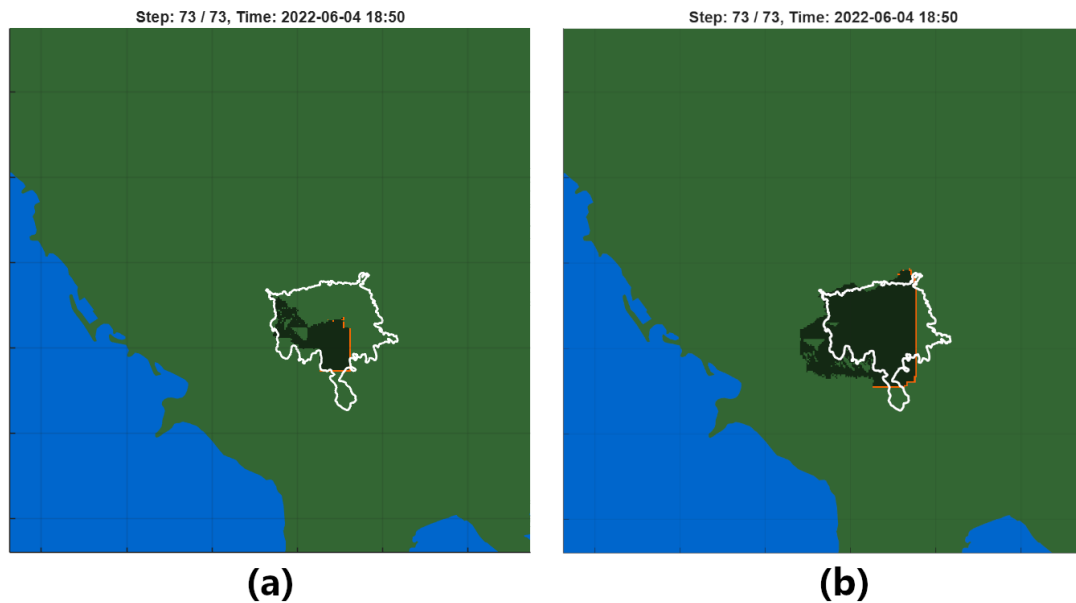


Figure 6.9: Comparison of burned area (black) after 6-hours (a) without and (b) after parameter fine-tuning. Adapted from [157].

6.4.5 Weather-Coupled CA

The weather-coupled CA variant (Section 6.2.2) requires the fine-tuning of two additional decision variables (six in total), so $X = \{P_0, c_1, c_2, \alpha_s, a, b\}$, in order to account for fuel moisture content effects. The initial CA simulation conducted

with the same parameters used for the previous variant (Table 6.2) with the addition of the value 2.015 for a and 0.091 for b . Moreover, given that the term P_m is derived from a non linear regression analysis conducted by Fernandes [40], the decision variable a was constrained to the range $[2, 3.5]$ and variable b was constrained to the range $[0.05, 0.15]$ (Table 6.5). The ACO_R algorithm performed the fine tuning process using the parameter configuration listed in Table 6.6.

Table 6.5: Parameters used in the initial CA simulation for the weather-coupled variant.

Parameter	Value	Parameter	Value
P_0	0.450	a	2.015
c_1	0.045	b	0.091
c_2	0.131	l_{cell}	30.6m
α_s	0.078	dt	5min

Table 6.6: Parameters used for the ACO_R algorithm.

Parameter	Value	Parameter	Value
Number of ants	10	q	0.05
Solution archive size	40	ξ	0.20
Number of iterations	50		

The ACO_R optimization stage produced an optimal set of input parameters for the CA simulation, as listed in Table 6.7. Using the optimized parameter set, a full-scale 6-hour CA wildfire spread simulation was conducted, yielding a predicted burned area extent of 3727 burned cells. Figure 6.10 illustrates the actual observed burned area (white outline), as reported in the EMSR576 report. The results of the initial 6-hour simulation without parameter optimization are presented in Figure 6.10(a), while the optimized simulation output obtained after the ACO_R calibration stage is shown in Figure 6.10(b).

Table 6.7: Optimal parameter set for the weather coupled CA model obtained through ACO_R optimization.

Parameter	Value	Parameter	Value
P_0	0.5159	a	3.3634
c_1	0.2200	b	0.1462
c_2	0.3500	burned cells in 6-hours:	3727 burned cells
α_s	0.6170		

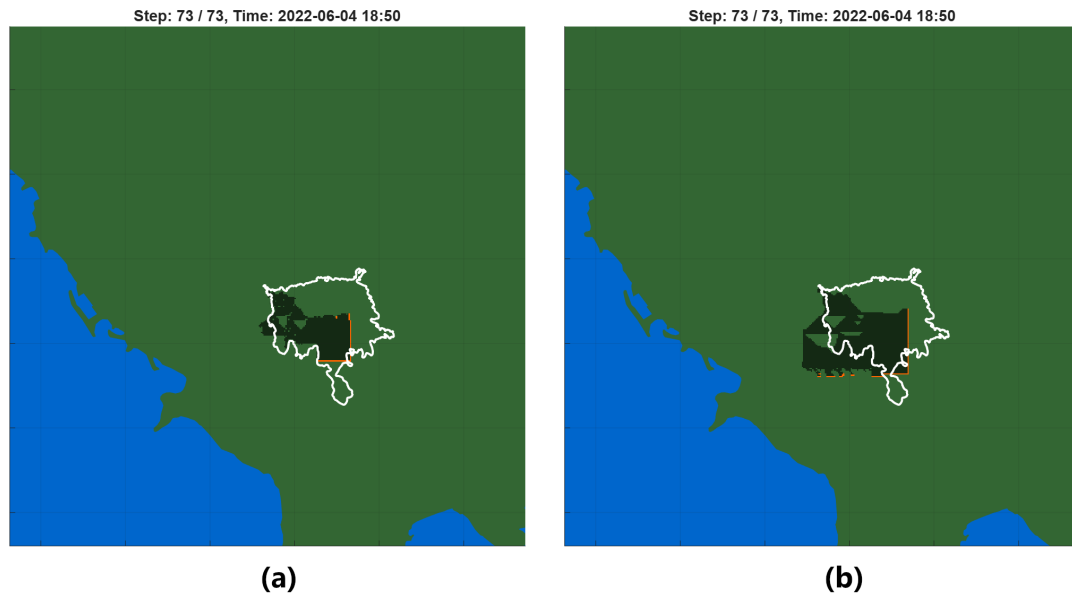


Figure 6.10: Comparison of burned area (black) after 6-hours (a) without and (b) after parameter fine-tuning. Adapted from [157].

6.4.6 Dynamic Neighborhood CA

The CA variant employing a dynamic neighborhood configuration (Section 6.2.3) requires the fine-tuning of one additional decision variable beyond those used in the previous variant (seven in total), so $X = \{P_0, c_1, c_2, \alpha_s, a, b, s\}$, in order to account for the wind speed threshold s governing the transition from the standard neighborhood configuration to the extended Moore neighborhood scheme. The initial CA simulation conducted with the parameters listed in Table 6.8. ACO_R algorithm performed the fine tuning process using the same parameter configuration with the previous experiment (Table 6.6).

Table 6.8: Parameters used in the initial CA simulation for the dynamic neighborhood configuration variant.

Parameter	Value	Parameter	Value
P_0	0.450	b	0.091
c_1	0.045	s	50m/hr
c_2	0.131	l_{cell}	30.6m
α_s	0.078	dt	5min
a	2.015		

The ACO_R optimization stage produced an optimal set of input parameters for the CA simulation, as listed in Table 6.9. Using the optimized parameter set, a full-scale 6-hour CA wildfire spread simulation was conducted, yielding a predicted burned area extent of 4275 burned cells. Figure 6.11 illustrates the actual observed burned area (white outline), as reported in the EMSR576 report. The results of the initial 6-hour simulation without parameter optimization are presented in Figure 6.11(a), while the optimized simulation output obtained after the ACO_R calibration stage is shown in Figure 6.11(b).

Table 6.9: Optimal parameter set for the dynamic neighborhood CA model obtained through ACO_R optimization.

Parameter	Value	Parameter	Value
P_0	0.3907	a	2.5758
c_1	0.0367	b	0.1412
c_2	0.2285	s	36.6141km/hr
α_s	0.6231	burned cells in 6-hours:	4275 burned cells

6.4.7 Analysis of Simulation and Fine-Tuning Results

Table 6.10 presents a summary of the experimental results (before and after the fine-tuning procedure), along with their deviation from the reported burned area in terms of the size of the areas and the overlap between the actual and simulated areas.

Table 6.10: Comparison of burned areas for each examined test scenario.

Experiment	Burned area	% of reported burned area	Overlap with reported area (%)
EMSR576 Report Estimation ^a (Figure 6.7)	432.2 ha	100%	100%
6hr Baseline initial CA Simulation (Figure 6.9(a))	149.3 ha	-65.45%	32.78%
6hr Baseline CA Simulation after ACO _R optimization (Figure 6.9(b))	538.5 ha	+24.59%	80.72%
6hr initial CA with weather coupling (Figure 6.10(a))	182.6 ha	-57.75%	38.24%
6hr CA with weather coupling after ACO _R optimization (Figure 6.10(b))	348.9 ha	-19.27%	50.58%
6hr initial CA with dynamic neighborhood (Figure 6.11(a))	182.6 ha	-57.75%	38.24%
6hr CA with dynamic neighborhood after ACO _R optimization (Figure 6.11(b))	400.2 ha	-7.40%	46.53%

^a The actual duration of the wildfire could not be verified. The original burned area can be further adjusted to 447 ha by closing holes in the polygons resulting from remote sensing estimation. Percentage calculations applied to the officially reported burned area.

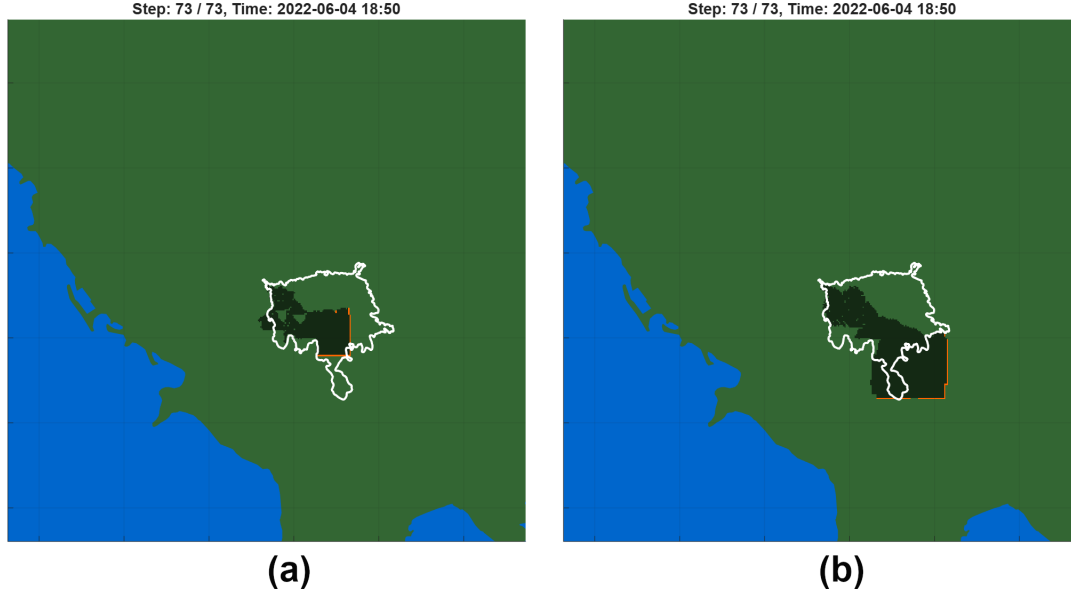


Figure 6.11: Comparison of burned area (black) after 6-hours (a) without and (b) after parameter fine-tuning. Adapted from [157].

It should be noted that the primary objective of this work was not to achieve an exact reconstruction of the Glyfada test case, as the precise environmental and operational conditions of the real event are difficult to determine, with several parameters remaining unrecorded (e.g., mid-flame wind speeds and aerial drop locations), but rather to demonstrate the applicability of a dynamic fine-tuning mechanism based on ACO_R . If exact reconstruction had been the goal, that could potentially lead to overfitting toward the specific observed burned area.

Although the ACO_R framework is capable of handling hundreds of decision variables, a relatively simplified set of CA models was deliberately adopted to preserve model interpretability and reduce computational complexity for portable devices. Additionally, the probability parameter P_0 was also optimized, in contrast to most related studies, where it is assumed to be a constant, representing the probability that a cell ignites under no-wind and no-slope conditions. As the calibration was performed against an elliptical envelope model that incorporates the ROS obtained from Rothermel’s formulation [39] when calculating the spread vectors, the underlying physical behavior associated with P_0 is preserved throughout the optimization process and is implicitly captured through its interaction with

the remaining tunable parameters.

A key aspect of the proposed fine-tuning mechanism is that the probability coefficients $(P_0, a_s, c_1, c_2, a, b, s)$ are treated by the ACO_R algorithm in an agnostic manner with respect to their physical properties and underlying meaning. During the optimization process, the algorithm handles all probability parameters as dimensionless variables to be adjusted, so as to optimize the objective function. Ignition probabilities are calculated as the product of multiple contributing factors (e.g., wind, slope, and fuel flammability), and regardless of what values are assigned to them, the resulting value remains a dimensionless probability. Consequently, there is no unique or direct mapping between probabilities across different simulation outputs. That model-agnostic nature of the fine-tuning process may affect the overall physical behavior of the CA model. For example, the substantial reduction of P_0 from 0.45 to 0.085240 in the baseline CA case, markedly lowers the base ignition probability under no-wind and no-slope conditions, which directly contributes to reducing the systematic estimation of the burned area. At the same time, the increase in c_1 and c_2 amplifies the model's sensitivity to wind speed and direction, leading to a more responsive and directional spread pattern that better reflects the dynamics of wind-driven wildfires. However, the third CA variant (dynamic neighborhood configuration) resulted in a significant overestimation of the burned area (Figure 6.11(b)) outside the reported boundaries. This variant is particularly sensitive to wind compared to the other CA models, making it more prone to amplifying wind-driven spread. The discrepancy likely arises because the model overestimates fire propagation due to wind and/or spotting effects, which typically occur over much shorter time intervals often as brief, intermittent bursts. Interpreting the optimization result sets in the context of the underlying CA transition rules not only clarifies the role of each parameter, but also enhances the physical plausibility and practical relevance of the calibrated model, a capability that is not supported by the current tuning mechanism.

A significant performance limitation arises from the implementation of the objective function, as each evaluation of the objective function, performed for every candidate solution generated during the optimization process, requires the execu-

tion of a limited-duration CA simulation. Consequently, this procedure constitutes the dominant computational overhead of the proposed method, as analyzed in Section 6.3.5. Moreover, the objective function evaluates only the difference in area between the expected and simulated burned regions, which is a relatively coarse and dimensionless measure. While it serves its purpose as an efficient baseline metric, a more rigorous evaluation could be achieved using metrics that capture geometric similarity, such as the Hausdorff distance [138], or a multi-objective function that simultaneously accounts for both area and spatial correspondence between the predicted and theoretical burned shapes.

The applicability of the proposed fine-tuning mechanism is not limited to the Glyfada wildfire case study, indicating potential generalizability across diverse wildfire scenarios. Unlike conventional CA wildfire spread simulation approaches, which rely on fixed parameterizations calibrated for specific experimental conditions, the proposed methodology introduces a dynamic, data-driven calibration framework based on a metaheuristic optimization scheme with demonstrated effectiveness in literature. This allows the CA parameters to be iteratively updated as new observations become available, making the method inherently adaptable to different wildfire scenarios, fuel types, and environmental conditions. It is designed to operate as an integrated component within the crowdsourcing based wildfire simulation framework presented in Chapter 3, where environmental and fire behavior data are dynamically streamed in near real time through in field observations provided by firefighters and other stakeholders [22]. As landscape and environmental conditions dynamically evolve during the wildfire event, the method can be deployed as client-side code on mobile devices to continuously recalibrate any CA model implemented within the simulation software, enabling adaptive and progressively improved simulations.

Chapter 7

Conclusions

This dissertation presented a software framework for near-real-time wildfire simulation driven by crowdsourced geometric observations collected in-field, addressing a critical gap between the dynamic nature of active fire events and the static assumptions underlying conventional simulation software. However, VGI-based data acquisition also presents challenges, including variability in data quality, non constant update intervals, possible human errors, and differences in contributor expertise. Therefore, crowdsourced observations should not be considered a complete replacement for automated sensing systems, or remote-sensing datasources, but rather a complementary data source, especially valuable when network coverage is limited or unreliable. Data validation techniques, contributor reliability assessment, and fusion with other data sources should also be considered for future work to maximize the value of human-generated information.

The framework rests on three tightly coupled contributions that collectively cover a substantial segment of the processing chain from raw volunteer input to adaptive simulation output.

The *Wildfire Simulation Ontology* (WSO) (Chapter 4) established the semantic foundation of the simulation inputs, providing a formal, machine interpretable representation of wildfire environment components and their interrelations. By mapping heterogeneous VGI contributions onto a unified ontological structure, WSO resolves the vocabulary and schema inconsistencies that are inherent to crowdsourced data collection, ensuring that geometrically and semanti-

cally diverse submissions can be normalized into representations that are directly consumable by downstream processing stages. Beyond its immediate role in the wildfire simulation software, WSO constitutes a reusable domain resource that can support interoperability across wildfire management platforms and emergency management platforms, facilitating the integration of institutional data sources alongside volunteer contributions.

The data fusion module (Chapter 5) addressed the record linkage problem arising from the redundancy and geometric inconsistency of independently contributed geometric shapes delineations. By formulating correspondence as a Quadratic Assignment Problem and solving it with an Ant Colony Optimization algorithm operating on spatio-temporally clustered submission sets, the proposed approach produces simplified representations that preserve both local geometric fidelity and global structural consistency across matched contours. The resulting simplified geometry reduces the computational overhead required for geometry processing by the core simulation software. The two-stage DBSCAN clustering, first over the temporal domain and subsequently over the spatial domain within each time window, proved essential to the linkage step, ensuring that ACO correspondence is applied exclusively to geometrically coherent groups of submissions and thereby preventing the contamination of matchings by spatially disjoint observations. Experimental results demonstrate that assignments generated by the ACO algorithm outperform the baseline approach in the majority of record linkage validation cases. It is worth noting, that the method yields fused feature perimeters that more faithfully represent the ground truth than approaches relying on descriptor similarity alone, particularly in scenarios characterized by nonuniform geometric distortion or partial occlusion of the observed perimeter. This improvement is attributed to the modified pheromone update mechanism, which explicitly incorporates the human related characteristics and reliability of individual contributors.

The dynamic parameter optimization component (Chapter 6) addressed the fundamental limitation of static CA simulator calibration [15] by coupling the simulation engine to a continuously operating ACO_R optimization loop. As new fused

user originated observations become available through the DTN module (or any other form of connectivity), ACO_R explores the continuous CA parameter space for configurations that minimize the discrepancy between predicted and theoretical burned estimations, progressively refining its search through updated information. This mechanism enables the simulator to respond autonomously to abrupt environmental transitions, such as wind direction changes, without operator intervention. The integration of ACO_R driven recalibration with the ACO-based fusion module creates a closed feedback loop in which *uniquely identified* observational inputs continuously drive improved simulation accuracy, and simulation outputs can in turn inform the prioritization of future observation efforts.

Several directions remain open for future investigation. The current WSO formalization covers the core components required for simulation, but its extension to encompass suppression resource allocation, atmospheric model coupling, and fire crews locations would substantially broaden its applicability. Extending the ACO-based fusion module to incorporate contributor reliability weights derived from historical submission quality, represents a natural and practically important generalization that should be investigated in future publications. With respect to the ACO_R optimization module, the present formulation is mainly limited by the objective function and the construction of the theoretical envelope. More complex shapes derived from historical wildfire spread patterns should be evaluated, particularly in cases where fire suppression tactics have influenced and disrupted the natural evolution of fire perimeters. Additionally, a large-scale field validation study, conducted in collaboration with wildfire management agencies and integrating submissions from volunteers via the VGI platform, should help to assess the framework’s performance under realistic deployment conditions and to characterize the sensitivity of simulation quality to the volume and spatial distribution of crowdsourced contributions.

The architecture presented in this work lends itself naturally to the implementation of a digital twin [172] for wildfire management, in which crowdsourced observations continuously update a virtual representation of the wildfire environment, enabling real-time fire behavior simulation and dynamic risk assessment for

civilian populations. Similar digital twin paradigms have recently been explored in the literature for integrating real-time data streams with wildfire simulation models to support situational awareness and operational decision-making [173, 174, 175].

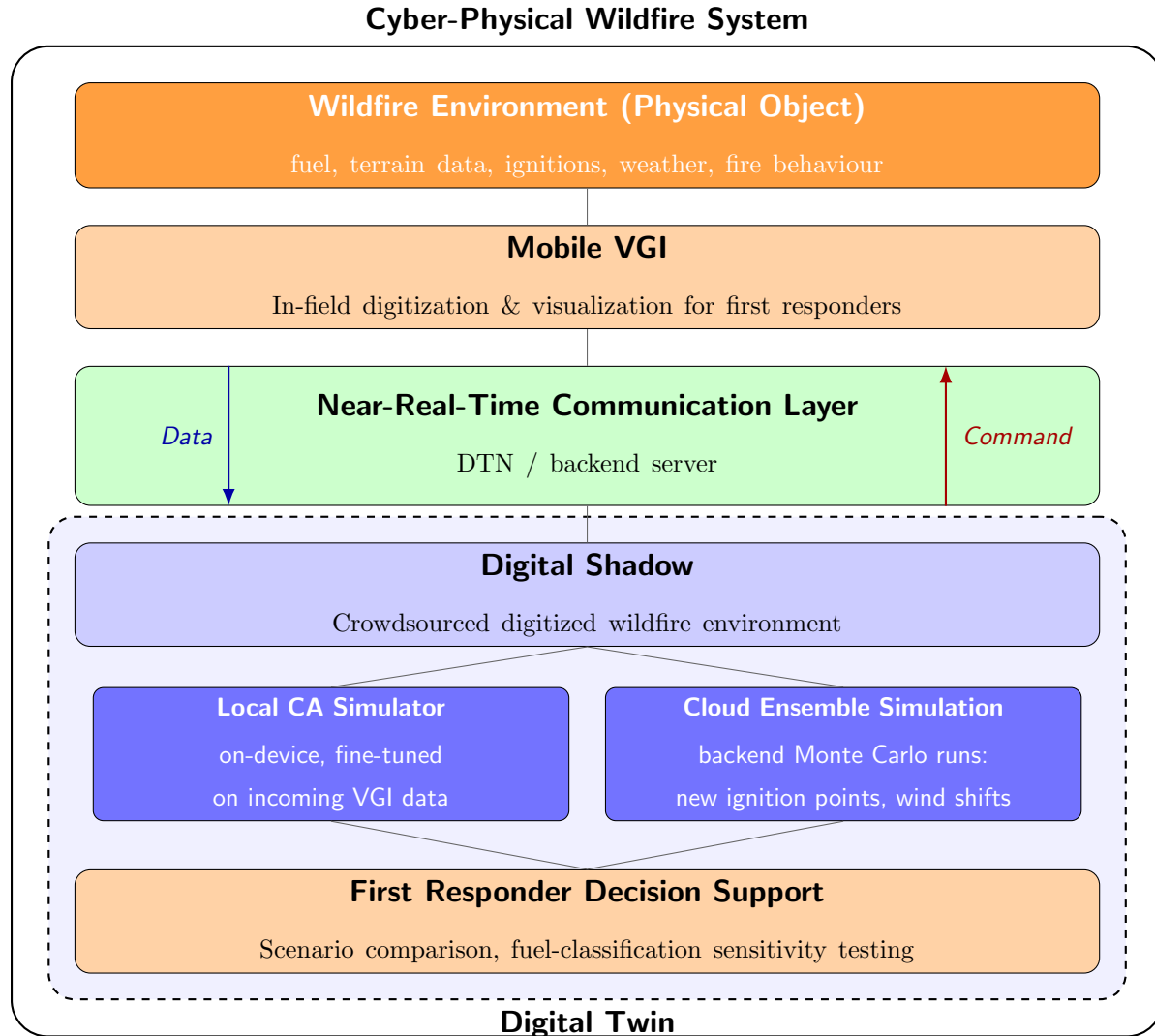


Figure 7.1: Conceptual architecture for a wildfire digital twin as a future extension of the proposed framework. Inspired by [172].

Figure 7.1 illustrates a cyber-physical system, in which crowdsourced data originated from in-field deployed firefighters does more than produce a snapshot of the environment and instead sustains a continuously updated, bidirectional link between the physical fire and its virtual representation. On the ingestion side,

newly digitized fuel and terrain data would be relayed to peers, or to a backend server to recalibrate the underlying CA fire-spread model as conditions evolve, keeping the virtual environment aligned with the ground truth. This recalibrated model could then drive cloud-based ensemble simulations — e.g., Monte Carlo runs over alternative ignition points, shifting wind conditions or newly emerged mobility patterns — giving incident commanders a probabilistic forecast rather than a single deterministic one. Closing the loop, these forecasts would be pushed back to responders in the field to inform operational decisions, whose outcomes in turn generate new VGI input for the next cycle. A lightweight local instance of the simulator would further let individual responders interactively test their own uncertainty — for example, examining how the predicted spread changes if a digitized patch falls ambiguously between two fuel categories — giving them a direct, on-the-ground sense of how much their contributions and classification accuracy matter to the forecast. Together, these extensions would move the system from a digitization tool toward a true digital twin: an automatically and continuously synchronized physical–virtual pair supporting both prediction and real-time decision-making.

Appendix A

Acronyms

ACO Ant Colony Optimization

ACO_R Ant Colony Optimization for Continuous Domains

API Application Programming Interface

BLE Bluetooth Low Energy

CA Cellular Automata

CLC CORINE Land Cover

D2D Device-to-Device

DEM Digital Elevation Model

DOLCE Descriptive Ontology for Linguistic and Cognitive Engineering

DTN Delay Tolerant Networking

EMC Equilibrium Moisture Content

FMC Fuel Moisture Content

GeoJSON Geographic JavaScript Object Notation

ICN Information-Centric Networking

JSON JavaScript Object Notation

JSON-LD JavaScript Object Notation for Linked Data

MCA Macroscopic Cellular Automata

MHD Modified Hausdorff Distance

MODIS Moderate Resolution Imaging Spectroradiometer

NDVI Normalized Difference Vegetation Index

OWL Web Ontology Language

P2P Peer-to-Peer

PWA Progressive Web Application

RDF Resource Description Framework

RDFS RDF Schema

ROS Rate of Spread

SAVI Soil-Adjusted Vegetation Index

SHACL Shapes Constraint Language

SPARQL SPARQL Protocol and RDF Query Language

VGI Volunteered Geographic Information

VPD Vapor Pressure Deficit

WUI Wildland–Urban Interface

WSO Wildfire Simulation Ontology

XML Extensible Markup Language

Bibliography

- [1] J. T. Abatzoglou and A. P. Williams, “Impact of anthropogenic climate change on wildfire across western US forests,” *Proceedings of the National Academy of Sciences*, vol. 113, no. 42, pp. 11 770–11 775, Oct. 2016. [Online]. Available: <https://pnas.org/doi/full/10.1073/pnas.1607171113>
- [2] J.-l. Dupuy, H. Fargeon, N. Martin-StPaul, F. Pimont, J. Ruffault, M. Guijarro, C. Hernando, J. Madrigal, and P. Fernandes, “Climate change impact on future wildfire danger and activity in southern Europe: a review,” *Annals of Forest Science*, vol. 77, no. 2, p. 35, Jun. 2020. [Online]. Available: <https://annforsci.biomedcentral.com/articles/10.1007/s13595-020-00933-5>
- [3] D. M. J. S. Bowman, J. K. Balch, P. Artaxo, W. J. Bond, J. M. Carlson, M. A. Cochrane, C. M. D’Antonio, R. S. DeFries, J. C. Doyle, S. P. Harrison, F. H. Johnston, J. E. Keeley, M. A. Krawchuk, C. A. Kull, J. B. Marston, M. A. Moritz, I. C. Prentice, C. I. Roos, A. C. Scott, T. W. Swetnam, G. R. Van Der Werf, and S. J. Pyne, “Fire in the Earth System,” *Science*, vol. 324, no. 5926, pp. 481–484, Apr. 2009. [Online]. Available: <https://www.science.org/doi/10.1126/science.1163886>
- [4] R. Shakesby and S. Doerr, “Wildfire as a hydrological and geomorphological agent,” *Earth-Science Reviews*, vol. 74, no. 3-4, pp. 269–307, Feb. 2006. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S0012825205001467>
- [5] G. Certini, “Effects of fire on properties of forest soils: a review,” *Oecologia*, vol. 143, no. 1, pp. 1–10, Mar. 2005. [Online]. Available: <http://link.springer.com/10.1007/s00442-004-1788-8>
- [6] L. M. Carbone, J. Tavella, V. Marquez, L. Ashworth, J. G. Pausas, and R. Aguilar, “Fire effects on pollination and plant reproduction: a quantitative review,” *Annals of Botany*, vol. 135, no. 1-2, pp. 43–56, Feb. 2025. [Online]. Available: <https://academic.oup.com/aob/article/135/1-2/43/7618781>

- [7] S. Wagenius, J. Beck, and G. Kiefer, “Fire synchronizes flowering and boosts reproduction in a widespread but declining prairie species,” *Proceedings of the National Academy of Sciences*, vol. 117, no. 6, pp. 3000–3005, Feb. 2020. [Online]. Available: <https://pnas.org/doi/full/10.1073/pnas.1907320117>
- [8] J. E. Keeley, J. G. Pausas, P. W. Rundel, W. J. Bond, and R. A. Bradstock, “Fire as an evolutionary pressure shaping plant traits,” *Trends in Plant Science*, vol. 16, no. 8, pp. 406–411, Aug. 2011. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S1360138511000835>
- [9] D. G. McCullough, R. A. Werner, and D. Neumann, “Fire and Insects in Northern and Boreal Forest Ecosystems of North America,” *Annual Review of Entomology*, vol. 43, no. 1, pp. 107–127, Jan. 1998. [Online]. Available: <https://www.annualreviews.org/doi/10.1146/annurev.ento.43.1.107>
- [10] K. C. Ryan, E. E. Knapp, and J. M. Varner, “Prescribed fire in North American forests and woodlands: history, current practice, and challenges,” *Frontiers in Ecology and the Environment*, vol. 11, no. s1, Aug. 2013. [Online]. Available: <https://esajournals.onlinelibrary.wiley.com/doi/10.1890/120329>
- [11] A. Bento-Gonçalves and A. Vieira, “Wildfires in the wildland-urban interface: Key concepts and evaluation methodologies,” *Science of The Total Environment*, vol. 707, p. 135592, Mar. 2020. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S0048969719355871>
- [12] S. Teymoor Seydi, J. T. Abatzoglou, M. W. Jones, C. A. Kolden, G. Filippelli, M. D. Hurteau, A. AghaKouchak, C. H. Luce, C. Miao, and M. Sadegh, “Increasing global human exposure to wildland fires despite declining burned area,” *Science*, vol. 389, no. 6762, pp. 826–829, Aug. 2025. [Online]. Available: <https://www.science.org/doi/10.1126/science.adu6408>
- [13] X. Ju, S. Zhong, Z. Yue, and W. Zhu, “Dynamic risk assessment of WUI fires based on knowledge-driven approach and scenario simulation,” *International Journal of Disaster Risk Reduction*, vol. 130, p. 105796, Nov. 2025. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S221242092500620X>
- [14] N. Perello, A. Trucchia, F. Baghino, B. Asif, L. Palmieri, N. Rebora, and P. Fiorucci, “Cellular automata-based simulators for the design of prescribed fire plans: the case study of Liguria, Italy,” *Fire Ecology*, vol. 20, no. 1, p. 7, Jan. 2024. [Online]. Available: <https://fireecology.springeropen.com/articles/10.1186/s42408-023-00239-7>
- [15] I. Karakonstantis and G. Xylomenos, “A Review of Two-Dimensional Cellular Automata Models for Wildfire Simulation: Methods, Capabilities, and Limitations,” *Fire*, vol. 9, no. 3, p. 108, Mar. 2026. [Online]. Available: <https://www.mdpi.com/2571-6255/9/3/108>

- [16] —, “Influence of Input Data Uncertainty on Cellular Automata-Based Wildfire Spread Simulation,” *Information*, vol. 17, no. 3, p. 289, Mar. 2026. [Online]. Available: <https://www.mdpi.com/2078-2489/17/3/289>
- [17] M. G. Cruz and M. E. Alexander, “Uncertainty associated with model predictions of surface and crown fire rates of spread,” *Environmental Modelling & Software*, vol. 47, pp. 16–28, 2013. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S1364815213001035>
- [18] A. Cardil, S. Monedero, C. A. Silva, and J. Ramirez, “Adjusting the rate of spread of fire simulations in real-time,” *Ecological Modelling*, vol. 395, no. C, pp. 39–44, None 2019. [Online]. Available: <https://ideas.repec.org/a/eee/ecomod/v395y2019icp39-44.html>
- [19] European Emergency Number Association (EENA), “Public warning,” <https://eena.org/portfolio/public-warning/>, 2020, accessed: 2026-05-20.
- [20] I. Karakonstantis and G. Xylomenos, “A web-based simulation interface for fire dynamics analysis and decision support,” in *Proceedings of the International Conference on Telecommunications (ICT)*, May 2026.
- [21] —, “An Ontology for Dynamic Wildfire Simulations,” in *2025 IEEE Conference on Standards for Communications and Networking (CSCN)*. Bologna, Italy: IEEE, Sep. 2025, pp. 1–6. [Online]. Available: <https://ieeexplore.ieee.org/document/11230697/>
- [22] —, “Improving Wildfire Simulations via Geometric Primitive Analysis in Noisy Crowdsourced Data,” *Applied Sciences*, vol. 15, no. 16, p. 8844, Aug. 2025. [Online]. Available: <https://www.mdpi.com/2076-3417/15/16/8844>
- [23] A. L. Sullivan, “Wildland surface fire spread modelling, 1990 - 2007. 1: Physical and quasi-physical models,” *International Journal of Wildland Fire*, vol. 18, no. 4, p. 349, 2009. [Online]. Available: <http://www.publish.csiro.au/?paper=WF06143>
- [24] —, “Wildland surface fire spread modelling, 1990 - 2007. 3: Simulation and mathematical analogue models,” *International Journal of Wildland Fire*, vol. 18, no. 4, p. 387, 2009. [Online]. Available: <http://www.publish.csiro.au/?paper=WF06144>
- [25] J. G. Freire and C. C. DaCamara, “Using cellular automata to simulate wildfire propagation and to assist in fire management,” *Natural Hazards and Earth System Sciences*, vol. 19, no. 1, pp. 169–179, Jan. 2019. [Online]. Available: <https://nhess.copernicus.org/articles/19/169/2019/>

- [26] A. Alexandridis, D. Vakalis, C. Siettos, and G. Bafas, “A cellular automata model for forest fire spread prediction: The case of the wildfire that swept through Spetses Island in 1990,” *Applied Mathematics and Computation*, vol. 204, no. 1, pp. 191–201, Oct. 2008. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S0096300308004943>
- [27] H. R. Dehkordi, “Mathematical modeling of wildfire propagation in an agricultural land,” in *Proceeding Series of the Brazilian Society of Computational and Applied Mathematics*, Dec. 2021. [Online]. Available: <https://proceedings.sbmac.org.br/sbmac/article/view/3651>
- [28] E. Catchpole, T. Hatton, and W. Catchpole, “Fire spread through nonhomogeneous fuel modelled as a Markov process,” *Ecological Modelling*, vol. 48, no. 1-2, pp. 101–112, Oct. 1989. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/0304380089900628>
- [29] V. Mallet, D. Keyes, and F. Fendell, “Modeling wildland fire propagation with level set methods,” *Computers & Mathematics with Applications*, vol. 57, no. 7, pp. 1089–1101, Apr. 2009. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S0898122108006329>
- [30] S. Behzadi and Z. Mousavi, “A novel agent-based model for forest fire prediction,” *Earth Observation and Geomatics Engineering*, no. Online First, Dec. 2019. [Online]. Available: <https://doi.org/10.22059/eoge.2020.283932.1051>
- [31] J. Katan and L. Perez, “ABWiSE v1.0: toward an agent-based approach to simulating wildfire spread,” *Natural Hazards and Earth System Sciences*, vol. 21, no. 10, pp. 3141–3160, Oct. 2021. [Online]. Available: <https://nhess.copernicus.org/articles/21/3141/2021/>
- [32] X. Zhong, M. Duckham, D. Chong, and K. Tolhurst, “Real-time estimation of wildfire perimeters from curated crowdsourcing,” *Scientific Reports*, vol. 6, no. 1, p. 24206, Jul. 2016. [Online]. Available: <http://www.nature.com/articles/srep24206>
- [33] J. von Neumann, “The general and logical theory of automata,” in *Cerebral Mechanisms in Behaviour*, L. A. Jeffress, Ed. Wiley, 1951.
- [34] X. Xiao, P. Wang, and K.-C. Chou, “Cellular Automata and Its Applications in Protein Bioinformatics,” *Current Protein & Peptide Science*, vol. 12, no. 6, pp. 508–519, Sep. 2011. [Online]. Available: <http://www.eurekaselect.com/openurl/content.php?genre=article&issn=1389-2037&volume=12&issue=6&spage=508>
- [35] F. Jopp, H. Reuter, and B. Breckling, Eds., *Modelling Complex Ecological Dynamics: An Introduction into Ecological Modelling for Students, Teachers*

- Sciencists*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011. [Online]. Available: <https://link.springer.com/10.1007/978-3-642-05029-9>
- [36] A. Alexandridis, D. Vakalis, C. Siettos, and G. Bafas, “A cellular automata model for forest fire spread prediction: The case of the wildfire that swept through Spetses Island in 1990,” *Applied Mathematics and Computation*, vol. 204, no. 1, pp. 191–201, Oct. 2008. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S0096300308004943>
 - [37] A. Alexandridis, L. Russo, D. Vakalis, G. V. Bafas, and C. I. Siettos, “Wildland fire spread modelling using cellular automata: evolution in large-scale spatially heterogeneous environments under fire suppression tactics,” *International Journal of Wildland Fire*, vol. 20, no. 5, p. 633, 2011. [Online]. Available: <http://www.publish.csiro.au/?paper=WF09119>
 - [38] J. G. Freire and C. C. DaCamara, “Using cellular automata to simulate wildfire propagation and to assist in fire management,” *Natural Hazards and Earth System Sciences*, vol. 19, no. 1, pp. 169–179, Jan. 2019. [Online]. Available: <https://nhess.copernicus.org/articles/19/169/2019/>
 - [39] R. C. Rothermel, “A mathematical model for predicting fire spread in wild-land fuels,” Research Paper INT-115, U.S. Department of Agriculture, Intermountain Forest and Range Experiment Station, Ogden, UT, Tech. Rep., 1972.
 - [40] P. A. Martins Fernandes, “Fire spread prediction in shrub fuels in Portugal,” *Forest Ecology and Management*, vol. 144, no. 1-3, pp. 67–74, Apr. 2001. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S0378112700003637>
 - [41] European Union, “Copernicus Land Monitoring Service, European Environment Agency (EEA).” [Online]. Available: <https://land.copernicus.eu/pan-european/corine-land-cover>
 - [42] W. Velasquez, A. Munoz-Arcentales, T. M. Bohnert, and J. Salvachua, “Wildfire Propagation Simulation Tool using Cellular Automata and GIS,” in *2019 International Symposium on Networks, Computers and Communications (ISNCC)*. Istanbul, Turkey: IEEE, Jun. 2019, pp. 1–7. [Online]. Available: <https://ieeexplore.ieee.org/document/8909129/>
 - [43] J. W. Rouse, R. H. Haas, J. A. Schell, and D. W. Deering, “Monitoring vegetation systems in the great plains with ERTS,” in *NASA Goddard Space Flight Center 3d ERTS-1 Symp., Vol. 1, Sect. A*, 1973. [Online]. Available: <https://api.semanticscholar.org/CorpusID:133358670>

- [44] A. Huete, “A soil-adjusted vegetation index (SAVI),” *Remote Sensing of Environment*, vol. 25, no. 3, pp. 295–309, Aug. 1988. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/003442578890106X>
- [45] E. R. Hunt, P. C. Doraiswamy, J. E. McMurtrey, C. S. Daughtry, E. M. Perry, and B. Akhmedov, “A visible band index for remote sensing leaf chlorophyll content at the canopy scale,” *International Journal of Applied Earth Observation and Geoinformation*, vol. 21, pp. 103–112, Apr. 2013. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S0303243412001791>
- [46] A. A. Gitelson, Y. J. Kaufman, R. Stark, and D. Rundquist, “Novel algorithms for remote estimation of vegetation fraction,” *Remote Sensing of Environment*, vol. 80, no. 1, pp. 76–87, Apr. 2002. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S0034425701002899>
- [47] A. Trucchia, M. D’Andrea, F. Baghino, P. Fiorucci, L. Ferraris, D. Negro, A. Gollini, and M. Severino, “PROPAGATOR: An Operational Cellular-Automata Based Wildfire Simulator,” *Fire*, vol. 3, no. 3, p. 26, Jul. 2020. [Online]. Available: <https://www.mdpi.com/2571-6255/3/3/26>
- [48] I. Karafyllidis and A. Thanailakis, “A model for predicting forest fire spreading using cellular automata,” *Ecological Modelling*, vol. 99, no. 1, pp. 87–97, Jun. 1997. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S0304380096019424>
- [49] L. Sun, C. Xu, Y. He, Y. Zhao, Y. Xu, X. Rui, and H. Xu, “Adaptive Forest Fire Spread Simulation Algorithm Based on Cellular Automata,” *Forests*, vol. 12, no. 11, p. 1431, Oct. 2021. [Online]. Available: <https://www.mdpi.com/1999-4907/12/11/1431>
- [50] Z. Wang, “Current forest fire danger rating system (CFFDRS),” in *Proceedings of the Asia-Oceania Symposium on Fire Science & Technology*, no. 1. Fire Safety Science, 1992, pp. 121–125. [Online]. Available: https://publications.iafss.org/publications/aofst/1/121/view/aofst_1-121.pdf
- [51] H. Zhuang, N. Liu, X. Xie, X. Xu, M. Li, Y. Zhang, and R. Wang, “Simulating wildfire spread based on continuous time series remote sensing images and cellular automata,” *International Journal of Wildland Fire*, vol. 34, no. 1, Jan. 2025. [Online]. Available: <https://www.publish.csiro.au/WF/WF24130>
- [52] T. Ghisu, B. Arca, G. Pellizzaro, and P. Duce, “An optimal Cellular Automata algorithm for simulating wildfire spread,” *Environmental Modelling & Software*, vol. 71, pp. 1–14, Sep. 2015. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S1364815215001322>

- [53] J. Glasa and L. Halada, “A Note on Mathematical Modelling of Elliptical Fire Propagation,” *Computing and Informatics*, vol. 30, no. 6, pp. 1303–1319, May 2012. [Online]. Available: <https://www.cai.sk/ojs/index.php/cai/article/view/817>
- [54] M. Alexander, “Estimating the length-to-breadth ratio of elliptical forest fire patterns,” in *Proceedings of the eighth conference on fire and forest meteorology*, vol. 29. Detroit, MI, USA: Society of American Foresters, 1985, pp. 84–85.
- [55] D. Jaeggi, G. Parks, T. Kipouros, and P. Clarkson, “The development of a multi-objective tabu search algorithm for continuous optimisation problems,” *European Journal of Operational Research*, vol. 185, no. 3, p. 1192 – 1212, 2008, cited by: 147.
- [56] F. M. Finney, “FARSITE: A fire area simulator for fire managers,” *Weise, D. R.; Martin, R. E. (tech. coords.). The Biswell Symposium: Fire issues and solutions in urban interface and wildland ecosystems*, 1995.
- [57] M. A. Finney, “Farsite: Fire area simulator—model development and evaluation,” USDA Forest Service, Rocky Mountain Research Station, Ogden, Utah, USA, Tech. Rep. Research Paper RMRS-RP-4, 1998.
- [58] G. A. Trunfio, D. D’Ambrosio, R. Rongo, W. Spataro, and S. Di Gregorio, “A New Algorithm for Simulating Wildfire Spread through Cellular Automata,” *ACM Transactions on Modeling and Computer Simulation*, vol. 22, no. 1, pp. 1–26, Dec. 2011. [Online]. Available: <https://dl.acm.org/doi/10.1145/2043635.2043641>
- [59] S. D. Gregorio and R. Serra, “An empirical method for modelling and simulating some complex macroscopic phenomena by cellular automata,” *Future Generation Computer Systems*, vol. 16, no. 2-3, pp. 259–271, Dec. 1999. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S0167739X99000515>
- [60] C. Weinhouse and J. Augustin, “Leveraging Cellular Automata for Real-Time Wildfire Spread Modeling in California,” Oct. 2025, arXiv:2510.09708 [physics]. [Online]. Available: <http://arxiv.org/abs/2510.09708>
- [61] W. Jiang, F. Wang, L. Fang, X. Zheng, X. Qiao, Z. Li, and Q. Meng, “Modelling of wildland-urban interface fire spread with the heterogeneous cellular automata model,” *Environmental Modelling & Software*, vol. 135, p. 104895, Jan. 2021. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S136481522030952X>

- [62] S. Zhao, “Simulation of Mass Fire-Spread in Urban Densely Built Areas Based on Irregular Coarse Cellular Automata,” *Fire Technology*, vol. 47, no. 3, pp. 721–749, Jul. 2011. [Online]. Available: <http://link.springer.com/10.1007/s10694-010-0187-4>
- [63] K. Kalabokidis, N. Athanasis, and M. Vaitis, “OntoFire: an ontology-based geo-portal for wildfires,” *Natural Hazards and Earth System Sciences*, vol. 11, no. 12, pp. 3157–3170, Dec. 2011. [Online]. Available: <https://nhess.copernicus.org/articles/11/3157/2011/>
- [64] M. Iacovou, S. Kontogiannis, and K. Avgerinakis, “Ontology data insights and alerts for wildfire protection,” in *Proceedings of the 13th Hellenic Conference on Artificial Intelligence*, ser. SETN '24. New York, NY, USA: Association for Computing Machinery, 2024. [Online]. Available: <https://doi.org/10.1145/3688671.3688737>
- [65] M. F. Goodchild and J. A. Glennon, “Crowdsourcing geographic information for disaster response: a research frontier,” *International Journal of Digital Earth*, vol. 3, no. 3, pp. 231–241, Sep. 2010. [Online]. Available: <https://www.tandfonline.com/doi/full/10.1080/17538941003759255>
- [66] Y. R. Shi, R. C. Levy, L. A. Remer, S. Mattoo, and G. T. Arnold, “Investigating the spatial and temporal limitations for remote sensing of wildfire smoke using satellite and airborne imagers during firex-aq,” *Journal of Geophysical Research: Atmospheres*, vol. 129, no. 2, p. e2023JD039085, 2024, e2023JD039085 2023JD039085. [Online]. Available: <https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2023JD039085>
- [67] R. S. Allison, J. M. Johnston, G. Craig, and S. Jennings, “Airborne optical and thermal remote sensing for wildfire detection and monitoring,” *Sensors*, vol. 16, no. 8, 2016. [Online]. Available: <https://www.mdpi.com/1424-8220/16/8/1310>
- [68] M. F. Goodchild and L. Li, “Assuring the quality of volunteered geographic information,” *Spatial Statistics*, vol. 1, pp. 110–120, 2012. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S2211675312000097>
- [69] H. Li, B. Zhao, and A. Fuxman, “The wisdom of minority: discovering and targeting the right group of workers for crowdsourcing,” in *Proceedings of the 23rd International Conference on World Wide Web*, ser. WWW '14. New York, NY, USA: Association for Computing Machinery, 2014, p. 165–176. [Online]. Available: <https://doi.org/10.1145/2566486.2568033>
- [70] Y. Li, J. Gao, C. Meng, Q. Li, L. Su, B. Zhao, W. Fan, and J. Han, “A survey on truth discovery,” *SIGKDD Explor. Newsl.*, vol. 17, no. 2, p. 1–16, Feb. 2016. [Online]. Available: <https://doi.org/10.1145/2897350.2897352>

- [71] J. O'Brien, M. Serra, A. Hudson-Smith, S. Psarra, A. Hunter, and M. Zaltz-Austwick, "Ensuring vgi credibility in urban-community data generation: A methodological research design," *Urban Planning*, vol. 1, no. 2, pp. 88–100, 2016. [Online]. Available: <https://www.cogitatiopress.com/urbanplanning/article/view/620>
- [72] N. Bogdos and E. S. Manolakis, "Crowd-Sourced Wildfire Spread Prediction With Remote Georeferencing Using Smartphones," *IEEE Access*, vol. 7, pp. 102 102–102 112, 2019. [Online]. Available: <https://ieeexplore.ieee.org/document/8777073/>
- [73] —, "A tool for simulation and geo-animation of wildfires with fuel editing and hotspot monitoring capabilities," *Environmental Modelling & Software*, vol. 46, pp. 182–195, Aug. 2013. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S1364815213000728>
- [74] X. Fu, Y. Hu, P. Sutrave, P. A. Beerel, and B. Raghavan, "FireLoc: Low-latency Multi-modal Wildfire Geolocation," in *Proceedings of the 22nd ACM Conference on Embedded Networked Sensor Systems*. Hangzhou China: ACM, Nov. 2024, pp. 1–14. [Online]. Available: <https://dl.acm.org/doi/10.1145/3666025.3699318>
- [75] A. C. Oliveira, L. C. Botega, J. F. Saran, J. N. Silva, J. O. Melo, M. F. Tavares, and V. P. Neris, "Crowdsourcing, data and information fusion and situation awareness for emergency Management of forest fires: The project DF100Fogo (FDWithoutFire)," *Computers, Environment and Urban Systems*, vol. 77, p. 101172, Sep. 2019. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S0198971517303241>
- [76] H. Shachaf, C. Price, D. Rostkier-Edelstein, and C. Mass, "On the potential of using smartphone sensors for wildfire hazard estimation through citizen science," *Natural Hazards and Earth System Sciences*, vol. 24, no. 9, pp. 3035–3047, Sep. 2024. [Online]. Available: <https://nhess.copernicus.org/articles/24/3035/2024/>
- [77] R. Seager, A. Hooks, A. P. Williams, B. Cook, J. Nakamura, and N. Henderson, "Climatology, Variability, and Trends in the U.S. Vapor Pressure Deficit, an Important Fire-Related Meteorological Quantity," *Journal of Applied Meteorology and Climatology*, vol. 54, no. 6, pp. 1121–1141, Jun. 2015. [Online]. Available: <https://journals.ametsoc.org/view/journals/apme/54/6/jamc-d-14-0321.1.xml>
- [78] W. Zhong, X. Mei, F. Niu, X. Fan, S. Ou, and S. Zhong, "Fusing Social Media, Remote Sensing, and Fire Dynamics to Track Wildland-Urban Interface Fire," *Remote Sensing*, vol. 15, no. 15, p. 3842, Aug. 2023. [Online]. Available: <https://www.mdpi.com/2072-4292/15/15/3842>

- [79] A. Sasa and M. Krisper, “Ontology-based knowledge management in service-oriented systems,” *Journal of Information and Organizational Sciences*, vol. 35, no. 1, Jun. 2011. [Online]. Available: [//jios.foi.hr/index.php/jios/article/view/197](http://jios.foi.hr/index.php/jios/article/view/197)
- [80] J. Florath, J. Chanussot, and S. Keller, “Utilizing Volunteered Geographic Information for Real-Time Analysis of Fire Hazards: Investigating the Potential of Twitter Data in Assessing the Impacted Areas,” *Fire*, vol. 7, no. 1, p. 6, Dec. 2023. [Online]. Available: <https://www.mdpi.com/2571-6255/7/1/6>
- [81] A. Sathiyamoorthy, W. Zhou, X. Zhou, X. Li, and I. Gondal, “MFISP: A Multimodal Fire Spread Prediction Framework,” Oct. 2025, arXiv:2510.23934 [cs]. [Online]. Available: <http://arxiv.org/abs/2510.23934>
- [82] NASA LANCE FIRMS, “NASA Fire Information for Resource Management System (FIRMS),” 2025, near real-time active fire data. Accessed: 2026-04-09. [Online]. Available: <https://earthdata.nasa.gov/firms>
- [83] M. A. Finney, “Fire growth using minimum travel time methods,” *Canadian Journal of Forest Research*, vol. 32, no. 8, pp. 1420–1424, Aug. 2002. [Online]. Available: <http://www.nrcresearchpress.com/doi/10.1139/x02-068>
- [84] G. Xylomenos, C. N. Ververidis, V. A. Siris, N. Fotiou, C. Tsilopoulos, X. Vasilakos, K. V. Katsaros, and G. C. Polyzos, “A Survey of Information-Centric Networking Research,” *IEEE Communications Surveys & Tutorials*, vol. 16, no. 2, pp. 1024–1049, 2014. [Online]. Available: <http://ieeexplore.ieee.org/document/6563278/>
- [85] G. Xylomenos, G. Pavlou, I. Psaras, and I. Karakonstantis, “Named Functions at the Edge,” in *2019 IEEE Symposium on Computers and Communications (ISCC)*. Barcelona, Spain: IEEE, Jun. 2019, pp. 1–6. [Online]. Available: <https://ieeexplore.ieee.org/document/8969730/>
- [86] V. Agafonkin, “Leaflet: an open-source javascript library for mobile-friendly interactive maps,” 2011, accessed: 2026-05-15. [Online]. Available: <https://leafletjs.com/>
- [87] F. Isayama and T. Shigeyasu, “A study on DTN routing protocol for disaster situation while considering age of information,” *Internet of Things*, vol. 34, p. 101756, Nov. 2025. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S2542660525002690>
- [88] H. Chenji and R. Stoleru, “Delay-tolerant networks (DTNs) for emergency communications,” in *Advances in Delay-Tolerant Networks (DTNs)*. Elsevier, 2021, pp. 103–134. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/B9780081027936000060>

- [89] K. Fall and S. Farrell, "DTN: an architectural retrospective," *IEEE Journal on Selected Areas in Communications*, vol. 26, no. 5, pp. 828–836, Jun. 2008. [Online]. Available: <http://ieeexplore.ieee.org/document/4530739/>
- [90] S. T. Shah, S. F. Hasan, B.-C. Seet, P. H. J. Chong, and M. Y. Chung, "Device-to-Device Communications: A Contemporary Survey," *Wireless Personal Communications*, vol. 98, no. 1, pp. 1247–1284, Jan. 2018. [Online]. Available: <http://link.springer.com/10.1007/s11277-017-4918-4>
- [91] I. Horrocks, P. F. Patel-Schneider, and F. van Harmelen, "From shiq and rdf to owl: the making of a web ontology language," *Journal of Web Semantics*, vol. 1, no. 1, pp. 7–26, 2003. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S1570826803000027>
- [92] T. Tudorache, "Ontology engineering: Current state, challenges, and future directions," *Semantic Web*, vol. 11, no. 1, pp. 125–138, 2020.
- [93] T. R. Gruber, "Toward principles for the design of ontologies used for knowledge sharing?" *International Journal of Human-Computer Studies*, vol. 43, no. 5-6, pp. 907–928, Nov. 1995. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S1071581985710816>
- [94] M. Kavouras, A. Darra, M. Kokla, S. Kontaxaki, G. Panopoulos, and E. Tomai, *Geographic Information Science - An Integrated Approach and Special Topics*. Association of Greek Academic Libraries, 2015.
- [95] G. Babitski, S. Bergweiler, O. Grebner, D. Oberle, H. Paulheim, and F. Probst, "Soknos – using semantic technologies in disaster management software," in *The Semantic Web: Research and Applications*, G. Antoniou, M. Grobelnik, E. Simperl, B. Parsia, D. Plexousakis, P. De Leenheer, and J. Pan, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 183–197.
- [96] H. E. Anderson, "Aids to determining fuel models for estimating fire behavior," U.S. Department of Agriculture, Forest Service, Intermountain Forest and Range Experiment Station, Ogden, UT, Tech. Rep. INT-GTR-122, 1982. [Online]. Available: <https://www.fs.usda.gov/treearch/pubs/6447>
- [97] J. H. Scott and R. E. Burgan, "Standard fire behavior fuel models: a comprehensive set for use with Rothermel's surface fire spread model," U.S. Department of Agriculture, Forest Service, Rocky Mountain Research Station, Ft. Collins, CO, Tech. Rep. RMRS-GTR-153, 2005. [Online]. Available: <https://www.fs.usda.gov/treearch/pubs/9521>
- [98] E. Hyvönen, "Preventing ontology interoperability problems instead of solving them," *Semantic Web*, vol. 1, no. 1-2, pp. 33–37, 2010. [Online]. Available: <https://journals.sagepub.com/doi/abs/10.3233/SW-2010-0014>

- [99] J. Portisch, M. Hladik, and H. Paulheim, “Background knowledge in ontology matching: A survey,” *Semantic Web*, vol. 15, no. 6, pp. 2639–2693, 2022.
- [100] G. Karvounarakis, S. Alexaki, V. Christophides, D. Plexousakis, and M. Scholl, “RQL: a declarative query language for RDF,” in *Proceedings of the 11th international conference on World Wide Web*. Honolulu Hawaii USA: ACM, May 2002, pp. 592–603. [Online]. Available: <https://dl.acm.org/doi/10.1145/511446.511524>
- [101] P. Masa, G. Meditskos, S. Kintzios, S. Vrochidis, and I. Kompatsiaris, “Ontology-based Modelling and Reasoning for Forest Fire Emergencies in Resilient Societies,” in *Proceedings of the 12th Hellenic Conference on Artificial Intelligence*. Corfu Greece: ACM, Sep. 2022, pp. 1–9. [Online]. Available: <https://dl.acm.org/doi/10.1145/3549737.3549765>
- [102] “Shapes constraint language (SHACL),” W3C, Tech. Rep., Jul. 2017. [Online]. Available: <https://www.w3.org/TR/shacl/>
- [103] H. Paulheim, S. Döweling, K. Tso, F. Probst, and T. Ziegert, “Improving usability of integrated emergency response systems: The soknos approach,” in *INFORMATIK 2009 - Im Focus das Leben, Beiträge der 39. Jahrestagung der Gesellschaft für Informatik e. V. (GI)*, vol. 154, 01 2009, pp. 1435–1349.
- [104] A. Karakostas, S. Vrochidis, Y. Kompatsiaris, B. Kantsepolsky, J. Moßgraber, S. Dasiopoulou, B. Mandler, A. Karppinen *et al.*, “beAWARE: Enhancing decision support and management services in extreme weather climate events,” in *15th International Conference on Information Systems for Crisis Response and Management (ISCRAM)*, 2018.
- [105] E. Kontopoulos, P. Mitzias, J. Moßgraber, P. Hertweck, H. van der Schaaf, D. Hilbring, F. Lombardo, D. Norbiato, M. Ferri, A. Karakostas, S. Vrochidis, and I. Kompatsiaris, “Ontology-based representation of crisis management procedures for climate events,” May 2018. [Online]. Available: <https://doi.org/10.5281/zenodo.1243535>
- [106] M. Couturier and E. Wilkinson, “Open advanced system for improved crisis management (OASIS),” in *Proceedings of the 2nd International Conference on Information Systems for Crisis Response and Management (ISCRAM 2005)*, Brussels, Belgium, Apr. 2005.
- [107] M. de la Asunción, L. Castillo, J. Fdez-Olivares, O. García-Pérez, A. González, and F. Palao, “Siadex: An interactive knowledge-based planner for decision support in forest fire fighting,” *AI Commun.*, vol. 18, no. 4, p. 257–268, Dec. 2005.

- [108] D. S. Nau, T. C. Au, O. Ilghami, U. Kuter, J. W. Murdock, D. Wu, and F. Yaman, “Shop2: An htn planning system,” *Journal of Artificial Intelligence Research*, vol. 20, p. 379–404, Dec. 2003. [Online]. Available: <http://dx.doi.org/10.1613/jair.1141>
- [109] Z. Yue, W. Zhu, X. Mei, and S. Zhong, “An integrated graph-spatial method for high-performance geospatial-temporal semantic query,” *International Journal of Applied Earth Observation and Geoinformation*, vol. 137, p. 104437, 2025. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S1569843225000846>
- [110] K. Kyzirakos, M. Karpathiotakis, G. Garbis, C. Nikolaou, K. Bereta, I. Papoutsis, T. Herekakis, D. Michail, M. Koubarakis, and C. Kontoes, “Wildfire monitoring using satellite images, ontologies and linked geospatial data,” *Journal of Web Semantics*, vol. 24, pp. 18–26, 2014, the Semantic Web Challenge 2012. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S1570826814000031>
- [111] R. G. Raskin and M. J. Pan, “Knowledge representation in the semantic web for earth and environmental terminology (sweet),” *Computers & Geosciences*, vol. 31, no. 9, pp. 1119–1125, 2005, application of XML in the Geosciences. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0098300405001020>
- [112] K. Bereta, G. Xiao, and M. Koubarakis, “Answering geosparql queries over relational data,” *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. XLII-4/W2, pp. 43–50, 2017. [Online]. Available: <https://isprs-archives.copernicus.org/articles/XLII-4-W2/43/2017/>
- [113] K. Kyzirakos, M. Karpathiotakis, K. Bereta, G. Garbis, C. Nikolaou, P. Smeros, S. Giannakopoulou, K. Dogani, and M. Koubarakis, “The spatiotemporal rdf store strabon,” in *Advances in Spatial and Temporal Databases*, M. A. Nascimento, T. Sellis, R. Cheng, J. Sander, Y. Zheng, H.-P. Kriegel, M. Renz, and C. Sengstock, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 496–500.
- [114] D. Kolas, “Supporting Spatial Semantics with SPARQL,” *Transactions in GIS*, vol. 12, pp. 5–18, Dec. 2008. [Online]. Available: <https://onlinelibrary.wiley.com/doi/10.1111/j.1467-9671.2008.01132.x>
- [115] P.-Y. Chen, C.-W. Lien, F.-J. Chu, P.-S. Ting, and S.-M. Cheng, “Supervised collective classification for crowdsourcing,” in *2015 IEEE Globecom Workshops (GC Wkshps)*, 2015, pp. 1–6.

- [116] T. N. T. Tran, A. Felfernig, and V. M. Le, “An overview of consensus models for group decision-making and group recommender systems,” *User Model. User-adapt Interact.*, vol. 34, no. 3, pp. 489–547, Jul. 2024.
- [117] T. Dokeroglu, E. Sevinc, T. Kucukyilmaz, and A. Cosar, “A survey on new generation metaheuristic algorithms,” *Computers & Industrial Engineering*, vol. 137, p. 106040, 2019. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0360835219304991>
- [118] K. Bereta and M. Koubarakis, “Ontop of geospatial databases,” in *The Semantic Web – ISWC 2016*, P. Groth, E. Simperl, A. Gray, M. Sabou, M. Krötzsch, F. Lecue, F. Flöck, and Y. Gil, Eds. Cham: Springer International Publishing, 2016, pp. 37–52.
- [119] M. E. Alexander and M. G. Cruz, “Limitations on the accuracy of model predictions of wildland fire behaviour: A state-of-the-knowledge overview,” *The Forestry Chronicle*, vol. 89, no. 03, pp. 372–383, Jun. 2013. [Online]. Available: <http://pubs.cif-ifc.org/doi/10.5558/tfc2013-067>
- [120] S. Lohmann, V. Link, E. Marbach, and S. Negru, “Webvowl: Web-based visualization of ontologies,” in *Knowledge Engineering and Knowledge Management: EKAW 2014 Satellite Events, VISUAL, EKM1, and ARCOE-Logic, Linköping, Sweden, November 24-28, 2014. Revised Selected Papers*. Berlin, Heidelberg: Springer-Verlag, 2014, p. 154–158. [Online]. Available: https://doi.org/10.1007/978-3-319-17966-7_21
- [121] J. Bleiholder and F. Naumann, “Data fusion,” *ACM Comput. Surv.*, vol. 41, no. 1, Jan. 2009. [Online]. Available: <https://doi.org/10.1145/1456650.1456651>
- [122] —, “Conflict handling strategies in an integrated information system,” in *Proceedings of the IJCAI Workshop on Information on the Web (IIWeb)*, 2006.
- [123] A.-C. Mailleux, C. Detrain, and J.-L. Deneubourg, “Triggering and persistence of trail-laying in foragers of the ant *Lasius niger*,” *Journal of Insect Physiology*, vol. 51, no. 3, pp. 297–304, Mar. 2005. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S0022191004002100>
- [124] M. Dorigo and T. Stützle, *Ant Colony Optimization*. The MIT Press, Jun. 2004. [Online]. Available: <https://direct.mit.edu/books/book/2313/Ant-Colony-Optimization>
- [125] I. Karakonstantis and A. Vlachos, “The ant colony optimisation solving continuous problems,” *International Journal of Computational Intelligence Studies*, vol. 2, no. 3/4, p. 350, 2013. [Online]. Available: <http://www.inderscience.com/link.php?id=57637>

- [126] O. Van Kaick, G. Hamarneh, H. Zhang, and P. Wighton, "Contour Correspondence via Ant Colony Optimization," in *15th Pacific Conference on Computer Graphics and Applications (PG'07)*. Maui, HI, USA: IEEE, Oct. 2007, pp. 271–280. [Online]. Available: <http://ieeexplore.ieee.org/document/4392737/>
- [127] E. L. Lawler, "The quadratic assignment problem," *Management Science*, vol. 9, no. 4, pp. 586–599, 1963. [Online]. Available: <https://doi.org/10.1287/mnsc.9.4.586>
- [128] S. Belongie, J. Malik, and J. Puzicha, "Shape matching and object recognition using shape contexts," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 4, pp. 509–522, Apr. 2002. [Online]. Available: <http://ieeexplore.ieee.org/document/993558/>
- [129] A. Berg, T. Berg, and J. Malik, "Shape matching and object recognition using low distortion correspondences," in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, vol. 1, 2005, pp. 26–33 vol. 1.
- [130] E. Schubert, J. Sander, M. Ester, H. P. Kriegel, and X. Xu, "DBSCAN Revisited, Revisited: Why and How You Should (Still) Use DBSCAN," *ACM Transactions on Database Systems*, vol. 42, no. 3, pp. 1–21, Sep. 2017. [Online]. Available: <https://dl.acm.org/doi/10.1145/3068335>
- [131] H. W. Kuhn, "The hungarian method for the assignment problem," *Naval Research Logistics Quarterly*, vol. 2, no. 1-2, pp. 83–97, 1955. [Online]. Available: <https://onlinelibrary.wiley.com/doi/abs/10.1002/nav.3800020109>
- [132] R. Burkard, M. Dell'Amico, and S. Martello, *Assignment Problems*. USA: Society for Industrial and Applied Mathematics, 2009.
- [133] L. A. K. Ayad and S. P. Pissis, "MARS: improving multiple circular sequence alignment using refined sequences," *BMC Genomics*, vol. 18, no. 1, p. 86, Jan. 2017.
- [134] M. Dorigo, *Ant colony optimization*. MIT Press, 2007, vol. 2, arXiv: 1307.4186v1 Publication Title: Scholarpedia Issue: 3. [Online]. Available: http://www.scholarpedia.org/article/Ant_colony_optimization
- [135] J. Cohen, "The wildland-urban interface fire problem," *Fremontia*. 38 (2)-38 (3): 16-22., pp. 16–22, 2010.
- [136] W. E. Mell, S. L. Manzello, A. Maranghides, D. Butry, and R. G. Rehm, "The wildland–urban interface fire problem—current approaches and research needs," *International Journal of Wildland Fire*, vol. 19, no. 2, pp. 238–251, 2010.

- [137] O. van Kaick, “Contour Correspondence via Ant Colony Optimization.” [Online]. Available: <https://www.mathworks.com/matlabcentral/fileexchange/24094-contour-correspondence-via-ant-colony-optimization>
- [138] M.-P. Dubuisson and A. Jain, “A modified Hausdorff distance for object matching,” in *Proceedings of 12th International Conference on Pattern Recognition*, vol. 1. Jerusalem, Israel: IEEE Comput. Soc. Press, 1994, pp. 566–568. [Online]. Available: <http://ieeexplore.ieee.org/document/576361/>
- [139] H. Alt, B. Behrend, and J. Blömer, “Approximate matching of polygonal shapes,” *Annals of Mathematics and Artificial Intelligence*, vol. 13, no. 3-4, pp. 251–265, Sep. 1995. [Online]. Available: <http://link.springer.com/10.1007/BF01530830>
- [140] H. Mitchell, S. Gwynne, E. Ronchi, N. Kalogeropoulos, and G. Rein, “Integrating wildfire spread and evacuation times to design safe triggers: Application to two rural communities using PERIL model,” *Safety science*, vol. 157, p. 105914, 2023.
- [141] T. J. Cova, P. E. Dennison, T. H. Kim, and M. A. Moritz, “Setting wildfire evacuation trigger points using fire spread modeling and GIS,” *Transactions in GIS*, vol. 9, no. 4, pp. 603–617, 2005.
- [142] M. Hajian, E. Melachrinoudis, and P. Kubat, “Modeling wildfire propagation with the stochastic shortest path: A fast simulation approach,” *Environmental Modelling & Software*, vol. 82, pp. 73–88, Aug. 2016. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S1364815216300767>
- [143] Y. Liu, H. Liu, Y. Zhou, and C. Sun, “Spread vector induced cellular automata model for real-time crown fire behavior simulation,” *Environmental Modelling & Software*, vol. 108, pp. 14–39, Oct. 2018. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S1364815217305613>
- [144] M. Kim, C. Pais, and M. C. Gonzalez, “Fire spread simulations using Cell2Fire on synthetic and real landscapes,” *Scientific Reports*, vol. 15, no. 1, p. 25173, Jul. 2025. [Online]. Available: <https://www.nature.com/articles/s41598-025-05706-6>
- [145] L. Hernández Encinas, S. Hoya White, A. Martín Del Rey, and G. Rodríguez Sánchez, “Modelling forest fire spread using hexagonal cellular automata,” *Applied Mathematical Modelling*, vol. 31, no. 6, pp. 1213–1227, Jun. 2007. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S0307904X06000916>

- [146] G. D. Richards, “A general mathematical framework for modelling two-dimensional wildland fire spread,” *International Journal of Wildland Fire*, vol. 5, no. 2, p. 63 – 72, 1995, cited by: 101.
- [147] R. Hooke and T. Jeeves, ““direct search” solution of numerical and statistical problems,” *Journal of the ACM (JACM)*, vol. 8, no. 2, p. 212 – 229, 1961, cited by: 3524; All Open Access, Bronze Open Access.
- [148] E. Couce and W. Knorr, “Statistical parameter estimation for a cellular automata wildfire model based on satellite observations,” in *Modelling, Monitoring and Management of Forest Fires II*, ser. FIVA10, vol. 1. WIT Press, May 2010, p. 47–55. [Online]. Available: <http://dx.doi.org/10.2495/FIVA100051>
- [149] K. I. M. McKinnon, “Convergence of the nelder–mead simplex method to a nonstationary point,” *SIAM Journal on Optimization*, vol. 9, no. 1, pp. 148–158, 1998. [Online]. Available: <https://doi.org/10.1137/S1052623496303482>
- [150] K. Clarke, J. Brass, and P. Riggan, “A cellular automaton model of wild-fire propagation and extinction,” *Photogrammetric Engineering and Remote Sensing*, vol. 60, no. 11, pp. 1355–1367, Nov. 1994.
- [151] A. J. Simard, “The moisture content of forest fuels - I: a review of the basic concepts,” Government of Canada, Department of Forestry and Rural Development, Forest Fire Research Institute, Ottawa, Ontario, Tech. Rep. Information Report FF-X-14, 1968.
- [152] J. D. Cohen and J. E. Deeming, “The national fire-danger rating system: basic equations,” U.S. Department of Agriculture, Forest Service, Pacific Southwest Forest and Range Experiment Station, Berkeley, CA, Tech. Rep. PSW-GTR-82, 1985. [Online]. Available: <https://www.fs.usda.gov/treearch/pubs/27298>
- [153] R. M. Nelson Jr., “A method for describing equilibrium moisture content of forest fuels,” *Canadian Journal of Forest Research*, vol. 14, no. 4, pp. 597–600, Aug. 1984. [Online]. Available: <http://www.nrcresearchpress.com/doi/10.1139/x84-108>
- [154] T. Hu, L. Ma, Y. Gao, J. Fan, and L. Sun, “Modification and Comparison of Methods for Predicting the Moisture Content of Dead Fuel on the Surface of *Quercus mongolica* and *Pinus sylvestris* var. *mongolica* under Rainfall Conditions,” *Fire*, vol. 6, no. 10, p. 379, Oct. 2023. [Online]. Available: <https://www.mdpi.com/2571-6255/6/10/379>
- [155] E. A. Catchpole, W. R. Catchpole, N.R.Viney, W. L. McCaw, and J. B. Marsden-Smedley, “Estimating fuel response time and

- predicting fuel moisture content from field data,” *International Journal of Wildland Fire*, vol. 10, no. 2, pp. 215–222, Jul. 2001. [Online]. Available: <https://connectsci.au/wf/article/10/2/215/22564/Estimating-fuel-response-time-and-predicting-fuel>
- [156] J. K. Hiers, C. L. Stauhammer, J. J. O’Brien, H. L. Gholz, T. A. Martin, J. Hom, and G. Starr, “Fine dead fuel moisture shows complex lagged responses to environmental conditions in a saw palmetto (*Serenoa repens*) flatwoods,” *Agricultural and Forest Meteorology*, vol. 266–267, pp. 20–28, Mar. 2019. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S0168192318303927>
- [157] I. Karakonstantis and G. Xylomenos, “Optimizing fire perimeter geometry in cellular automata wildfire models using ant colony optimization,” *Fire*, vol. 9, no. 5, 2026. [Online]. Available: <https://www.mdpi.com/2571-6255/9/5/212>
- [158] I. Karakonstantis and A. Vlachos, “Ant Colony Optimization for Continuous Domains applied to Emission and Economic Dispatch Problems,” *Journal of Information and Optimization Sciences*, vol. 36, no. 1–2, pp. 23–42, Mar. 2015. [Online]. Available: <http://www.tandfonline.com/doi/full/10.1080/02522667.2014.932094>
- [159] K. Socha and M. Dorigo, “Ant colony optimization for continuous domains,” *European Journal of Operational Research*, vol. 185, no. 3, pp. 1155–1173, 2008, iSBN: 0377-2217.
- [160] G. Bilchev and I. C. Parmee, “The ant colony metaphor for searching continuous design spaces,” in *Evolutionary Computing*, T. C. Fogarty, Ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 1995, pp. 25–39.
- [161] N. Monmarché, G. Venturini, and M. Slimane, “On how *Pachycondyla apicalis* ants suggest a new search algorithm,” *Future Generation Computer Systems*, vol. 16, no. 8, pp. 937–946, Jun. 2000. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S0167739X00000479>
- [162] J. Dréo and P. Siarry, “Continuous interacting ant colony algorithm based on dense heterarchy,” *Future Generation Computer Systems*, vol. 20, no. 5, pp. 841–856, Jun. 2004. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S0167739X03002139>
- [163] L. Liu, Y. Dai, and J. Gao, “Ant Colony Optimization Algorithm for Continuous Domains Based on Position Distribution Model of Ant Colony Foraging,” *The Scientific World Journal*, vol. 2014, pp. 1–9, 2014. [Online]. Available: <http://www.hindawi.com/journals/tswj/2014/428539/>

- [164] G. Leguizamón and C. A. C. Coello, “An alternative acor algorithm for continuous optimization problems,” in *Proceedings of the 7th International Conference on Swarm Intelligence*, ser. ANTS’10. Berlin, Heidelberg: Springer-Verlag, 2010, p. 48–59.
- [165] J. G. De Freitas and K. Yamanaka, “An accelerated and robust algorithm for ant colony optimization in continuous functions,” *Journal of the Brazilian Computer Society*, vol. 27, no. 1, p. 16, Dec. 2021. [Online]. Available: <https://journal-bcs.springeropen.com/articles/10.1186/s13173-021-00116-8>
- [166] I. Knight and J. Coleman, “A Fire Perimeter Expansion Algorithm-Based on Huygens Wavelet Propagation,” *International Journal of Wildland Fire*, vol. 3, no. 2, pp. 73–84, Jun. 1993. [Online]. Available: <https://connectsci.au/wf/article/3/2/73/22003/A-Fire-Perimeter-Expansion-Algorithm-Based-on>
- [167] C. E. V. Wagner, “A Simple Fire-Growth Model,” *The Forestry Chronicle*, vol. 45, no. 2, pp. 103–104, Apr. 1969. [Online]. Available: <http://pubs.cif-ifc.org/doi/10.5558/tfc45103-2>
- [168] H. E. Anderson, *Predicting wind-driven wild land fire size and shape*. Research Paper INT-RP-305. Ogden, UT: USDA Forest Service, Intermountain Forest and Range Experiment Station. 26 p., 1983.
- [169] FireLab, “Elliptical Dimensions - Technical Documentation,” FlamMap Online Help, Missoula Fire Sciences Laboratory, U.S. Forest Service, Tech. Rep., Aug. 2025. [Online]. Available: https://owfflammaphelp62.firenet.gov/LegacyFARSITE/Tech_Elliptical_Dimensions.htm
- [170] A. Simard, G. Young, and R. Redmond, “Airpro - an Air Tanker Productivity Computer Simulation Model - The equations (Documentation),” Forest Fire Research Institute, Canadian Forestry Service, Department of Fisheries and the Environment, Tech. Rep., 1978, information Report FF-X-66.
- [171] P. Andrews, C. Bevins, and R. Seli, *BehavePlus fire modeling system, version 4.0: user’s guide. General Technical Report, RMRS-GTR-106WWW Revised*. Ogden, UT: USDA Forest Service, Rocky Mountain Research Station, 2008.
- [172] N. Julien and E. Martin, “How to characterize a digital twin: A usage-driven classification,” *IFAC-PapersOnLine*, vol. 54, no. 1, pp. 894–899, 2021, 17th IFAC Symposium on Information Control Problems in Manufacturing INCOM 2021. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S2405896321008557>

- [173] Y. Huang, J. Li, and H. Zheng, “Modeling of wildfire digital twin: Research progress in detection, simulation, and prediction techniques,” *Fire*, vol. 7, no. 11, 2024. [Online]. Available: <https://www.mdpi.com/2571-6255/7/11/412>
- [174] M. H. Raha, A. R. Tavakkoli, C. Webb, M. Habibpour, J. Coen, E. Rowell, and F. Afghah, “Firewin: Digital twin advancing multi-modal sensing, interactive analytics for wildfire response,” 2025. [Online]. Available: <https://arxiv.org/abs/2510.18879>
- [175] M. Alhazmi, O. P. Li, D. Huo, H. Zhang, and Z. Bao, “Can digital twin technology revolutionize wildfire management and energy resilience in los angeles?” *Energy Reports*, vol. 14, pp. 3118–3131, 2025. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S2352484725005244>