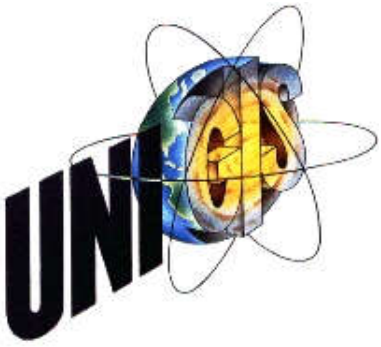


Master Thesis

submitted within the UNIGIS MSc. programme
at the Department of Geoinformatics - Z_GIS
University of Salzburg, Austria
under the provisions of UNIGIS joint study programme with
Kathmandu Forestry College (KAFCOL), Kathmandu, Nepal

WebGIS based Wildfire Simulation

By

Bishrant Adhikari

GIS_104195

A thesis submitted in partial fulfilment of the requirements of
the degree of
Master of Science (Geographical Information Science & Systems) – MSc (GISc)

Advisor (s):

Dr. Shahnawaz Shahnawaz

Kathmandu, January 2017

Science Pledge

By my signature below, I certify that my project report is entirely the result of my own work. I have cited all sources of information and data I have used in my project report and indicated their origin.

Kathmandu, January 22, 2017

Place and Date	Signature

Acknowledgements

I would like to give my sincere thanks to my immediate advisor Dr. Shahnawaz for his great assistance in pursuit of my whole UNIGIS course as well as this research. His constructive comments and suggestions were invaluable for completion of this thesis.

I am grateful to Mr. Durgendra Man Kayastha, who has always been my well-wisher and source of inspiration for guiding me during the early days of my graduate studies. I would like to thank Dr. Ambika Prasad Gautam and Dr. Ram Asheshwor Mandal for their continious support and assistance during my whole time in this Msc. program.

The constructive comments and suggestions that i got from my classmate Mr. Atul Man Joshi were very helpful for completing this course. I would also like to thank Mr. Khimlal Gautam for his help and kind suggestions.

I am highly indebted to all the instructors and professors who put their faith on me and urged me to do better.

Lastly, i would be nowhere near this without the immense help, support and motivation from my family. I cannot express my gratitude towards my Late GrandFather, my GrandMother, Mom, Dad and Brother for their forgiveness and sacrificing their luxury for my happiness.

Bishrant Adhikari

January, 2017

Abstract:

Despite several ecological advantages and being a necessary component in human civilization, wildfires have been causing tremendous loss of life and property. Wildfires have been widely regarded as one of the severe disaster in the United States. Proper management of fire requires scientific knowledge about the fire and its expected behavior. Wildfire modelling is typically associated with the numerical simulation of wildland fires with the purpose of understanding and predicting the fire behaviour. Wildfire modelling facilitates fire suppression activities by providing information about the rate of spread and intensity of fire. Typically, wildfire modelling could be categorized into two components namely fire behavior and fire spread models.

Wildfires are highly dynamic events which is directly affected by continuously changing environmental conditions such as wind speed, wind direction and humidity. Thus, for predicting fire behavior and spread, it is highly imperative to use updated dataset. Simulation of fire in real-time provides most-reliable information to the users that could be used for proper allocation of resources as well as devising evacuation plans for risk prone areas.

Despite several attempts to simulate wildfires accurately, there has been a lack of real-time modelling system that could automatically respond to updated information. This research explains the need for developing WebGIS based wildfire modelling system and its possible areas of applications. A full-fledged WebGIS based wildfire modelling system is developed using open-sources algorithms that could satisfactorily simulate wildfire spread in near-real time basis. This system utilizes client server architecture and does not require users to install anything on their machine. The data download and model re-initialization is performed automatically by the system without the need of user input. The interactive GUI could be used by users connected via internet to simulate wildfire spread on their desired area for their chosen

time steps. This system tightly integrates GIS with fire modelling to provide users with seamless experience.

Wildfire spread equation developed by Rothermel (1972) along with Huygen's wavelet propagation theory is used to determine where and when the fire is going to spread. The model has been revised per the findings of Albini (1976) to calculate fire behavior in the form of fire intensity and rate of spread. Custom developed Python scripts are used to quantify the wildfire behavior. It also uses QGIS libraries for Python to perform basic GIS analysis tasks such as dissolve and clip.

The results obtained from this model was tested against the results obtained from FARSITE (Fire Area Simulator) under similar conditions. FARSITE has been a standard wildfire modelling system in several federal organization such as United States Forest Service and National Park service as it has been found to predict wildfire behavior fairly accurately under homogenous fuel conditions. Upon comparison with FARSITE, the model produced satisfactory results. Although, some of the predictions were greater than that of FARSITE, the general shape and behavior of wildfire in both cases were almost similar. This discrepancy could be due to omission of Crown Fire simulation and Spotting model.

This study concludes that WebGIS based wildfire simulation could be a great tool for predicting wildfire spread in near-real time. The adoption of open-end architecture and open-source system allows for future improvements as well. This system could be improvised by including both Crown fire model and spotting model using faster computation resources for simulating wildfire accurately and in an efficient way.

Table of Contents

Science Pledge ii
Acknowledgements iii
Abstract: iv
Table of Contents vi
List of Tables x
List of Figures xi
List of Maps xii
Abbreviations xiii
Chapter 1: Introduction 1
1.1. Background 1
1.2. Brief History of fire modelling 2
1.3. Environmental modelling in GIS 3
1.4. WebGIS and Fire Modelling 5
1.5. Rationale 8
1.6. Hypothesis 9
1.7. Study Area 9
1.8. Research Objectives 10
1.9. Research steps and methods 11
1.9.1. Research and quantification of wildfire behaviour 11
1.9.2. Development of wildfire spread model using wildfire behaviour inputs 11
1.9.3. Presentation and comparison of results 12
1.10. Expected Results 12
1.11. Assumptions and Limitations 12
1.12. Contribution to Knowledge 13
1.13. Outline of Thesis 14
Chapter 2: Principles of Wildfire Modelling 16
2.1. Introduction 16
2.2. Wildfire Behavior 16
2.2.1. Definition 16
2.2.2. Origins of wildfire 16
2.2.3. Factors affecting spread of wildfires 17
2.3. Wildfire spread models: 21
2.3.1. Crown fire models: 22
2.3.2. Surface fire models: 22

2.3.3. Ground fire models 22
2.3.4. Spotting wildfire models: 23
2.4. Fire Behavior 24
2.4.1. The rate of Spread (ROS): 24
2.4.2. Fireline Intensity (FLI): 25
2.4.3. Flame Size: 26
2.4.4. Heat Per Unit Area (HPA): 27
2.5. Types of mathematical wildfire models 27
2.5.1. Stochastic models: 27
2.5.2. Deterministic models: 28
2.5.3. Probabilistic models: 29
2.6. Fire Spread Simulations: 30
2.6.1. Vector based approach using Huygen's Wavelet Principle: 31
2.6.2. Raster Based simulation: 34
2.6.3. Other propagation methods: 37
2.7. Existing Fuel and Fire Behaviour modelling systems: 38
2.7.1. FARSITE 38
2.7.2. FlamMap 39
2.7.3. BehavePlus 39
2.7.4. NEXUS 40
2.7.5. FMAPlus 40
2.7.6. Virtual Fire 40
2.7.7. WIFIRE 41
Chapter 3: Development of Wildfire Model 42
3.1. Introduction: 42
3.2. Fire Model development guidelines: 42
3.2.1. Data acquisition and evaluation: 42
3.2.2. Calculation of fire variables: 42
3.2.3. Interpretation of fire variables and generation of secondary indices 43
3.2.4. Visualisation and inference of results 43
3.3. Choice of type fire model type 43
3.4. Development of Fire spread models: 44
3.4.1. Raster vs. Vector based approach: 44
3.5. Fire spread simulation using Huygen's principle 45
3.5.1. Slope and Wind Factors: 47
3.5.2. Elliptical dimension: 48
3.6. Development of Fire behaviour models 49
3.6.1. Fire Intensity: 49
3.6.2. The rate of spread: 50
3.6.3. Fuel Models 52
3.6.4. Fuel particle moisture content (M_f) 54
Chapter 4: Design and Development of WebGIS based Wildfire Modelling System 59

4.1. WebGIS and Fire modelling: 59
4.2. Data acquisition and management: 60
4.2.1. Identification of data sources: 61
4.2.1. Topographic Data 62
4.2.2. Fuels data: 62
4.2.3. Live Fuel Moisture: 63
4.2.4. Weather Data: 64
4.2.5. Other Data 66
4.2.6. Summary of acquired data sets: 67
4.2.7. Acquisition of data in near real time: 67
4.3. Data processing and Validation: 68
4.4. Development of WebGIS based wildfire model 68
4.4.1. System Architecture: 68
4.4.2. Integration of wildfire modelling system in Python: 72
4.4.3. Visualisation of results: 74
Chapter 5: Evaluation and Discussion 75
5.1. Introduction 75
5.2. Choice of FARSITE for comparison: 75
5.3. Simulation Results 79
5.3.1. Sample Site-1 79
5.3.2. Sample Site-2: 82
5.3.3. Sample Site-3 85
5.3.4. Sample Site-4 88
5.3.5. Sample Site-5 91
5.3.6. Sample Site-6 94
5.3.7. Sample Site-7 97
5.3.8. Sample Site-8 100
5.3.9. Summary of observation from sample sites 103
5.4. Visualisation of results with time slider 104
5.5. Discussion 104
Chapter 6: Conclusion and Recommendation 106
6.1. Conclusion 106
6.1.1. To develop wildfire behavior model 106
6.1.2. To develop wildfire spread model 106
6.1.3. Verification and comparison of results with standardized model results 106
6.1.4. Visualization of the modelling and simulation results 107
6.1.5. Discussion 107
6.2. Problems encountered and Limitations 108
6.3. Recommendations and Future improvements 109
6.3.1. Utilization of better resolution input data 109
6.3.2. Incorporation of Crown and Spotting fire models 109
6.3.3. Testing with real wildfire data 110
6.3.4. Utilization of parallel computing 110

References 111
Appendix.. 119

List of Tables

Table 3- 1 NFDRS fuel model 52

Table 3- 2 Albini 13 Fuel Model 53

Table 4- 1 Required datasets for fire modelling and their purpose and sources 61

Table 4- 2 MODIS bands used for NDWI calculation..... 63

Table 4- 3 Summary of acquired data sets..... 67

Table 5- 1 Sample site and fuel types 78

Table 5- 2 Sample site-1 summary and modelling conditions..... 79

Table 5- 3 Sample site-1 comparative results 81

Table 5- 4 Sample site-2 summary and modelling conditions..... 83

Table 5- 5 Sample site-2 comparative results 84

Table 5- 6 Sample site-3 summary and modelling conditions..... 86

Table 5- 7 Sample site-3 comparative results 87

Table 5- 8 Sample site-4 summary and modelling conditions..... 89

Table 5- 9 Sample site-4 comparative results 90

Table 5- 10 Sample site-5 summary and modelling conditions..... 92

Table 5- 11 Sample site-5 comparative results 93

Table 5- 12 Sample site-6 summary and modelling conditions..... 95

Table 5- 13 Sample site-6 comparative results 96

Table 5- 14 Sample site-7 summary and modelling conditions..... 98

Table 5- 15 Sample site-7 comparative results 99

Table 5- 16 Sample site-8 summary and modelling conditions..... 101

Table 5- 17 Sample site-8 comparative results 102

Table 5- 18 Summary of modelling results from sample sites 104

List of Figures

Figure 2- 1 Ground, Surface and Crown fires along with ladder effect 23

Figure 2- 2 Fire Spotting phenomenon 24

Figure 2- 3 Flame size in terms of flame height, length and flame depth 26

Figure 2- 4 Elliptical representation of wildfire spread 32

Figure 2- 5 Huygen's principle for steady wind V adapted for fire spread showing elliptical fire spread and creation of new fire front at time $t_0 + t$ 33

Figure 2- 6 Huygen's elliptical expansion in fire spread (grey ellipses) starting from red dot (ignition point) 33

Figure 2- 7 CA cell neighbourhood 35

Figure 2- 8 Fire spread and direction of the wind in CA method 35

Figure 3- 1 Elliptical wavelet dimensions in fire spread 46

Figure 3- 2 Guidelines for estimating live fuel (foliage) moisture content 55

Figure 3- 3 Architecture of wildfire model 58

Figure 4- 1 System Architecture 69

Figure 4- 2 Overview of system processes 71

Figure 5- 1 Modelling result in WebGIS interface: Site-1 82

Figure 5- 2 Modelling result in WebGIS interface: Site-2 85

Figure 5- 3 Modelling result in WebGIS interface: Site-3 88

Figure 5- 4 Modelling result in WebGIS interface: Site-4 91

Figure 5- 5 Modelling result in WebGIS interface: Site-5 94

Figure 5- 6 Modelling result in WebGIS interface: Site-6 97

Figure 5- 7 Modelling result in WebGIS interface: Site-7 100

Figure 5- 8 WebGIS simulation result for Site-8 103

List of Maps

Map 1- 1 Study Area 10

Map 5- 1 Distribution of sample sites 78

Map 5- 2 Intersecting area between FARSITE and model results, Site-1 80

Map 5- 3 Comparison of Modelling result with FARSITE result: Site-1 81

Map 5- 4 Intersecting area between FARSITE and model results, Site-2 83

Map 5- 5 Comparison of Modelling result with FARSITE result 84

Map 5- 6 Intersecting area between FARSITE and model results, Site-3 86

Map 5- 7 Comparison of model result with FARSITE, Sample site-3 87

Map 5- 8 Intersecting area between FARSITE and model results, Site-4 89

Map 5- 9 Comparison of Modelling result with FARSITE result: Site-4 90

Map 5- 10 Intersecting area between FARSITE and model results, Site-5 92

Map 5- 11 Comparison of Modelling result with FARSITE result: Site-5 93

Map 5- 12 Intersecting area between FARSITE and model results, Site-6 95

Map 5- 13 Comparison of Modelling result with FARSITE result: Site-6 96

Map 5- 14 Intersecting area between FARSITE and model results, Site-7 98

Map 5- 15 Comparison of Modelling result with FARSITE result: Site-7 99

Map 5- 16 Intersecting area between FARSITE and model results, Site-8 101

Map 5- 17 Comparison of Modelling result with FARSITE result: Site-8 102

Abbreviations

AAN	Artificial Neural Network
BLM	Bureau of Land Management
CA	Cell Automata
DEM	Digital Elevation Model
EMBYR	Ecological Model for Burning the Yellowstone Region
FBP	Fire Behaviour Prediction
FDRS	Fire Danger Rating System
FI	Fire Intensity
GDAL	Geospatial Data Abstraction Library
GFMC	Global Fire Monitoring Center
GIS	Geographical Information System
GPS	Global Positioning System
GUI	Graphical User Interface
HPA	Heat released per-unit area
HTML	Hyper Text Markup Language
HTTP	Hyper Text Transfer Protocol
LANDFIRE	Landscape Fire and Resource Management Planning Tools Project
MODIS	Moderate-Resolution Imaging Spectroradiometer
NAD	North American Datum
NDFRS	National Fire Danger Rating system
NDVI	Normalized Difference Vegetation Index
NDWI	Normalized Difference Water Index
NPS	National Park Service
NWCG	National Wildfire Coordinating Group
NWS	National Weather Service
OGC	Open Geospatial Consortium
ROS	Rate of Spread
USA	United States of America
USDA	United States Department of Agriculture
USFS	United States Forest Service
USGS	United States Geological Survey
UTC	Coordinated Universal Times
UTM	Universal Transverse Mercator

Chapter 1: Introduction

1.1. Background

The term wildfire refers to an uncontrolled fire having a high capability of burning larger area in a short duration. Wildfire primarily refers to fires igniting especially on the countryside. Fires have been existent from the beginning of human evolution and have been widely accepted as a valuable natural resource/phenomenon responsible for driving the ecosystem forward. Fires have been found to help the growth of healthy vegetation and at the same time removing unwanted dead vegetation and providing essential nutrients to the soil. Despite these and other advantages of fires, they pose a stronger threat to the environment, the life of living beings and the properties as well. Forest fires are exponentially ravaging several parts of the world over the decades due to climate and human geography (Pausas, Llovet, Rodrigo, & Vallejo, 2009).

During the late 1870s and 1920, massive wildfires spread across millions of acres and destroyed towns and caused thousands of fatalities (Pyne, 1982). States such as Wisconsin, Michigan, Montana, Minnesota, Idaho, Washington, Oregon, South Carolina and Illinois saw the greater destruction of lives and property than last 50 years. Since 1985, wildfires in the wildland-urban interface has been recognised as a major problem thus leading to the drafting of policies such as National Fire Plan in 2000, Federal Wildland Fire Management Policy (2001), Firewise programs and Healthy forests restoration act (2003). The current focus on wildfire management is on increasing suppression effects in WUI so that the vast amount of money being spent makes tangible benefits to the affected community (Dillon, Menakis, & Fay, 2014). Although there has been a recent emphasis on "*fire use*" the majority of fires are suppressed as compared to those allowed to burn. Up until today, the "*fire exclusion paradigm*" still continues. These exclusion paradigms have resulted in changes in vegetation-fuel structures and fuel accumulation leading to more high-intensity fires. However, since prescribed burns do not occur naturally, we need to understand how changes in weather conditions and other variations in topography along

with others are affecting the fire behaviour. Thus, for an informed decision making in fire management, it is highly imperative to adopt wildfire modelling.

Current focus domains on wildfires are in conflagrations caused by lightning discharges or acts of man in wilderness areas close enough to one or more urban interfaces that they threaten people, property, infrastructure and business enterprise. These wilderness areas typically contain national forests, national parks or resorts and the adjacent towns containing large, expensive homes. Notable hazards put forth by wildfires are Fire, hot gases, and smoke, burned-out slopes (with increased susceptibility to insect infestation, erosion, and landslides). Wildfires also result in local changes in air quality and weather conditions (Liu, Kahn, Chaloulakou, & Koutrakis, 2009).

Wildfire modelling is typically associated with the numerical simulation of wildland fires with the purpose of understanding and predicting the fire behaviour. It supports wildland fire suppression activities especially the safety of firefighters and the public who are near or in the active fire risk region. Simulation of fire in real time will give them most reliable information about the possible nature of the fire which could be used for the proper allocation of resources and even plan evacuations before it's too late. Using the forecasted data in such models would even allow us to roughly predict the nature of fire at some point in future.

1.2. Brief History of fire modelling

Attempts have been made to make most comprehensive fire modelling and simulation models starting from the beginning of the 19th century. Although clearly in practice for environmental modelling, fire modelling, does not have a clear distinction between mathematical and computational models and can be broadly classified into stochastic models, deterministic and probabilistic models (Sibolla, 2009). These models will be discussed in detail in Chapter 2. Most of the models are developed for a particular area and are often not suitable for direct implementation in a different area other than which it has been prepared for.

Out of several attempts in fire modelling, the works are done by Byram (1959), and Rothermel (1972) laid the foundation of using mathematical equations to predict wildfire behaviour. Later with the development of computerised models by Albini (1982) increased the practicality of wildfire modelling by a huge margin. Then the adoption of fuel models, availability of weather data and development of wildfire modelling software such as FARSITE, FSPRO, BEHAVE and BEHAVEPLUS closely integrated fire behaviour study in several federal organisations and even private institutions such as insurance companies. The advent of GIS and graphical user interface increased the user's productivity by reducing the learning curve heavily. Different notable systems and software in the domain of wildfire modelling are discussed in detail in chapter 2.

1.3. Environmental modelling in GIS

A multitude of environmental phenomenon such as flooding, landslide, wildfire, biological processes along with atmospheric systems is currently implementing GIS to model them (Clarke, Parks, & Crane, 2002). Although they vary in different scales, complexity, the amount of data and their approach to implementation of GIS, they are attempting the represent the complex nature of earth's processes using models and equations. The presence of several sub-components which are further comprised of smaller components makes the modelling of environmental systems a very complex and challenging task. Thus, it has been agreed across the scientific community that the environmental models should can produce "good" or "reasonable" results in contrary to the cent percent "perfect" results. Also since a model is an abstraction of reality that is limited by the number of factors taken into consideration depending on the modeller's degree of knowledge, the accurate results from a model are barely expected (Brimicombe, 2003).

The model, however, should be able to represent the dynamic nature of environmental phenomenon across different levels of heterogeneity. Also, Clarke, Parks, & Crane (2002) argues that it should be able to make use of clearly define contemporary events and make scientific inferences about the possible future behaviour of the

phenomenon being modelled. Brimicombe (2003) mentions that a model should provide the users with a mechanism of visualisation of complex phenomenon along with their interactions. Brimicombe (2003) also states the model should be based on scientific theories that could either bring new knowledge or could test the validity and possibly extend the existing theory. A model typically uses variables and parameters to represent and predict environmental phenomenon. Out of different types of models such as hardware models, natural models, computational and mathematical models identified by Brimicombe (2003), the computational and mathematical models are the most widely used ones in wildfire domain. Computational and mathematical models use equations, functions to use existing data to predict wildfire behaviour often with the help of a computer or other kinds of computation machines. The detailed discussion about different types of models in wildfire domain is discussed in Chapter 2. Primarily the selection of the model is guided by the kind of phenomenon being modelling and the expertise of the modeller in that domain.

A modelling task begins with the identification of the problem, study and quantification of variables affecting that phenomenon and later developing the whole model using this knowledge. A modeller should clearly define the scale he/she is working on and the assumptions made for that task along with the parameters that are considered.

Initially, during early 1990s environmental modelling and GIS were considered isolated with each other with each one performing different duties. The integration of GIS in modelling started with a loose coupling strategy during the 1990s where GIS was used as a visualisation and data processing tool (Brimicombe, 2003). GIS was not regarded as being capable of representing the complexity and dynamic nature of an environmental phenomenon such a wildfire. Later during 2000, a concept called tight coupling was developed where both GIS and modelling capabilities were accessible via a standard interface which was often graphical. They used to store data in common location that could be accessed by both GIS and environmental model. The latest development in GIS integration with environmental modelling is known as embedded systems where environmental models are built right into GIS system. Thus, all the GIS functionality is

readily available to the models such as access to scripting tools, GUI and visualisation features.

With the current development of processing capabilities of computer and their ability to handle the immense amount of data from diverse sources, the integration of GIS and environmental models have been even intense and widely implemented. Nowadays Geocomputation is used which utilises the higher speed of computer systems to perform typical modelling tasks such as data analysis, modelling, simulation and visualisation of spatial data (Câmara, & Monteiro, 2001). The basic GIS functionality is nowadays being extended using programming tools and scripting to deliver highly customised and user-centric modelling and simulation tools for the users. These models provide spatial computation and visualisation capability on top of standard GIS functionality. Users nowadays do not need to worry about understanding the complexity of the models to execute them.

Wildfire is a dynamic phenomenon as its behaviour is changing throughout its spread duration (Trollope, Trollope, & Hartnett, 2002). Also as the wildfire progresses, it could encounter with different kinds of surfaces and environmental conditions such as bare ground, dense forest, residential areas, roads or water surfaces. Thus it is highly imperative that the model should be tightly coupled with GIS to provide the efficient way to process, analyse and visualise wildfire data and its properties. The use of programming tools will further enhance the user experience by proper utilisation of Geocomputation for utilisation of near real-time inputs and near real-time outputs.

1.4. WebGIS and Fire Modelling

The current advent of Internet and World Wide Web has led to a worldwide adoption of WebGIS in their existing GIS systems (Zheng, Rahim, & Pan, 2000). Often there exists some degree of confusion about the different terminologies that are used to refer to an online facet of GIS. The notable ones are Internet GIS as noted by Peng (1999), GIS online by Plewe (1997), Web-based GIS or simply referred to as WebGIS by Grunwald, Reddy,

Mathiwalagan and Bloom in 2003. Ever since, the term WebGIS has been commonly used to refer to a mechanism of using the internet to perform spatial analysis, exchange data and visualise/present the results to the users. It adopts a client-server computing architecture which is widely adopted in World Wide Web domain. There exist two types of WebGIS architecture namely map services and data services (Mathiyalagan, Grunwald, Reddy, & Bloom, 2005). The map services allow the users to visualise the information on the web without downloading it onto their local machines whereas data services provide the actual spatial and non-spatial data to the users who can then perform analysis, visualisation and other tasks on their local machines.

The common aspect in different implementations of WebGIS system is that it allows the users to use the processing capability of the server to without having to install complex systems and software on their machines. One of the primary advantages of using WebGIS system is it allows users to access several functionalities independent of their platform, devices and operating systems. In most of the cases, a standard web browser that is World Wide Web Consortium (W3C) complaint is enough to make use of the system. One of the notable advantages of WebGIS is its ability to update information in real time. The users could see the changes in the form of maps, charts and tables as soon as the data in the server is updated. This has also allowed the ingestion of newer data from smartphones, sensor devices and so on to provide updated information to the users. The WebGIS systems automatically run the processing as soon as they receive an update, so the newer results could be produced instantly by the server and disseminated to the users. Currently, the WebGIS systems are widely being developed for mobile devices such as tablets, smartphones and GPS devices which have made the field inspection and spatial data collection and visualisation even more intuitive.

The tremendous cost of acquiring a desktop GIS is often eliminated by a cheaper or even free WebGIS system that performs some of the essential functionality needed to the user without having to buy them(Alesheikh, Helali, & Behroz, 2002). The dissemination of spatial information has been even easier than before. WebGIS has satisfied the user's

demand for mobility and convenience in spatial data analysis and visualisation by providing them access to these tools from anywhere with just internet connection.

WebGIS has changed the way a traditional GIS would typically function in a right way. They have made the GIS functionality accessible to the non-GIS educated users. These users now could quickly generate spatial insights about their data and spatial phenomenon without having core technical GIS knowledge. Even with a short history of around two and half decades, WebGIS has managed to facilitate open accessibility and democratisation of spatial data as well as its effective dissemination (Dragicevic, 2004). Dragicevic (2004) also mentions that the advanced cartographic and geospatial visualisation tools along with interactive map have further increased their usability. The users could now pan and zoom around their maps. They could further extract more information from the features using identify functionality. The advanced query and filtering feature now available in most of the leading WebGIS systems such as ArcGIS online, CartoDB WebGIS interface has now allowed advanced users to perform sophisticated analysis.

Decision making has been greatly benefited from the advent of WebGIS. Planners could readily execute models, simulate future scenario and generate insights from their data in almost real time. The planners could better orient their resources using WebGIS platforms. The monitoring and inspection of the projects and their progress have also been a piece of cake as the higher level personnel do not need to delve deeper into the technicality of the model and its inputs.

In the event of a wildfire, the user of WebGIS can immensely assist in the management of emergency responses and allocation of resources. It could also help the planners to estimate the damage potential of a particular wildfire under specific conditions. Firefighting agencies will be the most benefited users of WebGIS system if it is implemented in wildfire modelling and simulation. They could devise their plan to tackle the fire based on its predicted behaviour in the future. If properly implemented a WebGIS based wildfire modelling could save lives and minimise the loss of properties and infrastructure.

Despite these advantages, WebGIS also has its limitations. Typically a higher speed internet connection is required for proper functioning of the system because of its intensive use of graphics. Also, the cost associated with hosting and maintaining the server is often a major setback for implementing organisations with limited resources. The deployment of WebGIS also requires the expertise of both spatial development and information technology as well as networking.

1.5. Rationale

Different agencies are dealing actively with the fire activities. For appropriately dealing with fire events, the complexity of the system along with the interaction among various components should be properly understood. Despite the development of a multitude of modelling and simulation systems, there has been a lack of adequate information and modelling systems. Also, these software have a steep learning curve and requires complex setup processes, making it tough for ordinary users to access these functionalities. Since fire is a major concern to not only the experts but also the peoples, the risk of wildfire perceived by them should be better conveyed to them.

Different types of methodologies are used for computing and presenting wildfire behaviour. Some of the systems use their embedded GIS viewer to view the computation results whereas others rely on secondary GIS software to visualise the results. Despite the presence of a multitude of GIS-based wildfire modelling systems, none of them had been found to have been implemented in near real-time wildfire modelling and simulation. The systems existed so far have been more focused for experts and required the installation of the system on their machines. Some of the very first wildfire models have been implemented on the web which simulates the wildfire based on cell-automata based approach. These methods do not consider the effect of wind, topography and fuel inputs which are spatially variable in the real world. Such models and systems are just for illustration purposes and have almost no resemblance to the real world fire behaviour. Also, they typically simulate

fire using a randomly generated grid that is supposed to reflect real world which is not the case at most of the cases.

Thus the development of WebGIS based wildfire modelling system will allow the users to visualise and understand wildfire behaviour without any particular wildfire modelling expertise. The close integration of GIS in the modelling system further facilitates the user's interaction and usability.

1.6. Hypothesis

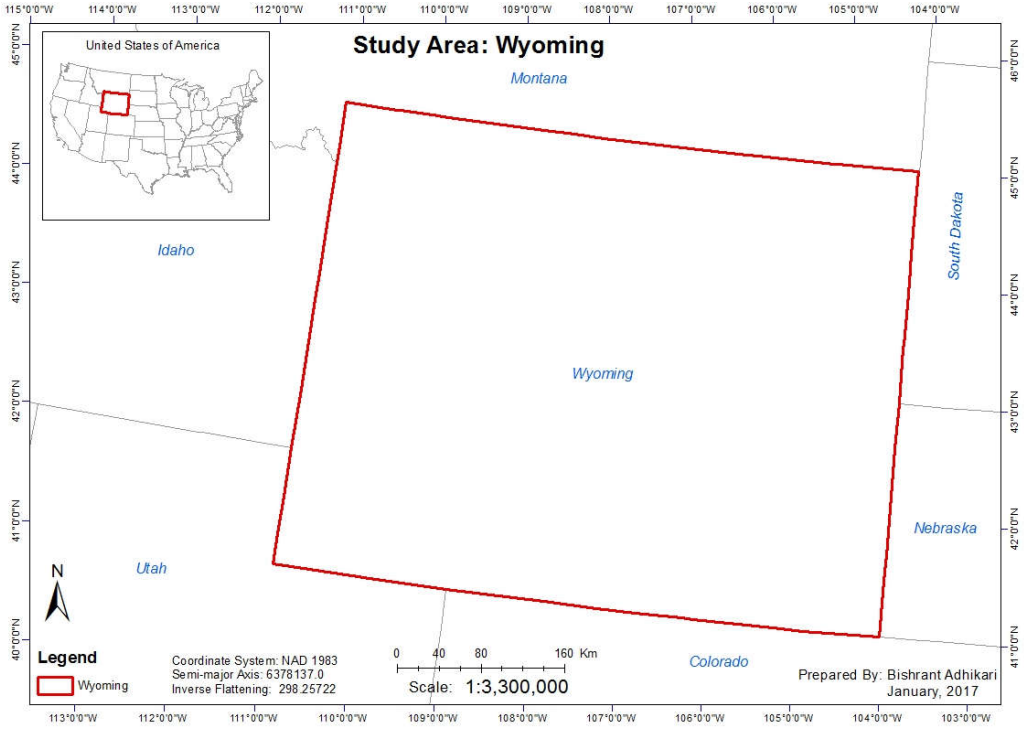
The hypothesis being tested in this study is that implementation of WebGIS platform for wildfire modelling leverages the current limitations of desktop based wildfire modelling systems. Upon successful completion, this architecture is believed to allow users to perform wildfire modelling through the web-based interface without having to install complex and resource heavy software on their machines. Also, it is believed that the adoption of open-end architecture using Open-source software will further enhance future improvements to the system.

Successful implementation of this system will allow firefighters and planners to make better-informed decisions about the possible wildfire behaviour. A real-world application would save lives, properties and conserve firefighting resources properly.

1.7. Study Area

I have selected the state of Wyoming as the study area for this project. This research was planned to be extended to the whole United States for full deployment, but because of lack of computational resources, it has been left out currently. Wyoming lies in an active wildfire region and still does not have a proper wildfire risk assessment portal. So, users cannot get updated information about the current fires and whether those fires are going to affect them and their property. Also, the information about the historical fire occurrence is available to the researcher, thus allowing for proper validation of the modelling results. In addition to that, I was able to collaborate with Wyoming State Forestry Division who have a

better real field expertise in fire management and mitigation and fire datasets that I had used for this research. One the system has been properly validated in the state of Wyoming; the model would be deployed for the whole American continent. Although it has been left out currently, the similarity in the fuel models and landscape makes the prototype easily scalable to a larger area. The map of the study area is presented below:



Map 1- 1 Study Area

1.8. Research Objectives

The primary and secondary objectives of this research are presented below:

Primary Objective:

- To develop predictive WebGIS based wildfire model for the State of Wyoming

For achieving primary objective following secondary objectives are outlined:

- To develop wildfire behaviour model
- To develop wildfire spread model
- Verification and comparison of results with standardised model results

- Visualisation of the modelling and simulation results

1.9. Research steps and methods

For completing the WebGIS based wildfire modelling research following is an overview of the steps that will be used:

- Research current state of fire modelling
- Research the coupling of GIS and WebGIS with wildfire modelling
- Research and study wildfire behaviour
- Research the approaches to wildfire modelling for spread modelling and behaviour modelling
- Develop wildfire simulation model using Huygen's and Rothermel's principle
- Develop WebGIS based wildfire modelling system
- Implement and perform comparison with standard modelling results
- Present the results and summarise the findings

The main components for performing this research are:

1.9.1. Research and quantification of wildfire behaviour

Since the primary objective of this study is to implement WebGIS for wildfire modelling, a proper understanding of wildfire behaviour is required. The wildfire behaviour will be reviewed and modelled using widely adopted equations and theories. The variables such as wildfire intensity and rate of spread will be quantified by utilising several spatial and non-spatial inputs such as weather, fuel types, topography among others.

1.9.2. Development of wildfire spread model using wildfire behaviour inputs

The study and quantification of wildfire behaviour and quantification of its variables are followed by the development of wildfire spread model using wildfire behaviour variables as inputs. Huygen's principle as discussed in detail in Chapter two and three will be used to predict how the wildfire is going to propagate in terrain with respect to time. Several factors

such as topography, the wind, weather affect the fire spread behaviour. The amount of destruction caused by fire also depends on the fire intensity and the area covered by the fire. Despite the presence of both vector and raster based approaches in spread modelling, vector based approach will be used because of its high precision and closer reflection of the real world phenomenon as compared to the raster-based method. The detailed discussion on the choice of modelling approach is presented in chapter 2. Huygen's principle of wave propagation will be used to predict how and where the fire is going to spread at a specified time from the initial ignition.

1.9.3. Presentation and comparison of results

The completed model will be presented in the form of web-based GUI for the users. The results obtained from the model will be compared to the results obtained under similar conditions from standard wildfire modelling system FARSITE. The comparative analysis and discussion of both results will be presented to validate the results achieved from the model.

1.10. Expected Results

The primary result of this study is a WebGIS based portal for users to perform wildfire spread simulation and visualise the results and interact with them such as time series controller among others.

1.11. Assumptions and Limitations

Fire spread model has been developed to account only for surface fires even though other types of fire spread such as Crown fire also happens in real world. Also, the phenomenon of spotting in wildfire spread has not been dealt with in this research. Because of this two phenomenon, the fire spread predicted by the model will not reflect the real-world scenario. The modelling results have been compared with simulation results obtained from

FARSITE under similar conditions. It is assumed that FARSITE produces fairly accurate results under homogenous fuel and weather conditions.

Several datasets were acquired for the purpose of this research such as wind speed, wind direction, temperature and humidity. It is assumed that those datasets are accurate and reflects true weather conditions of the study area. The fuel moisture data acquired using MODIS Terra satellite image is considered to be the closest approximation to the real world fuel moisture. The fuel moisture condition is expected to remain same for two days because of the time lag of two days between two subsequent MODIS satellite image captures.

The weather datasets are updated every 3 hours and have a coarser resolution of around 5km which could affect wildfire spread prediction. This lower resolution data and time lag between two following datasets have prevented me from getting even better simulation results. Other datasets such as fuel models acquired from LANDFIRE were developed for 2014 and might not reflect exact fuel conditions.

Efforts have been made to gather updated datasets as far as possible, but the limited availability of the relevant datasets and massive costs associated with the purchase of higher resolution datasets have prevented the fullest implementation of the developed system. Also, the highly dynamic nature of wildfire makes the computation cumbersome. Thus it is assumed that other inputs besides weather will remain same for one simulation run.

1.12. Contribution to Knowledge

Despite the presence of a multitude of fire modelling system, they often lack a closer integration of wildfire behaviour and wildfire spread models. Also, such models and tools are only available as desktop based solutions requiring complex installation on the user's side. This along with the cost associated with acquiring GIS software and the steep learning curve has been hindering the widespread use of wildfire modelling by the general public.

Development of WebGIS based wildfire modelling system will explore a new area of research by implementing complex wildfire models on WebGIS based infrastructure. Also,

it makes planning and response operation in wildfire scenario more accessible and efficient. Some of the application could be evacuation route planning, early alert systems based on user's geolocation and the establishment of firebreaks among others. The utilisation of real-time and forecasted data allows a closer perception of wildfire spread in the near future. The users would be able to access all the sophisticated functionality in a variety of devices without the need to install any plugins, applets or tools on their machine.

1.13. Outline of Thesis

The overall outline and short description of the thesis chapters are presented below:

Chapter 1: Introduction

This chapter provides an overview of the research. It starts with the brief background and introduction to wildfire modelling and simulation then progressing towards how we can integrate WebGIS into wildfire modelling. The rationale for doing this research along with the area of study and limitations of the research is discussed in this chapter.

Chapter 2: Principles of Wildfire Modelling

This chapter discusses the concepts and principles of wildfire modelling in detail. Various types of wildfire spread models as well as variables affecting wildfire behavior are discussed in detail. The fire behavior variables are quantified and its role in performing mathematical wildfire modelling is illustrated in detail. The existing fuel and fire behavior models and software are discussed along with their pros and cons. This chapter lays theoretical base for performing wildfire modelling later in chapter 3.

Chapter 3: Development of Wildfire Model

Using the basis of wildfire modelling principles discussed in Chapter 2, this chapter considers the guidelines for developing wildfire model and develops one. The datasets required along with their detailed description and acquisition mechanism is given in this Chapter. The calculation of fire variables and the effect of several factors such as weather, topography and fuel moisture is presented in detail. This chapter explains and justifies the use of different equations for calculating fire behavior variables such as fire intensity and

rate of spread. The role of fuel model and moisture content in determining fire spread is quantified in this chapter.

Chapter 4: Design and Development of WebGIS based Wildfire Model

This Chapter utilizes the mathematical equations chosen and developed in Chapter 3 to design and develop a WebGIS based wildfire system. The system architecture and the flow of system is illustrated in the form of flow diagrams and charts. A comprehensive discussion has been presented on the data acquisition and management mechanism for dynamic modelling. A dynamic wildfire model integrating several client and server based technologies is developed in this chapter. The scripts and source code developed as a result of this process is given in Appendix-C.

Chapter 5: Evaluation and Discussion

This chapter evaluates the results obtained after deploying the developed system. The results obtained from the modelling is compared with widely used fire modelling system FARSITE. The findings from the comparison along with the possible reasons for difference in results is discussed on the latter half of the chapter.

Chapter 6: Conclusion and Recommendation

This chapter concludes the findings from the research and compares it with the objectives set forth in Chapter 1. The lessons learnt after completing this research along with the recommendations for future improvements is presented in this chapter.

Chapter 2: Principles of Wildfire Modelling

2.1. Introduction

Before undertaking any fire modelling and prediction activities it is of utmost importance to understand the basic behavior of wildfires. The study of wildfire behavior will allow us to properly understand the dynamics of wildfire and the different factors that needs to be considered for proper fire modelling. The efficacy of the fire model depends heavily on the proper and comprehensive understanding of wildfire and its behavior. It would also facilitate the user to understand the domain of wildfire. One could put several questions about the source of wildfire and the factors governing its rate of spread which is very important while designing and developing fire spread models.

2.2. Wildfire Behavior

2.2.1. Definition

Wildfire is generally associated with all the kinds of fire events occurring in the nature. However, several authors have standardized what a wildfire precisely refers to. National Wildfire Coordinating Group(NWCG) in its Glossary of Wildland Fire Terminology has defined wildfire as *"An unplanned, unwanted wildland fire including unauthorized human-caused fires, escaped wildland fire use events, escaped prescribed fire projects, and all other wildland fires where the objective is to put the fire out".* Another most widely acclaimed definition in wildfire domain given by Global Fire Monitoring Center (GFMC) considers wildfire from two perspectives. Their first point of view specifies any uncontrolled and unplanned fire event requiring suppression activity to be undertaken can be classified as wildfire. In addition to this, any freely blazing fire in wildland which was uninfluenced by different suppression activities is termed as wildfire (GFMC, 2016).

2.2.2. Origins of wildfire

Wildfire is a very dynamic phenomenon and several factors are responsible for starting a wildfire. The most commonly associated cause for wildfire is the human intervention (Martinez, Vega-Garcia, & Chuvieco, 2009). National Park Service, the government agency in United States responsible for managing and protecting national parks has quoted that around ninety percent of wildfires occurring in the United States are ignited by human interference (NPS, 2016). Usually unattended campfire, intentional lighting and carelessly discarded cigarettes are the most commonly observed human activities resulting in wildfire. Another notable source of ignition for wildfire is lightning and lava (Chuvieco, 2003). In most cases, long-lasting hot lightning bolts causes wildfires if they come in contact with suitable ignition conditions such as dry fuel and less moisture (NPS, 2016).

2.2.3. Factors affecting spread of wildfires

Several factors affect how the fire is going to propagate in a terrain. The most notable factors the influence spread of wildfires are given below:

2.2.3.1. Vegetation:

The type and nature of vegetation present on the surface of any terrain governs the burn intensity as well as the rate of spread of fire. As for instance, small dry twigs burn faster as compared to larger logs. Also, if there are ladder fuels and the crown height is less, there are higher chances of torching crown fire. Thus, the probability of crown fire and fire intensity is influenced by both the vertical and horizontal arrangement of fuel (Omi, & Martinson, 2002). Vegetation is responsible for providing the fuels necessary for burning a fire. However not all the vegetation types respond equally during a fire event. So, a proper study of vegetation and is properties is required to quantify whether it provides adequate fuels to burn a fire is required (Skidmore, 2002). Several factors such as moisture content of

vegetation, their type and vegetation health needs to be considered before performing any analysis.

The moisture present in the vegetation determines how fast or slow the fire burns. Vegetation with less moisture content could ignite fire easily as compared to a damp/moist one. However, although the burnability of moist vegetation is less, it often has a smothering effect that allows the fire to be sustained for a longer time (Trollope, Trollope, & Hartnett, 2002). The presence of higher moisture content in vegetation deters pre-heating of fuels to ignition temperature by absorbing heat produced during fire event (Burgan, & Rothermel, 1984). This decreases the chances of ignition as a considerably higher amount of heat is required to evaporate the moisture before fire spread.

The type of vegetation determines the burn susceptibility during a fire event. The amount and type of fuels present vary across different vegetation. The kind of plant is often the primary factor allowing us to classify wildfires as bushfire, forest fire, desert fire and so on. Also, the density of the vegetation present determines whether there are enough quantities of fuel present to sustain the ignition. Also, the presence of homogenous vegetation type results in uniform wildfire spread.

The health of vegetation determines the amount of moisture present in the fuels. Live herbaceous vegetation is less susceptible to burn as compared to dead fuels. The presence of green fresh foliage which is in its early phases of growing cycles have highest moisture content ranging around 300%. As opposed to this, the vegetation which is old and completely cured is treated as dead fuel has moisture content only around 30%. This difference in moisture content determines their response during a fire event. Also, the land use pattern is affecting fire spread. As for instance, if the land is unattended for a longer period of time it often develops dry fuels making them susceptible to fire when ignited. As opposed to this, the currently farmed lands have greater resistance to fires. In addition to these, the chemical content of the fuels also affect the fire and its combustion behavior. The amount of fuel loading present directly affects surface fires as higher fuel loading facilitates wildfire more than sparse fuel loading.

2.2.3.2. Wind:

Wind velocity (comprising of both speed and direction) is another crucial fire model input that determines the direction and fire propagation along with its rate of spread (ROS) (Albini, 1976). Often midflame wind speed is taken as a deciding factor for calculating fire intensity and ROS. In the simplest terms, the stronger wind results in faster fire spreads. Also, the speed of wind determines the shape and eccentricity of the fire spread. The higher speeds results in more pointed shaped ellipse whereas lower speed wind results in more of a circular spread (Finney, 2003).

2.2.3.3. Topography:

Topography has been regarded as the most stable variable in fire behavior triangle because of its relative static nature. Another consideration in fire modelling are topographical factors such as aspect, slope and elevation. Aspect defined as orientation towards the sun dictates the amount of energy received from the sun. South slopes receive sunlight for a longer duration and are warmer allowing drier fuels during the fire season (Skidmore, 2002). In contrast to the south, North slopes have higher vegetation and higher moisture content which does not facilitate sooner fuel drying. However, once they start burning, they might burn even severely because of their higher fuel content.

Slope directly influences fire spread as fires spread faster uphill than downhill. It is primarily caused by increased pre-heating of the fuels upslope due to convection. This preheating eventually reduces the temperature requirement for ignition thus igniting fire in shorter time duration (Mansor et al., 2004). Presence of slope changes the angle of impact/spread of fire thus affecting larger area in lesser time.

During the fire progression, heat is developed at the front which more effectively pre-heats and dries upslope fuels, making for more rapid combustion (Carmo, Moreira, Casimiro, & Vaz, 2011).

Elevation also affects seasonal drying of fuels with lower elevations drying out fuels earlier in the year (Westerling, & Bryant, 2008). In addition to this, higher elevation locations have more tendency of being struck by lightning strikes and subsequent ignitions. Also, nature of topography such as the presence of narrow canyons could easily accelerate fire whereas the presence of barrier and natural breaks could stop the fire.

2.2.3.4. Weather:

Weather data also known as meteorological datasets comprises of relative humidity, temperature, and precipitation. These values allow us to calculate intensities of fires. Less humidity coupled with higher temperature and less or nil precipitation all fosters fire propagation. These variables also allow fire models to add negative cost to fire spread wherever there is noticeable precipitation that is most likely extinguish the fire. Under a given wind condition, the rate of spread of fire can be heavily altered due to fluctuations in temperature and relative humidity (McCaw, 2009). Also because of lesser moisture, fires spread more rapidly and ignite with higher intensity during warmer part of day such as noon. On the other hand, the presence of higher relative humidity and lower temperatures during the evenings often deters the fire spread thus reducing its rate of spread and even extinguishing the fire column. Since the weather variables such as temperature, relative humidity are highly dynamic, care should be taken while using them in modelling wildfires.

Relative humidity is intrinsically linked with fuel moisture content as higher relative humidity results in increased absorption of moisture by vegetation thus making them less likely to burn. On the other hand, higher temperatures, no precipitation and lower relative humidity are the most favourable weather condition for a wildfire (Skidmore, 2002).

2.2.3.5. Other Factors:

Other factors such as water bodies, roads and airstrips also affect wildfire behavior. These features could act as fire breaks or limit the fire intensity considerably. The presence

of settlement also changes the wildfire behavior because of different types of construction activities undertaken by humans. In most of the prescribed burns, fire lines in the form of roads are constructed to prevent fire from escaping beyond the desired area. Presence of water bodies such as rivers, dams and reservoirs usually extinguish fires.

2.2.3.6. Combined effects:

These factors and variables act in combination. At times, it might be seen that these variables are contradicting with each other. As for instance, if wind is blowing downhill, it could change the uphill-accelerating effect of slopes. If this situation is coupled with more dry fuels downhill, the effect of slope would be negated to some extent.

For a proper wildfire simulation, all the possible interactions between these and others secondary factors needs to be accurately considered. As wildfire phenomenon ranges from micro-scale to macro-scale, even the small scale variations in natural environment have a crucial impact on determining how the fire is going to behave. These interactions often could not be properly modelled because of lack of adequate data, knowledge about their possible impact and the sheer complexity required to incorporate them into the model properly. Despite several attempts to model these interactions precisely, those models have so far remained on the laboratory simulations. The different researchers that have been undertaken in wildfire simulation domain using different approaches and models are presented in the section below.

2.3. Wildfire spread models:

Wildfire spread models could generally be classified into four categories namely Crown fire models, surface fire models, ground fire models and spotting fire models (Pastor, Zarate, Planas, & Arnaldos, 2003). This classification has been made using the underlying physical system as the basis. All of these wildfire spread models are discussed briefly in following subsections:

2.3.1. Crown fire models:

A fire burning through the treetops or top layer of foliage is called crown or canopy fire (NPS, 2016). These type of fire usually propagates with great speeds and often jumps from tree crown to crown. This results in the crown fire being ahead of ground fire at most of the times. The primary fuel for igniting crown fires are tall trees as well as ladder fuels. Ladder fuels help the fire to propagate from the surface to the tree tops (Pastor, Zarate, Planas, & Arnaldos, 2003). Once these fires are started, they are tough to put out as they tend to behave independent of surface fire, so there have been practices of preventing crown fires by removing and thinning ladder fuels (NPS, 2016). Crown fire models account for the foliage content in the form of crown cover and crown base height along with crown density and its moisture to predict the behaviour of crown fires.

2.3.2. Surface fire models:

A fire that ignites and propagates on the surface of the ground with the help of fallen branches, leaf litter along with other kinds of fuels present on the earth surface is termed as surface fires (NPS, 2016). Fuels are ranging from small trees to bushes to herbaceous vegetation which is usually less than 2 meters in height responsible for starting and propagating surface fires (Pastor, Zarate, Planas, & Arnaldos, 2003). They could burn with either low or high intensity. Upon contact with suitable ladder fuels, they could also torch crown fires. Surface fire models usually are governed by the wind factor and the moisture content and burnability of the fuels present.

2.3.3. Ground fire models

A fire burning under the ground with the help of organic materials such as peat, coal tree roots is called ground fire (Pastor, Zarate, Planas, & Arnaldos, 2003). These fires are usually sustained for a long time due to smouldering (NPS, 2016). Since these fires burn beneath the surface, they might not appear on the surface. However, they do have the

potential to start a surface fire if they come across suitable fuel and oxidation availability. Ground fires are less catastrophic as compared to surface and crown fires. Ground fire models are usually implemented for modelling bushfires where there are lesser chances of starting a surface fire. A schematic diagram is given below to illustrate ground, surface and crown fires:

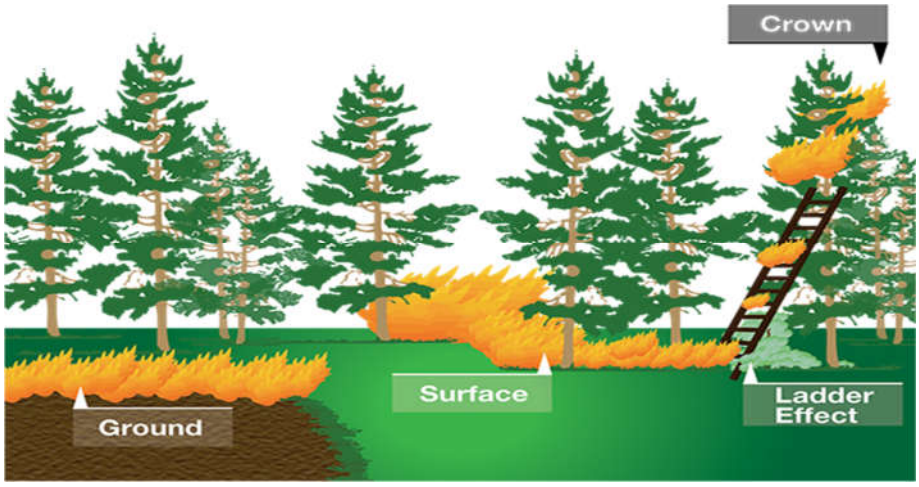


Figure 2- 1 Ground, Surface and Crown fires along with ladder effect
Source: Sarah Lynch-Walker http://advancedmastergardener.org/old/FireSafety/section4.html.

2.3.4. Spotting wildfire models:

Martin and Hillen (2016) defines spotting in wildfire as "*non-local creation of new fires, due to downwind ignition of brands launched from a primary fire*". These new fires start their own fire behaviour once they are started. Since the behaviour shown by the new fires are often unpredictable, spotting is regarded as the most severe problems in wildfire management (Martin, & Hillen, 2016). Several factors affect the impact of spotting on wildfire such as the source of firebrands, the distance they travel before coming in contact with fuel surface and finally their probability of starting a new fire once they land (Rothermel, 1983). Spotting is highly responsible for significant damage in wildland urban interface where the firebrands being carried by wind or fire twirl comes in contact with highly combustible roofs and other features. They are highly unpredictable, and there are high chances that they might remain unnoticed and start an even catastrophic fire is very high. Another factor such

as plume height, transition zone length and flame height along with wind speed affects the spotting range and intensity in wildfire event.

Figure 2- 2 Fire Spotting phenomenon

Most of the spotting models consider only the maximum distance and physical processes for predicting spotting. However mathematical models are derived now for predicting the impact of spotting using detailed physical processes (Martin, & Hillen, 2016).

2.4. Fire Behavior

The term fire behaviour refers to the wildfire displays such as how fast it is moving, how severe is its impact and so on. However not all the fire behaviour characteristics are quantifiable in the real world (WFM, 2012). Some of the significant fire behaviour that is quantitative are the rate of spread (ROS), fireline intensity (FI), flame size and heat released per unit area (HPA). A brief introduction along with their role in wildfire modelling is presented in the subsections below:

2.4.1. The rate of Spread (ROS):

The rate of spread (ROS) has been defined as "*the linear rate of advance of a flaming front into unburned fuel in the direction perpendicular to the fire front*" (WFM, 2012). Rothermel (1983) in simplest terms defines ROS as the horizontal distance covered by the

flame per unit of time. One of the most commonly practised methods of determining ROS measures the time required by a flame front to travel from one known point to the next known one.

ROS has been regarded as one of the most important fire behaviour characteristics because it dictates how much larger a fire could become and what is its probability of affecting areas of concern. ROS also directly impacts fire line intensity which in turn determines the catastrophic effect of fire.

The Wind is the role player in determining the rate of spread. ROS is greatest in the wind direction and is least in the opposite direction (Kennard, & Fowler, 2008). Other factors such as fuel moisture, slope and weather conditions also affect the rate of spread of wildfire (Rothermel, 1983).

2.4.2. Fireline Intensity (FLI):

The term fireline intensity (also known as Byram's fire intensity) refers to "*the rate of heat released per unit length of the fire front*" (Byram, 1959). Van Wagner (1977) argues FLI is the fundamental and most informative fire characteristic and is very easy to measure. The most widely used equation for calculating fireline intensity was given by Byram (1959) as:

FLI= H * W * ROS

Where FLI: Fireline intensity

H: Heat content

W: Fuel load consumed at flaming point

ROS: Rate of spread

In simplest terms, FLI = HPA * ROS

Where HPA: Heat released per unit area

The total fireline intensity is affected by the fuel load, total area burnt and the heat produced by the fire. Rothermel (1983) states FLI is significantly determined by the fuel moisture, relative humidity as well as wind speed.

Fireline intensity is often confused with fire severity. However, fire severity is used to describe the immediate effects of fire on soils, vegetation, litter, etc. whereas fireline intensity is the measure of physical change brought about by burning. Also, severity is not a quantitative measure and requires qualitative ranking of fires from low to high severity. In addition to that, fire severity directly depends on the duration of burn as fires burning for a longer period are more severe than temporary ones.

2.4.3. Flame Size:

Flame size refers to the physical dimensions of the fire flames. Flame height and flame length are the measures used to quantify the size of the flame. Flame height relates to the distance centre of flame depth to the tip of the flame. Whereas flame height is the vertical distance of the flame tip from the ground. Flame length could be either vertical or inclined in contrary to a flame height which is always vertical (Kennard & Fowler, 2008). An illustration of flame length and flame height is given below:

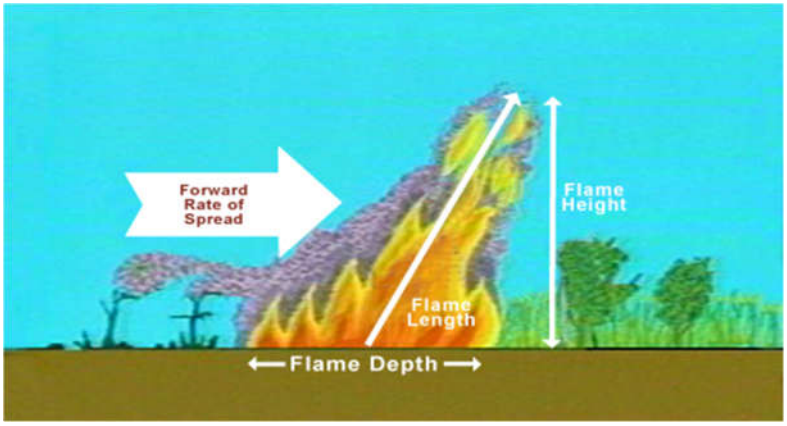


Figure 2- 3 Flame size in terms of flame height, length and flame depth
Source: Utah State University(Wildland Fire Management and Planning)

Hungerford (1991) regards flame length as a visual measure of fire intensity. High-intensity fires usually have larger flame lengths and have the capability of starting crown fires easily. Also, the wind is another crucial factor determining flame size. Trollope, Trollope, & Hartnett (2002) mentions stronger winds tilts the flame front towards the fuel thus reducing the flame height but increasing the flame length. Such condition causes the

fire to reach farther but lesser chances of crown fires. Flame length is observable by the human eye whereas fireline intensity is not. Proper calculation of flame lengths is problematic due to exponential increase and decrease in length in very short span of time (WFM, 2012).

2.4.4. Heat Per Unit Area (HPA):

WFM (2012) defines heat per unit area (HPA) as the "*amount of heat released per unit area during the short period of continuous flaming*". In mathematical terms, HPA can be calculated by as follows:

HPA = H * W

Where, H: fuel particle heat content (H) and

W: fuel load consumed during the passage of fire flaming front.

It has been proved that variations in H among different fuel particles are negligible thus HPA is assumed to be affected primarily by fuel load consumption(W). In other words, choice of fuel model governs HPA for surface fires and live and dead fuel moisture content (WFM, 2012)

2.5. Types of mathematical wildfire models

Attempts have been made to make comprehensive fire models and simulation models since the beginning of the 19th century. Although clearly in practice for environmental modelling, fire modelling, does not have a clear distinction between mathematical and computational models and can be broadly classified into stochastic models, deterministic and probabilistic models (Sibolla, 2009).

2.5.1. Stochastic models:

Stochastic models, also known as empirical models, rely on fire experiments performed either in laboratory and outdoor field conditions to model fire behaviour. They use statistical equations in the form of regression analysis which in turn predict various

observations of fire such as fire intensity and the rate of spread. (Glasa & Halada, 2008). Trollope, Trollope & Hartnett (2002) did a study in the calculation of fire intensity as a function of environmental factors such as fuel and meteorological conditions (Trollope, Trollope & Hartnett, 2002). The most notable stochastic model in fire simulation was developed by Rothermel (1972) that allows the users to calculate the rate of spread of fire. Rothermel's equation, after extensive testing has been found to be "robust and stable" (Ntaimo, Hu, & Sun, 2008). However, some researchers point to the difficulty of implementation of such equations in real world scenario (Higgins, Bond, Trollope, & Williams, 2008).

In Australia, the predominant operational stochastic fire spread prediction systems have been the McArthur Grassland (McArthur, 1966) and the Forest Fire Behavior Tables for Western Australia 3 (commonly called the Red Book) (Sneeuwjagt & Peet, 1985). More recently, the Commonwealth Scientific and Industrial Research Organization (CSIRO) Grassland Fire Spread Meter (GFSM) based on the empirical modeling of Cheney, Gould, & Catchpole (1998) has replaced the McArthur Grassland Fire Danger Rating System(FDRS) as the preferred tool for predicting fire behavior in grasslands (Cheney & Sullivan, 2008).

2.5.2. Deterministic models:

Deterministic models could further be classified as physical and semi-empirical models (Pastor, Zarate, Planas, & Arnaldos, 2003). Physical models are based on theory (both physical and chemical process of the phenomenon) and use differential equations with numerical solutions to describe the spread of fire. They are complex in nature and usually applied to small areas and laboratory scales because of its limitation arising with the coarser fuel and meteorological datasets (Johnston, Milne, & Kelso, 2006). These models are analytic nature and hold the ability to represent wildland fuels correctly using detailed fuel descriptions. Although they require much computation skill, they have been proven to

be more accurate and useful than empirical models (Albright & Meisner, 1999; Favier, 2004; Glasa & Halada, 2008; Johnston, Milne, & Kelso, 2006).

Most notable implementations of physical models in wildfire modelling are Wildland Fire Dynamics Simulator and Firestar (McGrattan, Hostikka, & Floyd, 2010). Complex physical models even allow fire models to feedback on other components such as the atmosphere. For example, National Center for Atmospheric Research developed a model called Coupled Atmosphere-Wildland Fire-Environment (CAWFE) model with Weather Research and Forecasting (WRF) model and University of Colorado Denver's combination of WRF with fire spread model.

Semi-empirical models also known as physical-statistical models use the laws of physics to model the transfer of heat energy which in turn are applied to a bulk of experimental fires to derive statistical correlation (Glasa & Halada, 2008). These models are well-suited for faster than real-time computation with low requirements of data and are much easier to implement than physical models (Glasa & Halada, 2008). However, these models are highly dependent on condition obtained by the source data and often requires significant approximations which might not be necessarily true (Sullivan, 2009a).

In the United States, the quasi-empirical model of Rothermel (1972) forms the basis of the National Fire Danger Rating System (Burgan, Klaver, & Klaver, 1998) and the fire behaviour prediction tool Behave (Andrews, 1986). The Rothermel (1972) model and its associated systems have been introduced to several countries, particularly Mediterranean Europe. In Australia, semi-empirical model FDRS have been used in fire simulation (McArthur, 1966). After about 60 years of research, a quasi-empirical Fire Behaviour Prediction(FBP) system is playing a pivotal role in forest fire prediction and danger rating system of Canada (Taylor & Alexander, 2006).

2.5.3. Probabilistic models:

Probabilistic models are based on calculation and prediction of the probability of fire spreading from one location to another. These models do not calculate fire behaviour

characteristics and variables but rather predict fire in an area by weighing values of parameters stored in contingency tables. As with stochastic models, these are only valid for the areas they are developed for (Albright & Meisner, 1999).

2.6. Fire Spread Simulations:

There have been several attempts at varying level of complexity and truthfulness in predicting the spread of fires in nature. The end goal of all those attempts is to produce a fire spread model that could be used easily and could provide timely updated information about the spread of fire to the concerned users (Sullivan, 2009b). Beer (1990) argues that the immense popularity and accessibility to personal computing and geographic information system has resulted in more and more methods of predicting fire spread. The initial models were mostly one-dimensional in nature thus failing to simulate fire perimeter accurately. Later two-dimensional models were developed to simulate fire propagation more accurately (Sullivan, 2009a). Two fundamental processes exist in any kind of fire spread simulations namely: proper representation of fire and propagation of fire perimeter in a way that properly represents real fire behaviour (Sullivan, 2009b).

Two types of approaches exist in spread modelling namely vector and grid-based models. These models differ in the way they represent the modelling surface and the process they use to spread the fire (Yassemi, Dragicevic, & Schmidt, 2008).

Sullivan (2009b) states that vector-based model considers the fire perimeter as a closed curve comprising of connected points. The fire is then propagated in the form of polygon growing with respect to time (Yassemi, Dragicevic, & Schmidt, 2008). Ellipse is the most widely used geometrical shape to represent fire growth. Majority of vector based approaches have adopted Huygen's wavelet principle to simulate fire propagation (Yassemi, Dragicevic, & Schmidt, 2008).

Raster/grid-based models use adjacent independent cells to represent fire. These types of approaches divide the landscape into either regular or irregular grid cells often sharing some characteristics. Then the fire is propagated using transition rules or probability

rules. Cell automation and Band percolation methods are the most widely used raster based models (Pastor, Zarate, Planas, & Arnaldos, 2003).

Despite both of the approaches having their set of pros and cons, when it becomes necessary to simulate fire propagation in a heterogeneous environment, vector based approaches are more suitable. As grid based methods are the generalisation of the landscape that is bounded by their cell size, they could not accurately represent complex conditions present there (Yassemi, Dragicevic, & Schmidt, 2008). To adequately account for a highly dynamic event such as wildfire that is influenced by changing meteorological conditions and fuel conditions, it is highly imperative to adopt a vector-based model. Despite its ability to simulate fire more accurately, it often demands complex computations. In this respect, raster based approaches produce results in lesser amount of time. There have been practices of performing fire modelling in raster and converting back to vector format later, but they have in most cases resulted in the loss of accuracy (Perry, 1998). A brief introduction of the practices being done in vector based approach and grid based methods is presented in the subsections below:

2.6.1. Vector based approach using Huygen's Wavelet Principle:

Huygen's principle of wave propagation is the most widely used elliptical wave propagation method used for approximating fire front. It was initially developed for modelling propagation of light waves which was adapted to fire propagation first by Anderson et al in 1982. The basic underlying theory considers each point lying on the wave front as the new source for the wavelets to move forward. In this approach the fire spread from new positions depends on their local conditions and the spread model being adopted. As the time step moves forward, one wave could give rise to more waves increasing the chances of fire spread (Perry, 1998).

Ellipse is used to define the shape attained by new fires. Despite several alternatives such as a double ellipse, tear drops and lemniscate amongst others, in most of the scenarios ellipse has been found to best represent wildfire propagation (Anderson, Catchpole, de

Mestre, & Parkes, 1982; French, Anderson, Catchpole, 1990). Several parameters such as rate of spread, current wind speed and direction are crucial inputs for determining the shape of the ellipse. The major axis of the ellipse is aligned to the direction of the wind. As the wind speed increases, the ellipses attains longer major axis. Another parameter determining the shape of the ellipse is its length to breadth ratio (L:B) and distance of the rear focal point from the centre. The distance of rear focal point from the centre is used to quantify how much backfire is going to happen. A basic illustration of this concept is given in the figure below:

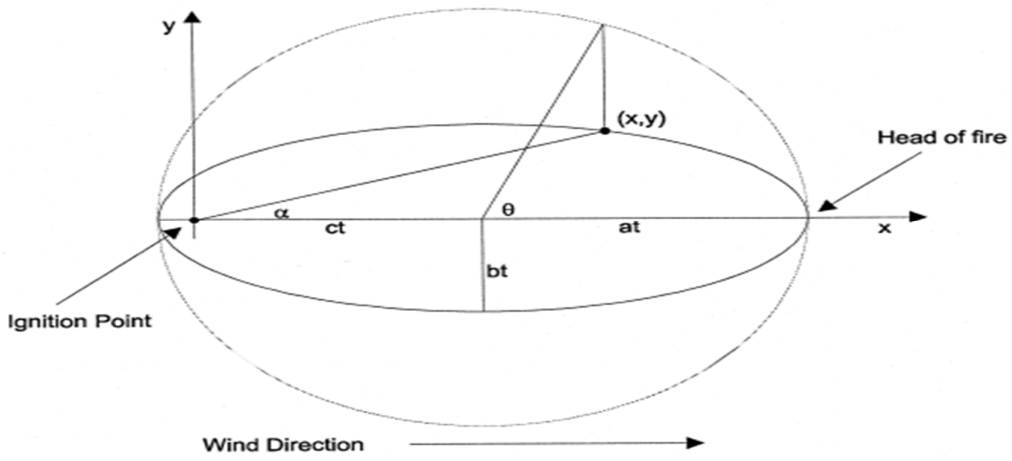


Figure 2- 4 Elliptical representation of wildfire spread
Source: (Perry, 1998)

Where, at and bt represent the major and minor axis of the ellipse, Θ represents angle subtended by the circle enclosing the ellipse, α represents the angle subtended by real focal point(fire ignition point) on a point on the ellipse, ct represents the distance of fire ignition point from centre of the ellipse.

Ellipse similar to this will then start a new fire from its fire front. Several new ellipses will start, the aggregation of which gives us the new fire front. A basic explanation using a single set of ellipses given initially by Anderson, Catchpole, de Mestre, & Parkes(1982) by adopting Huygen's principle for steady wind is presented below:

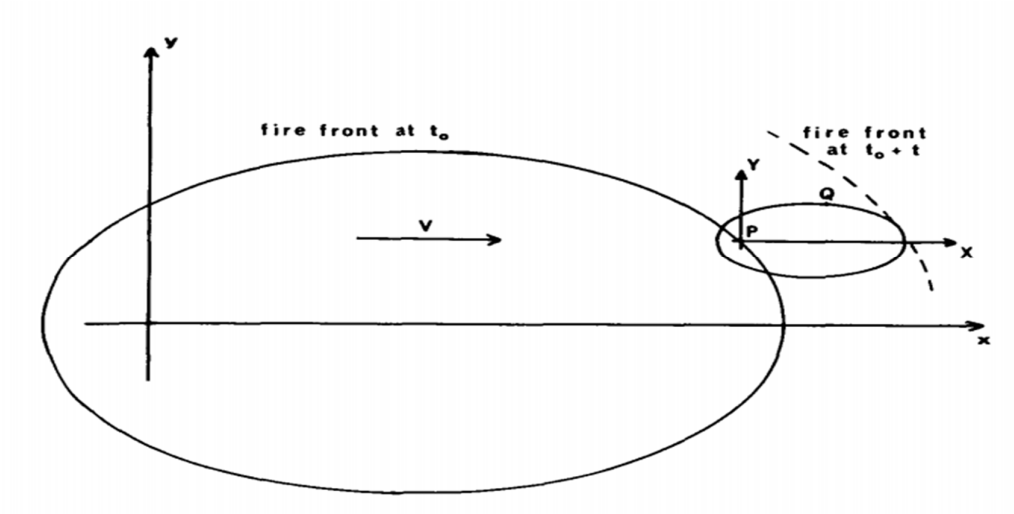


Figure 2- 5 Huygen's principle for steady wind V adapted for fire spread showing elliptical fire spread and creation of new fire front at time $t_0 + t$
Source: (Anderson, Catchpole, de Mestre, & Parkes, 1982)

As the time progresses and the complexity of the landscape increases more than one ellipse are created and collectively account for the propagation of new fire front as given by Rios, Jahn, & Rein (2014) below:

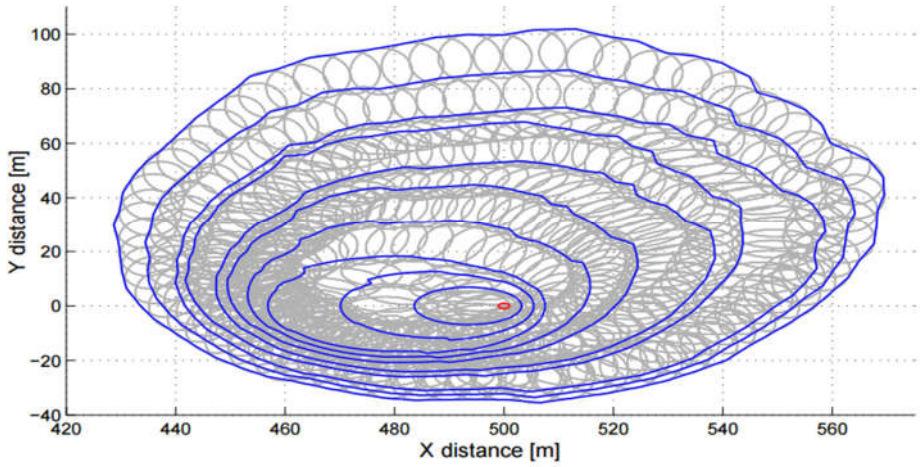


Figure 2- 6 Huygen's elliptical expansion in fire spread (grey ellipses) starting from red dot (ignition point)
Source: (Rios, Jahn, & Rein, 2014)

Where The fuel depth varies from (0.6±0.3m), and wind ranges from (5±2 m/s) with a time step of 1 min.

Some of the important application of Huygen's wavelet principle in wildfire simulations are FARSITE, SinoFire, Canadian Fire Behavior Prediction (FBP) system, Prometheus some of which are discussed in a later section in this chapter.

2.6.2. Raster Based simulation:

In raster based simulation, raster grid of cells are used to represent fire. The cells in the grid could either be unburnt, burning or burnt (Sullivan, 2009b). As discussed above, this method has a tradeoff between resolution and data volume but does provide faster computation speeds. Two types of approaches are most widely used under raster based approach namely Band percolation and cell automata models both of which are discussed in subsections below:

2.6.2.1. Cell Automata:

Cell Automata (CA) is the most widely used raster based fire propagation model. It is also one of the simplest fire models developed in GIS is cellular automation model which depicts the growth of fire through the change in state of the cell with respect to time. Under this approach, the fire is propagated using a simple set of rules which dictates the interactions between a cell's neighbouring cells in a lattice (Sullivan, 2009b). The state of a cell depends on the previous state of the cell as well as the state of neighbouring cells. These models have been useful to depict the fire spread phenomenon in all directions because of their ease of implementation in a GIS environment.

CA models were initially developed by Von Neumann in 1966 for representing complex systems to be used in the modelling environment. This approach considers two-dimensional grid of cells to predict the state of the cell. The state of the cell is affected by its eight direct neighbouring cells as shown below:

i-1, j-1	I, j-1	i+1, j-1
i-1, j	i, j	i+1, j
i-1, j+1	i, j+1	i+1, j+1

Figure 2- 7 CA cell neighbourhood

Other sets of rules dictating CA-based fire propagation are:

1) A cell starts burning if any of its eight neighbouring cells are burning
2) A cell has the probability of igniting *p* which depends on its fuel, moisture and landscape parameters
3) A cell will burn for specified time and will stop burning after that

Perry (1998) highlights the ability of these models to implement deterministic physical models as being the most lucrative reason for fire modellers to use this approach. Yassemi, Dragicevic, & Schmidt (2008) modified the default CA model to incorporate a semi-empirical approach to predict fire spread. They have integrated rate of spread (ROS) as well as the direction of spread using the orientation of wind to propagate fire. This angle subtended to the normal axis by the wind direction affects the spread distance and number of cells being affected by the fire. They have thus developed different sets of equations for different quadrant winds shown below:

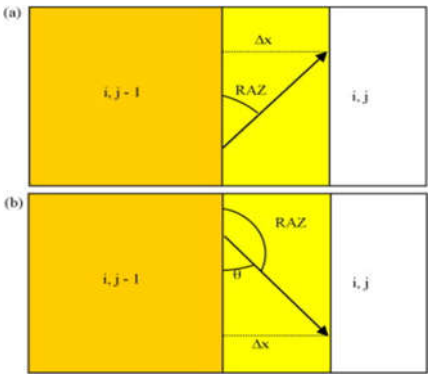


Figure 2- 8 Fire spread and direction of the wind in CA method

Case (a) Fire spreading east RAZ: Direction of travel, where $0 < RAZ \leq 90°$, the eastern component of velocity vector, ROS_East (m/min), ROS_East=ROS*sin(RAZ)

Case (b) where $90° \leq RAZ < 180°$, speed is derived by:

ROS_East=ROS*sin(180−RAZ)

In a similar fashion, they calculated ROS in other directions as well.

One of the significant advantages of the model developed by Yassemi, Dragicevic, & Schmidt (2008) is that it also accounts for the backwards rate of spread. This backward rate of spread computation utilises the outputs from semi-empirical Canadian fire danger rating system (CFDRS). While comparing with vector based Prometheus model, it was observed that the results from this cell automation model were in reasonably good agreement (Yassemi, Dragicevic, & Schmidt, 2008).

2.6.2.2. Bond Percolation:

Percolation in its core technical description refers to *"the process of transport of fluids through porous materials in material science and the transport through a randomly distributed media in mathematics"* (Sullivan, 2009b). If the percolation occurs in the boundaries lattice cells, it is called bond percolation. In a similar fashion as CA method, the landscape is subdivided into cells with different representation such as quantity of fuel moisture, slope, aspect, and type of vegetation as well as the weather information such as wind and humidity (Pastor, Zarate, Planas, & Amaldos).

McCarthy (1997) states bond percolation is concerned with the probability of spread and ignition associated with each cell. In this type of model, historical data also plays a crucial role as the adjustments are made to the derived probability using experimental mathematical models. Some of the notable applications of bond percolation technique in wildfire spread modelling are Ecological Model for Burning the Yellowstone Region also known as EMBYR and the bond percolation method used by Favier (2004).

According to Hargrove et. al. (2000), EMBYR was developed specifically for Yellowstone National Park of Wyoming United States of America for simulating wildfires and

predict future burn pattern. At its heart, its deals with landscape ecology at a larger scale and over a longer period. It applies probability function to determine whether a cell will be ignited and is capable of burning. Combustion of the cell is governed by the predefined critical value which is again a function of spread equation and number of burning neighbouring cells. As with the cell automata model, it calculates the critical value and probability at each time step. It has taken into consideration, the vegetation density to determine burn probability as the sparse presence of vegetation has a lesser likelihood of fire spread as compared to denser one. The Wind also plays a significant role in determining the probability of fire spread. The stronger winds introduce more bias while predicting the spread of fires. Also, the direction is biased is governed by wind. Hargrove et. al. (2000) performed simulations for different weather conditions using this model. This stochastic nature of EMBYR by being localised to Yellowstone scenario makes it difficult to readily apply to other areas as it would require re-calculation of probability values. EMBYR is only applicable in the post-fire situation.

Another notable mention in bond percolation technique is the model developed by Favier (2004). This approach considers two parameters namely combustibility and ignitability of the medium (Favier, 2004). This model has the capability of determining active fire spread on user defined time step basis. Unlike the EMBYR this model is based on physical behaviour which allows this model to be used in other areas as well. This model utilises the energy exchange process occurring between burning and non-burning regions to predict fire spread. It used energy balance equation to classify lattice cells into three states namely safe(S), burning (F) and burnt (B) (Favier, 2004).

Despite their ability to incorporate both physical and stochastic models in fire behaviour, bond percolation techniques are regarded as least favourite methods in wildfire spread simulation. Their major flaw is that they have no mathematical logic between the affecting factors and wildfire behaviour (Pastor, Zarate, Planas, & Arnaldos, 2003).

2.6.3. Other propagation methods:

Despite the widespread application of raster based and vector based models, there exists a considerable number of other propagation models. One of the notable wildfire propagation models is the Coupled fire-atmosphere model developed by Clark et. al. (1998) that links 3-dimensional meteorological conditions with the fire spread model developed by Rothermel.

Fuzzy logic has been applied by Vakalis et. al. (2004) to account for the complex nature of wildfires which were too complex to be handled by conventional quantitative techniques. This model has its unique advantage as it is closely related to human perception. Fuzzy logic is applicable where the exists higher level of uncertainty, or exact quantities of factors affecting fire spread are not known (Vakalis et..al. , 2004).

Another propagation model used in fire domain is Artificial Neural Networks (AAN) which can model complex systems recognising complicated relationships between factors and their collective impact (McCormick, 2002). McCormick (2002) implemented AAN model using grid-based approach to simulate wildfire for Grates Lakes region of United States using historical fire data from Huron National Forest in Michigan, United States of America. He has shown that the results from AAN model have posed it as a better alternative to grid-based fire modelling such as bond percolation and cell automata.

2.7. Existing Fuel and Fire Behaviour modelling systems:

Some of the notable fuels and fire behaviour modelling systems existing today are discussed in this section.

2.7.1. FARSITE

FARSITE is a fire growth and simulation modelling system that has been widely used for simulating the spread of wildfire and fire use for resource benefit across different landscapes used by US Forest Service, National Park Service(NPS), and several other federal land management agencies (Finney, Brittain, & Seli, 2004). It can incorporate several existing models for simulating surface fire, crown fire, spotting, post-frontal

combustion, and fire acceleration into a two-dimensional fire growth patterns. Some notable fire behaviour models in FARSITE are Rothermel's (1972) surface fire spread model, crown fire spread model of Rothermel's (1991), Albini's (1979) spotting model, Van Wagner's (1977) crown fire initiation model and Nelson's (2000) dead fuel moisture. FARSITE uses vector propagation method for delineating fire front.

2.7.2. FlamMap

Although there existed several computerised and manual systems for modelling wildland fire behaviour, they were not implemented at landscape level with interacting components and inputs being mapped using Geographic Information Systems (Finney, 2006). FlamMap is fire mapping and analysis system that calculates fire behaviour for each pixel within a landscape independently (Finney, 2006). It can calculate surface fire spread using Rothermel (1972), crown fire initiation using Van Wagner (1977), the crown fire spread using Rothermel (1991), and dead fuel moisture is calculated using Nelson model (Nelson, 2000). However, FlamMap cannot simulate temporal changes in fire behaviour that is induced by weather and diurnal fluctuations as FARSITE does. FlamMap can create raster maps of several potential fire behaviour such as spread rate, flame length, crown fire activity, mid-flame wind speeds among others which can be viewed either in FlamMap or other GIS or image viewing software packages (Finney, 2006). FlamMap cannot be regarded as a replacement of FARSITE and is not a complete fire spread simulation model since it calculates fire behaviour for a single set of environmental conditions (Finney, Brittain, & Seli, 2004).

2.7.3. BehavePlus

The Behave fuel modelling and fire prediction system were one of the earliest computer systems designed and developed for wildland fire management which updated and expanded over time to its modern iteration BehavePlus. Using a suite of fire behaviour

systems such as FlamMap, FARSITE and FSPro it allows users to quantify fire behaviour in the form of rate of spread, spotting distance, scorch height and tree mortality for quantifying fire effects in addition to fire environments such as fuel moisture and wind adjustment factor (Andrews, 1986). Even though it lacks the spatial display component, it provides graphs, tables and simple diagrams that could be coupled with spatial information for better decision making (Andrews, 2010).

2.7.4. NEXUS

NEXUS utilises the same geospatial fire models as BehavePlus and FARSITE to assess the crown fire hazard. Its unique advantage as compared to others is its ability to calculate crown fire potential index and incorporation of dynamic charts in output. Users could input point or plot to perform fire modelling which seamlessly incorporates surface and crown fire models. The inputs desired from users are the wind data, topographical data such as slope and surface as well as canopy fuel characteristics (Scott, 1999)

2.7.5. FMAPlus

FMAPlus is a set of programs that allows the users to estimate both surface and canopy fuel loading. It can also predict fire behaviour and its effects. FMAPlus is particularly useful in simulating the effects of thinning. In its core, FMAPlus uses Finney (1998) and Reinhardt (2001) wildfire simulation models to derive fire hazard indices and potential fire behaviour (WFM, 2012). It, however, requires tree list and fuel load by classes and component to perform those simulations.

2.7.6. Virtual Fire

One of the important fire simulation product that has been published to date is Virtual Fire, a web-based GIS platform for providing GIS capabilities to firefighting forces. It merges real-time information obtained from several remote weather stations and integrates with

other components such as daily updated satellite imagery to provide up to date fire based information. It uses Windows high-performance computing (HPC) platform for weather forecasting, fire simulation fire prevention and early warning (Kalabokidis et al., 2013). It allows non-GIS users to query the databases and get answers immediately and even locate points of interest in high-resolution satellite images, and can also get information from GPS receivers in real time (Microsoft, 2010). However, users are required to install Microsoft Silverlight plugin in their browsers, and latest Google Chrome browsers have already deprecated these plugins making it impossible to run on modern updated browsers (Google, 2014). Also, spread simulation requires users to submit a request to the administrator with all the required information and wait for an administrator to run the simulation and send back the results (Kalabokidis et al., 2013). So, real-time simulation on the user end is not practically implemented.

2.7.7. WIFIRE

Another significant wildfire simulation system currently being developed is called WIFIRE. It is being developed by University of California at San Diego with the help of National Science Foundation (NSF) grant for developing and deploying an end to end cyberinfrastructure. Its primary purpose is to perform a real-time data-driven simulation to predict wildfire and accurately visualise the wildfire behaviour (Altintas et al., 2015). Once fully developed it will integrate data from heterogeneous sources such as satellite data, and remote sensor data along with datasets from individual tweets to model fire behaviour. It will use resources from San Diego Supercomputer Center to run the data processing and complex simulation. Their end goal is to facilitate the development of wildfire control room so that wildfire monitoring and management along with resource allocation will become centralised (Altintas et al., 2015).

Chapter 3: Development of Wildfire Model

3.1. Introduction:

As discussed in the earlier chapter, there exist two types of wildfire model namely fire spread model and wildfire behaviour model. Fire spread model is concerned with predicting how the fire is going to propagate in a given environmental and topographical conditions. Fire behaviour model, on the other hand, is concerned with describing the characteristics of fire namely flame length, fire intensity. Fire behaviour model often allows us to quantify the destructive potential of the fire.

This chapter is focused on selection and development of wildfire model that incorporates both wildfire behaviour model and wildfire spread computation. Both the conceptual and theoretical design of the model will be discussed in this chapter. The results obtained from this model will be used later to integrate into WebGIS infrastructure to provide these modelling capabilities over the web. The visualisation part will be discussed in next chapter.

3.2. Fire Model development guidelines:

Although there exist a plethora of modelling guidelines for the development of environmental models, there is a dearth of proper guidelines while developing a fire model. The most comprehensive document for providing guidelines for mathematical fire model development was given by Rothermel (1972) who states the following steps are significant milestones:

3.2.1. Data acquisition and evaluation:

Data acquisition is the first phase of model development where the developers determine the required datasets and acquire them. Those acquired datasets need to be evaluated before use to make sure that they satisfy their desired purpose.

3.2.2. Calculation of fire variables:

This is the crucial step in fire modelling as it calculates the variables that define fire behaviour and spread variables such as rate of spread.

3.2.3. Interpretation of fire variables and generation of secondary indices

After the calculation of fire variables, they are used to generate secondary indices such as burn probability and risk. These complex secondary indices and variables need to be interpreted with the help of supporting information and classification.

3.2.4. Visualisation and inference of results

The final step of fire modelling is the display of the calculated results and generation of inference on fire characteristics. Care should be taken so that the desired message is conveyed to audience easily and in an understandable manner.

3.3. Choice of type fire model type

The discussion presented in the earlier chapter is used as a guide to select the type of wildfire model to be utilised for this research. We had three kinds of models namely probabilistic, deterministic and stochastic models for modelling wildfires.

Probabilistic models adopted the theory of probability for determining whether the fire would spread or be contained within a grid based cell. These models needed to be mainly developed and customised for a particular area and were not entirely mathematical, so this was not chosen for our purpose of research.

Empirical models also known as stochastic models were very accurate as they were specifically developed understanding the nature and its variables. These models require an extensive set of datasets to determine the parameters required for performing fire simulation. Those variables and the historical dataset are not available for this research, so this type of model was not the best choice in our case. These types of models, however, had a major flaw of being incompatible with the areas other than their initial development

area. Even if they were to be used in other areas, they would require recalculation of parameters to suit the new space.

Other types of model is a subclass of a deterministic model called as a physical model. As described earlier, it uses physical equations to model fire behaviour. Their usage of differential equations with numerical solutions often makes them complex in nature. This complex nature requiring numerous computations makes it only suitable for small areas and laboratory scales. Also, it needs finer fuel and meteorological datasets which are often not available in most of the cases. Despite being analytic and accurate in smaller scales, these types of models are not suited because of their computation overhead and detailed data requirement. Also for more accurate representation, the physical equations tend to be even more complex which makes them almost impractical in our research.

Deterministic models also have another type of model known as a semi-empirical model. It uses empirical data sets to fine tune the physical models. It merges the statistical correlation obtained from experimental and historical fires to implement laws of physics for modelling transfer of heat energy. The primary reasons for choosing a semi-empirical model for this research are listed below:

- Faster than real-time computation
- Lesser data requirement and easier to implement than physical models
- Ability to be incorporated in varied types of terrains
- Very accurate results on a medium scale
- Does Not require extensive testing as with empirical models

3.4. Development of Fire spread models:

3.4.1. Raster vs. Vector based approach:

As discussed in detail in the previous chapter, the vector-based approach is chosen for this research. The accuracy provided by the vector model is preferred over the resolution governed accuracy of the raster approach. Despite this approach requiring more

computation than the raster layer, the results obtained by several researchers such as Yassemi, Dragicevic, & Schmidt, 2008; Perry, 1998; Sullivan 2009b showed vector based approach was the best alternative for wildfire modelling. Raster based methods were found to have minor success in accurately representing the two-dimensional fire spread with acceptable accuracy which became even worse as the environmental conditions became more heterogeneous (French, 1992). Since wildfire is a highly dynamic event and its inputs such as meteorological conditions were often changing, the vector based approach was the preferred method as it could easily respond to temporal change in input datasets. Since raster based approach results in loss of accuracy, it was just used an input dataset type but not for the fire spread modelling.

While adopting a vector based approach for two-dimensional fire spread simulation, the ellipse was chosen as a standard fire shape. As discussed in the previous chapter, the ellipse has been found to most closely approximate fire shape at most of the scenarios (Van Wagner, 1969). Also because of its computational efficiency, it is preferred over other complex shapes such as lemniscate and tear drops which didn't provide a mathematical relationship with wind speed and direction (Richards, 1993). Research performed by Green et. al (1983) using simple ellipse approximated the fire growth empirical data as close as other shapes such as egg-shaped, fan-shaped. A simple ellipse can alter its shape and eccentricity based on changing wind and fuel conditions. This type of change happens very exponentially in the real world, and Alexander (1985) observed that ellipses could respond to the change in wind speed, slope steepness or both easily.

3.5. Fire spread simulation using Huygen's principle

A wave-based approach called Huygen's principle was selected for fire spread simulation. As it has been discussed in detail, only the justification for using it and its scientific relevance is presented here.

The core concept in Huygen's principle in wildfire simulation uses variables affecting fire spread at each point/vertex on fire perimeter to calculate ellipse size and orient it. Its

unique advantage is the calculation performed at a vertex is independent of others thus eliminating the introduction of possible bias (Finney, 1998). Finney (1998) mentions the previous application of Huygen's principle by Sanderlin and Sunderson (1975) into "radial fire propagation model". This research produced a satisfactory result when compared with observed fire growth (Finney, 1998). Later French (1992) considered four points on an ellipse which formed new fire perimeter. Huygen's principle has been successfully in fire spread simulation by several researchers such as Richards (1990), French (1992), Dorrer (1993), Wallace (1993) amongst others. Other notable computer applications of Huygen's principle for fire growth modelling are Coleman and Sullivan, 1996, Finney 1998, Richards and Bryce 1995.

For implementing Huygen's principle, the equations developed by Richards (1995) was used. Richards (1995) equations have been successfully applied in several of the current wildfire modelling systems such as FARSITE, FLAMMAP, BehavePlus, FSPro amongst others. It requires following inputs:

1) Component differential (x_s, y_s) showing orientation of each vertex on fire front
2) Maximum fire spread direction (θ)
3) Elliptical shape in terms of a, b, c

A simple elliptical shape is shown below to illustrate different components of the ellipse:

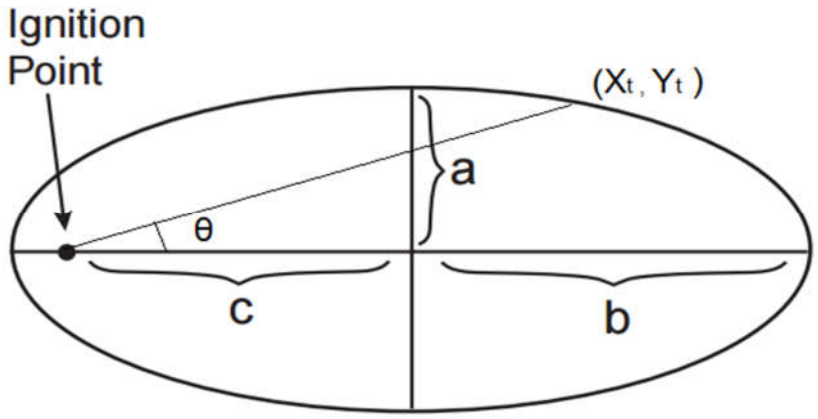


Figure 3- 1 Elliptical wavelet dimensions in fire spread

Adapted from (Finney, 1998)

The maximum fire spread direction (θ) is obtained from resultant wind and slope vector discussed later in this chapter. Now using these inputs Richards (1995) has computed orthogonal rate of spread differentials in meters per minute at time t as X_t, Y_t

$$X_t = \frac{a^2 \cos\theta(x_s \sin\theta + y_s \cos\theta) - b^2 \sin\theta(x_s \cos\theta - y_s \sin\theta)}{(b^2(x_s \cos\theta + y_s \sin\theta)^2 - a^2(x_s \sin\theta - y_s \cos\theta)^2)^{1/2}} + c\sin\theta$$

$$Y_t = \frac{-a^2 \sin\theta(x_s \sin\theta + y_s \cos\theta) - b^2 \cos\theta(x_s \cos\theta - y_s \sin\theta)}{(b^2(x_s \cos\theta + y_s \sin\theta)^2 - a^2(x_s \sin\theta - y_s \cos\theta)^2)^{1/2}} + c\cos\theta$$

3.5.1. Slope and Wind Factors:

Since Richards (1995) equations were developed for flat terrain, the inputs used previously such as the orientation of each vertex (Xs, Ys) and direction of maximum rate of spread (θ) must account for sloping terrain based on their aspect (ω).

The slope correction given by Finney (1998) as D_i for vertex i with coordinates (Xi, Yi) and aspect ωi is shown below:

$$Di = [(X_{i-1} - X_{i+1})^2 + (Y_{i-1} - Y_{i+1})^2]^{½} \cos \delta_i (1 - \cos \phi_i)$$

Where δ_i refers to difference in aspect direction and orientation angle of perimeter segment defined as:

$$\delta_i = \tan^{-1}[\tan(\omega_i - \tan^{-1}[(Y_{i-1} - Y_{i+1})/(X_{i-1} - X_{i+1})])/\cos \phi_i]$$

This correction is applied to each vertex as:

$$X_s = (X_{i-1} - X_{i+1}) \pm D_i \sin \omega_i$$

$$Y_s = (Y_{i-1} - Y_{i+1}) \pm D_i \cos \omega_i$$

Once these corrections are applied, and Xt and Yt are calculated they need to be again transformed back to horizontal plane using correction factor Dr to account for difference in spread rates

$$D_r = [X_t^2 + Y_t^2]^{1/2} \cos(\omega_i - \tan^{-1}(Y_t/X_t))(1-\cos \phi_i)$$

$X_t^* = X_t \pm D_r \sin \omega_i$

$Y_t^* = Y_t \pm D_r \cos \omega_i$

Where X_t^* and Y_t^* are desired final coordinates in the horizontal plane.

Now the values of θ range from 0 to 2π for each given vertex. Finney (1998) used Rothermel (1972), and Wilson (1980) fire spread equations to compute resultant vectors for slope(ϕ_s) and the wind(ϕ_w) as follows:

$\phi_s = 5.275\ \beta^{-0.3} \tan \phi^2$

$\phi_w = C(3.281U)^B\ (\beta/\beta_{op})^{-E}$

Where:

β, βop refers to packing ratio and an optimal packing ratio of fuel bed respectively

U is the mid flame wind speed in m/s

C, B, E are coefficients based on fuel particle size derived using Rothermel (1972) equation

3.5.2. Elliptical dimension:

The major axis(a), minor axis(b) and the distance from the rear focal point to the centre of the ellipse(c) are given by Finney (1998) using the fuel models definition and model developed by Anderson(1983). Here Anderson (1983) has used the length to breadth ratio (LB) assuming the fire will take the shape of a single ellipse.

$LB = 0.936\ e^{(0.2566U)} + 0.461\ e^{(-0.1548U)} - 0.397$

Where U is the effective midflame wind speed in m/s.

One of the crucial assumption made while developing this equation is "*U is the virtual wind speed that by itself would produce the combined effect of slope and wind on fire spread rate*" (Finney, 1998). For the surface fires, this equation involving the slope and wind factors are solved to calculate U. For the case of crown fire, the mid flame wind is assumed to be almost half of the open wind (Finney, 1998).

Finney (1998) has modified the original Anderson (1983) model by subtracting 0.397 so that ellipse would take the shape of a circle with LB equal to 1 on a flat terrain without any wind.

Alexander (1985) has developed an equation to find the eccentricity of ellipse termed as head to back ratio. Head to back ratio refers to *a/c* in ellipse shown previously

$$HB = (LB + (LB^2 - 1)^{0.5})/(LB - (LB^2 - 1)^{0.5})$$

This is a crucial equation which allows us to calculate the elliptical dimensions provided the Rate of spread(R) in meters/minute as follows:

$$a = 0.5(R + R/HB) / (LB)$$

$$b = (R + R/HB) / 2.0$$

$$c = b - R/HB$$

3.6. Development of Fire behaviour models

For correctly calculating how the fire is going to behave in the natural environment, it is essential to determine so of its necessary parameters. Fire behaviour models are developed to explain fire behaviour using variables such as fire intensity, flame size and fire danger rating indices. In this study, two fire behaviour variables namely fire intensity and rate of spread are taken into consideration. Computation of other variables such as burning indices is out of the scope of this research. Using fire intensity and rate of spread we could predict wildfire behaviour and also how the determine fire spread parameters discussed in earlier section.

3.6.1. Fire Intensity:

Fireline intensity as defined earlier is used to describe the rate of energy release per unit length for a fire front (Finney, 1998). Fireline intensity is closely linked with other forms of fire behaviours such as rate of spread, flame size and burning indices. This also shows the level of potential destruction that a fire could do under current situations.

Byram first gave the mathematical definition for fire intensity in 1959 as:

$I_b = H_A R / 60$

Where

H_A: is heat per unit area in Btu/ft^2

R: rate of spread in ft/min

One important note in this computation technique used by Byram and Rothermel is that it's in English system not metric. So, while developing system codes, the units are first converted into the British system and again converted back to the metric system for the users. Fireline intensity value is affected by fuel load, its moisture as well as relative humidity and wind speed.

3.6.2. The rate of spread:

The rate of spread is the most widely used fire behaviour parameter that quantifies how fast a fire is spread. It will also determine where the fire is going to spread to in a particular time interval. The rate of spread is used to determine the elliptical parameters discussed in fire spread model section. Many equations have been developed for predicting the rate of spread of wildfires. For this study, Rothermel (1972) equation is used for calculating the rate of spread. Rothermel equation has been successfully applied in several real-world fire modelling system such as Behave/BehavePlus, FARSITE, NEXUS and so on. This model was later modified by Albini (1976) to remove some of the flaws of Rothermel model and to implement it in a computer simulation.

The equation given by Rothermel (1972) for predicting the rate of spread is as follows:

$R= [I_R \xi (1 + \phi_W + \phi_S)]/[\rho_b \varepsilon Q_{ig}]$

Where,

R: Rate of spread in ft/min

ξ : Propagating flux ratio

ϕ_W : Wind coefficient

ϕ_S : Slope coefficient

ρ_b : Oven dry bulk density of fuel in lb/ft^3

ε : Effective heating number

Q_{ig}: Heat of preignition in Btu/lb

Several subsequent calculations are required for this rate of spread equation to complete which are discussed below. Different inputs are needed for performing these calculations which are listed and described below:

Inputs determined from Fuel datasets:

- Oven dry Fuel Loading ($\mathbf{W_o}$)
- Fuel Depth ($\boldsymbol{\delta}$)
- Fuel particle surface area to volume ratio ($\boldsymbol{\sigma}$)
- Fuel particle low heat content (**h**)
- Oven dry particle density ($\boldsymbol{\rho_p}$)
- Fuel particle moisture content ($\mathbf{M_f}$)
- Fuel particle total moisture content ($\mathbf{S_T}$)
- Fuel particle effective mineral content ($\mathbf{S_e}$)
- Moisture content of extinction ($\mathbf{M_x}$)

For determining these values, the fuels data was obtained from LANDFIRE (www.landfire.gov) program. Fuel model provides us information about fuel bed characteristics which are often time-consuming and even very difficult to measure. The fuel model selection was performing keeping following consideration in mind:

- The fuel model provides several crucial information such as fuel loading, surface area to Volume ratio for each fuel group
- Provides fuel particle density and heat content of fuel information
- Also provides moisture of extinction information
- Does Not merge classes so that diverse fuel types are classified into single one

3.6.3. Fuel Models

Fuel model was initially developed under National Fire Danger Rating system (NFDRS) to provide standard names and classes for determining fuel loading as well as other fuel parameters. It classified the fuel information into 20 different classes as shown below:

NFDRS Model	Fuel Name
A	Western grasses (annual)
C	Pine-grass Savanna
D	Southern rough
E	Hardwood litter (winter)
F	Intermediate brush
G	Short needle (heavy dead)
H	Short needle (normal dead)
I	Heavy slash
J	Intermediate slash
K	Light slash
L	Western grasses (perennial)
N	Sawgrass
O	High Pocosin
P	Southern pine plantation
Q	Alaskan black spruce
R	Hardwood litter (summer)
S	Tundra
T	Sagebrush-grass
U	Western Pines

Table 3- 1 NFDRS fuel model

The detailed fuel model parameters for NFDRS is given in Appendix A- 1

Currently, this model data is not updated and has been replaced by current models namely Albini's 13 Fuel model and Scott and Burgan's 40 fuel model. Both these models have been developed in such a way that they could be used in Rothermel spread computation.

Albini's fuel model initially developed in 1976 was expanded by Anderson (1982). This model has classified fuels into 13 classes and has also provided photographs and

conversion charts for easier identification and conversion between other similar models (Albini, 1976). The classification has four groups and 13 classes as given below:

Model Number	Name
Grass Group:	
1	Short Grass
2	Timber Grass and Understory
3	Tall Grass
Shrub Group:	
4	Chaparral
5	Brush
6	Dormant Brush
7	Southern Rough
Timber Group:	
8	Compact Timber Litter
9	Hardwood Litter
10	Timber Understory
Slash Group:	
11	Light Slash
12	Medium Slash
13	Heavy Slash

Table 3- 2 Albini 13 Fuel Model
Source: Albinit(1976)

The fuel parameters given by Albini's Model is presented in Appendix A-2

Scott and Burgan's Dynamic fuel model is an attempt to incorporate dynamic fuel load calculation. It eliminates the flaws in the previous method which assumed fuel bed would remain uniform all year long. It has used the concept of the live herbaceous load which is transferred to dead using live herbaceous moisture content (Scott, & Burgan, 2005). Also, this model is more distributed to account for fine classification of fuels. The concept of curing coefficient has made possible even more realistic fire behaviour modelling especially for the herbaceous fuel loads which is likely to behave differently in different seasons (Scott, & Burgan, 2005). In addition to these, it also provides comprehensive documentation of the parameters and classification criteria along with photographs. Scott

and Burgan (2005) have also developed a comparison chart for making them compatible with Albini's 13 fuel model. This model has classified the fuel types into 40 different classes shown in Appendix A-3

This 40 fuel model has been chosen for this research primarily because it is very comprehensive and does not require custom fuel models to be developed. Also, it is dynamic and covers a broad range of seasons (Scott, & Burgan, 2005). It has lesser gaps in classification as compared to Albini's 13 fuel models and also provides more fuel model choices.

Scott and Burgan (2005) have also provided a fuel model parameter table as shown in Appendix A-4 which allows us to determine fuel parameters such as:

- Oven dry Fuel Loading (W_o): Given as Fuel load (t/ac)
- Fuel Depth (δ) : Given as Fuel bed depth (ft)
- Fuel particle surface area to volume ratio (σ): Given as SAV ratio in $1/ft^3$
- Fuel particle low heat content (h): Given as Heat content in BTU/lb
- Moisture content of extinction (M_x): Given as Dead fuel extinction moisture(percent)

Other parameters which are not listed as fuel model parameters were calculated by Scott and Burgan (2005) using empirical experiments. The values given by them for different variables are given below:

- Oven dry particle density (ρ_p) = 32 lb/ft^3
- Fuel particle total moisture content (S_T) = 5.55 %
- Fuel particle effective mineral content (S_e) = 1.00 %

3.6.4. Fuel particle moisture content (M_f)

Despite all these variables none of these models gives us any information about the fuel particle moisture content. Fuel moisture plays a crucial role in governing fire behaviour. Rothermel (1983) mentions the role of fuel moisture and its importance in calculating fire

intensity as well as the heat required for igniting any fuel element. For correctly modelling fire spread, fuel moistures must be approximated scientifically. The fuel moisture that measured in the field would be the best choice for fire modelling. However, due to lack of data, results obtained from an empirical observation made by Burgan (1979) and Blackmarr(1971) along with others were used by National Wildfire Coordinating Group(NWCG) to publish a chart to estimate fuel moisture based on relative humidity and temperature. This estimation chart is shown in Appendix A-5.

Since we could acquire near real time relative humidity and temperature, the fuel moisture can be approximated in near real time. Furthermore, this moisture value needs to be corrected for slope, aspect and elevation (Rothermel, 1983). Also, the usage of shading value and seasonal correction should be applied to those values. Different correction values for different months of year allows us to account for the change in herbaceous fuel moisture content of the vegetation. The correction tables are given in Appendix A-6, A-7 and A-8.

Some of the fuel models also require live fuel moisture. Live fuel moisture could be directly measured or acquired from secondary sources such as National Fire Danger Rating Station(NFDRS) or could be estimated using indicator table. Since it was not feasible to measure directly and NFDRS data was not available following index table developed by Rothermel (1983) was used to determine the stage of vegetative development.

Stage of vegetative development	Moisture content
	Percent
Fresh foliage, annuals developing, early in growing cycle	300
Maturing foliage, still developing with full turgor	200
Mature foliage, new growth complete and comparable to older perennial foliage	100
Entering dormancy, coloration starting, some leaves may have dropped from stem	50
Completely cured	Less than 30, treat as a dead fuel

Figure 3- 2 Guidelines for estimating live fuel (foliage) moisture content
Source: Rothermel(1983)

From a long time, live fuel moisture calculation has been the most error prone part in fire behaviour estimation. Live fuel moisture has a direct impact on the estimation of fire behaviour and its analysis. Several attempts have been made by researchers using the empirical relationship between Normalized Difference Vegetation Index(NDVI), Normalized Difference Water Index (NDWI) with moisture content to estimate live fuel moisture. Most of the research showed a linear relationship between NDWI and fuel moisture (Burgan, & Hartford, 1997; Dennison, Roberts, Peterson, & Rechel, 2005). The fuel moisture content is defined as:

$FMC = (W_{fresh} - W_{dry})* 100 / W_{dry}$

With W_{fresh} referring to the fresh weight of vegetation and W_{dry}, the dry content of the same.

Several researchers such as Sims and Gamon (2003); Dennison, Roberts, Peterson, & Rechel (2005, 2006); Hao & Qu(2005) depicted that NDWI calculated from satellite images using shortwave infrared(SWIR) band at 1.24 μm could be. These shortwave bands have been found to be more sensitive to the changes in fuel moisture than near infrared bands (Hao, & Qu, 2005). NDWI allows us to quantify the vegetation water content using the reflectance values in NIR and SWIR bands. This combination of NIR and SWIR reflectance be believed to nullify the errors likely to be introduced because of leaf structure and dry matter content in the leaf (Ceccato et al., 2001). NDWI is calculated as follows:

$NDWI = (\rho_{NIR} - \rho_{SWIR})/(\rho_{NIR} + \rho_{SWIR})$

Where

ρ_{NIR} is reflectance at near infrared band and

ρ_{SWIR} is reflectance at short wave infrared band

NDWI is thus used in this research to estimate the live fuel moisture content. Now since all the variable were available following sets of equations were used to compute the rate of spread at the end.

Where

- R: Rate of spread in ft/min
- ξ : Propagating flux ratio
- ϕ_W : Wind coefficient
- ϕ_S : Slope coefficient
- ρ_b : Oven dry bulk density of fuel in lb/ft^3
- ε : Effective heating number
- Q_{ig}: Heat of preignition in Btu/lb
- Oven dry Fuel Loading (W_o)
- Fuel Depth ($\boldsymbol{\delta}$)
- Fuel particle surface area to volume ratio ($\boldsymbol{\sigma}$)
- Fuel particle low heat content (h)
- Oven dry particle density (ρ_p)
- Fuel particle moisture content (M_f)
- Fuel particle total moisture content (S_T)
- Fuel particle effective mineral content (S_e)
- Moisture content of extinction (M_x)

The detailed calculation steps along with inputs and variables is presented in Appendix A-9

Utilising all these equations and others the code base was developed. The full source code is given under Appendix C.

The overall architecture of fire model is as shown below:

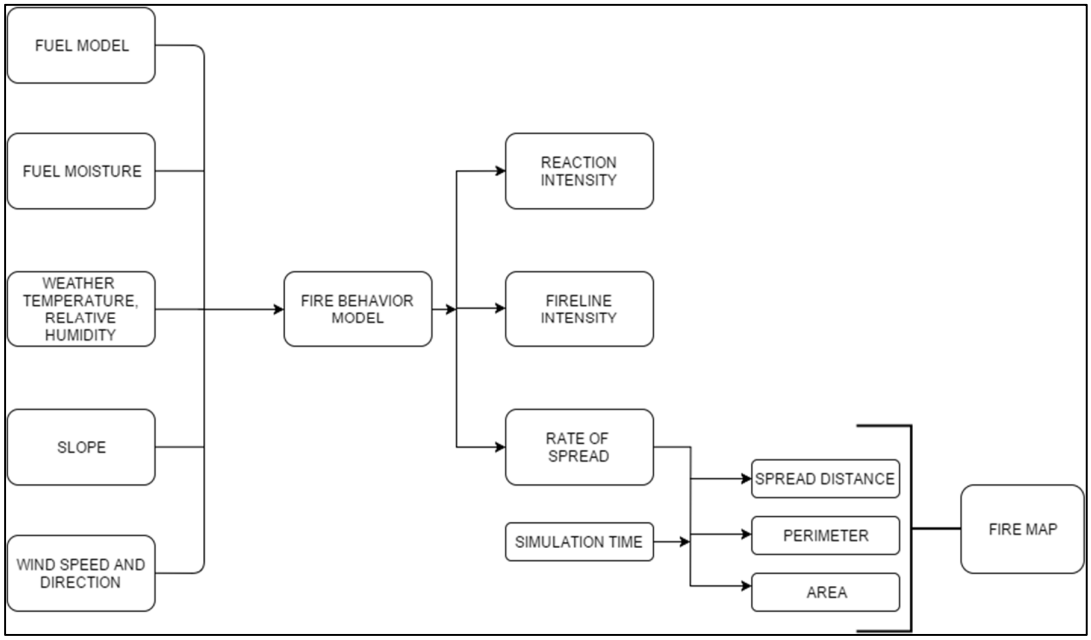


Figure 3- 3 Architecture of wildfire model

As discussed earlier, fuel dataset, fuel moisture data from comparison charts with corrections for temperature, humidity, the slope will be used to compute fire behaviour characteristics. Wind speed and wind direction determine where the fire is going to spread. Using all these inputs we could calculate reaction intensity along with fireline intensity and rate of spread values. The rate of spread is the most important variable in this research. The rate of spread will be used in conjunction with the simulation time provided by the user to calculate the propagation distance as well as fire perimeter and area covered by the fire. This spread range along with fire perimeter collectively allows us to portray a fire map to the user who could get a sense of fire behaviour on chosen time stamp.

Chapter 4: Design and Development of WebGIS based Wildfire Modelling System

From the previous chapter on fire model, the theoretical and computational aspect of fire model has been already discussed. This chapter is focused on detail discussion on the design and development of WebGIS based fire model. This chapter starts with the brief reiteration of WebGIS and fires modelling. The datasets that are going to be used and their roles are discussed later followed by an elaboration of their role on overall system architecture. The technical components and overall flow of the system are explained on the latter half of this chapter.

4.1. WebGIS and Fire modelling:

GIS has been highly regarded for its ability to model environmental processes and phenomenon. The integration of GIS in fire modelling has already been discussed in chapter one and two. The immense data volume primarily guides the choice of appropriate architecture for developing WebGIS based wildfire model and need to perform simulation in near-real time basis. Thus, custom developed Python scripts are used to model wildfire spread and visualise the spatial and non-spatial results inside the integrated WebGIS viewer at the same time. This type of architecture which is discussed later in this chapter allows the users to access standard GIS functionality and sophisticated wildfire modelling functionality at the same place. The utilisation of Geocomputation algorithms will allow the system users to access wildfire modelling features in near-real time. An independent architecture which does not rely on commercial software and client software is chosen to reach a broader set of audience. Time series based animation along with typical GIS controls allows the users to interact with the system in an intuitive way. Real-time data acquisition, their processing and analysis are kept in mind while designing the system architecture as discussed in coming sections.

4.2. Data acquisition and management:

Clark, Parks, & Crane (2002) mentions that the quality of the input data is reflected directly on the overall accuracy of the model. A properly validated and critiqued data allows us to increase the reliability of the results obtained from the model. Thus, it is highly imperative that we need to clearly identify the input data sources and properly validate before using them in our model. In the case of fire modelling, the dynamic nature of wildfire demands highly current data sources. Thus, care should be taken about the temporal accuracy of the datasets as well (Rothermel, 1972).

4.2.1. Identification of data sources:

Category	Raw data	Purpose	Source
Topography	Elevation	Calculation of FI	USGS[1]
Vegetation	Fuel Load	Calculation of FI and ROS	LANDFIRE[2]
	Satellite image	Calculation of Fuel moisture FI and ROS	MODIS[3]
Weather	Relative Humidity	Calculation of fire intensity(FI)	NWS[4]
	Temperature		
	Wind speed	Calculation of FI and Rate of spread(ROS)	NWS[4]
	Wind Direction	Predictive fire propagation simulation	
Natural Environment	Water Bodies	Fire breaks	USGS[5]
Infrastructure	Roads	Fire breaks	BLM[6]

Table 4- 1 Required datasets for fire modelling and their purpose and sources

[1] : United States Geological Survey(USGS): https://earthexplorer.usgs.gov/
[2] : Landscape Fire and Resource Management Planning Tools(LANDFIRE): http://landfire.cr.usgs.gov/viewer/viewer.html?bbox=-108.88,35.43,-104.17,41.70
[3] : Moderate-resolution imaging spectro radiometer (MODIS): https://worldview.earthdata.nasa.gov/
[4] : National Weather Service(NWS): http://graphical.weather.gov/xml/gribcut.php?var=wspd&lat1=39&lon1=-112&lat2=47&lon2=-102
[5]: United States Geological Survey(USGS) The National Map: ftp://rockyftp.cr.usgs.gov/vdelivery/Datasets/Staged/Hydrography/NHD/State/MediumResolution/GDB/NHD_M_56_Wyoming_ST.zip
[6] : Bureau of Land Management(BLM): https://www.blm.gov/style/medialib/blm/wy/resources/gis/office/transportation.Par.97044.File.dat/WYoRoads.zip

4.2.1. Topographic Data

Topographic data Slope, Aspect and Elevation were required for correctly modelling fire in undulating terrain. United States Geological Survey (USGS) EarthExplorer (https://earthexplorer.usgs.gov/) provides digital elevation model for the whole world. For this research, the freely available ASTER Global DEM was chosen as it is almost of same cell size as the fuels data which is sufficient in our cases. Although higher resolution DEM was also available for some parts of the study area, they were not used because of their gaps and resulting larger data volume requiring more resources for computation.

The acquired DEM was in GeoTiff format and was in WGS 1984 Geographic coordinate system. The data was then projected to the standard projection system used for this research. Slope and Aspect were calculated using open source Quantum GIS software. Thus, both slope and aspect have same cell size, format and projection system as the elevation layer.

4.2.2. Fuels data:

The fuels present in the surface needs to be categorised to proper fuel class to quantify for the various fuel parameters such as fuel bed depth, the moisture of extinction and heat content. As discussed earlier, Scott and Burgan's 40 fuel model has been chosen for this research because of its broad categorization of fuels. Fuels data was downloaded from LANDFIRE program (*http://landfire.gov/fuel.php*). LANDFIRE was a joint program of USDA, USFS, Department of Interior and The Nature Conservancy. It provides a common platform to acquire authoritative data sets of vegetation, fuels as well as wildland fires. The data for fuels has the option to use 13 Fuel models from Anderson as well as 40 fuel models developed by Scott and Burgan. The fuels data currently available was generated in 2014 and LANDFIRE (2015) says they are working on publishing a new version soon.

Its unique advantage is that it provides coverage for the whole United States without any gaps. This makes the future extension of the project possible. It bundles the information such as terrain descriptions, tree canopy and surface fuel models into a single multi-band raster called as Landscape (.LCP) file. This file follows OGC standard and is readable by Geospatial Data Abstraction Library (GDAL). It could also be used in popular fire modelling systems such as FARSITE, FlamMap and so on. The bands of this raster are interleaved by pixel, so it allows using various band information at the same time. It has a spatial resolution of 30 meters and uses USA Contiguous Albers Equal Area Conic Projection which is mentioned in detail in Appendix B-4.

4.2.3. Live Fuel Moisture:

To calculate the live fuel moisture content of the vegetation, MODIS satellite images were used. The primary reason for choosing MODIS over other satellites such as Landsat ™ is its better temporal accuracy of two days as compared to 16 days repeat cycle. MODIS rapid response system provides users with near real-time imagery. MODIS has been extensively used by United States Department of Agriculture and United States Forest Service to detect active fires (Sohlberg, Descloitres, & Bobbe, 2001). MODIS provides ortho-rectified images with resolution 250m, 500m and 1 Kilometres. It has 36 spectral bands to satisfy varied user needs from Atmosphere, Biosphere, Ocean, terrestrial hydrosphere, land surface cryosphere, human dimensions and so on. MODIS also provides false colour images along with true colour ones. The false colour images known as 721 are one of the most widely used MODIS product (MODIS, 2016) The bands that will be used for this research are given below:

Band	Wavelength(μm)	Bandwidth(μm)	Resolution(m)	Revisit time(days)
Band 2 (NIR)	0.841 to 0.876	841 - 876	250	2
Band 7 (SWIR)	2.105 to 2.155	2105 - 2155	500	2

Table 4- 2 MODIS bands used for NDWI calculation

Source: MODIS(2016)

The detailed list of specification for MODIS and the bands is given in Appendix B-1.

Since NIR band has a resolution of 250 and SWIR has a resolution of 500, NIR has also been resampled to the resolution of 500 for the reducing extra computation time.

Since it is not feasible for the user to download the image by themselves, an automatic script has been written to download MODIS data for Wyoming every two days and store in the hard drive of the server. Although it is realised that this does not represent most up to date fuel conditions, two days old data needs to be used for fuel moisture calculation because of unavailability of data. The source code for downloading MODIS images and calculating fuel moistures dynamically is given under Appendix C.

The data is downloaded in HDF format along with its metadata. The data is in Universal Transverse Mercator (UTM) which is transformed to Albers conic equal-area projection as mentioned in Appendix B-3. The data downloaded for this purpose is from MODIS Terra MOD09, the detailed description of which are given in Appendix B-2.

4.2.4. Weather Data:

One of the critical input for wildfire spread simulation is the weather data. Weather data such as wind direction, wind speed, relative humidity and temperature are always changing. To improve prediction of fire behaviour, we need to be updated information. Also, the weather data are acquired at one point so care should be taken while using that data for modelling future scenarios. Thus, it is required that the weather data should also consist of predictions that have been modelled for future as well. National Weather Service (NWS) provides weather data for a current time along with forecasts valid for 3-hour time span for next 56 steps. Thus, it provides us with time stamped weather data with a prediction for nearly 168 hours. Also the data is updated every 3 hours, 8 times a day and as soon as the new data is published at its predefined Coordinated Universal Times(UTC) (00:00, 03:00, 06:00, 09:00, 12:00, 15:00, 18:00, 21:00) the script given at Appendix C-3 downloads the updated data and uses in model computation. In the case when the user chooses simulation

time more than 3 hours, the forecasted weather conditions are used after that. Each band has timestamp interval for its validity which is crucial for automatically choosing weather information as model progresses. Also, if the user simulation time exceeds the forecasted data availability, the last predicted data (56th interval, 168th hour) will be used.

The downloaded data has a spatial resolution of 5079.406, 5079.406 meters in both X and Y direction. The projection system utilised by the dataset is Lambert Conformal Conic with two standard parallels as detailed in Appendix B-5. The data is provided in Gridded Binary Data Edition 2(GRIB2) format using Hypertext transfer protocol (HTTP). GRIB2 format follows OGC standard and facilitates easier conversion to commonly used raster formats such as GeoTIFF, NetCDF.

The datasets downloaded from NWS along with sample REST URL is given below.

4.2.4.1. Wind Speed

The wind speed measured at 10 meters from surface is given in meters/second and downloaded using the following URL

http://graphical.weather.gov/xml/gribcut.php?var=wspd&lat1=39&lon1=-112&lat2=47&lon2=-102

Where

- var is the weather variable being downloaded
- lat1 is the lower latitude
- lon1 is the smallest longitude of the desired extent
- lat2 is the highest latitude of the desired extent
- lon2 is the largest longitude of the desired extent

Since it had a limitation on downloading the whole data at once, the area was split into smaller indexes where were used to download the data. Those individual parts were merged and re-projected into project's projection system using GDAL as shown in Appendix - C

4.2.4.2. Wind Direction

The wind direction in compass degrees is downloaded using the following URL: *http://graphical.weather.gov/xml/gribcut.php?var=wdir&lat1=39&lon1=-112&lat2=47&lon2=-102*

4.2.4.3. Temperature

The temperature in degrees Kelvin is downloaded using the following URL: *http://graphical.weather.gov/xml/gribcut.php?var=temp&lat1=39&lon1=-112&lat2=47&lon2=-102*

4.2.4.4. Relative Humidity

Relative humidity in percentage is downloaded using the following URL: *http://graphical.weather.gov/xml/gribcut.php?var=rhm&lat1=39&lon1=-112&lat2=47&lon2=-102*

4.2.5. Other Data

4.2.5.1. Roads

Roads dataset is used to delineate fire break. The presence of roads brings discontinuity to the fire event. Also, roads do not have fuels to start or sustain an ignition. Roads data is downloaded from BLM website in shapefile format and is in North American 1983 geographic coordinate system.

4.2.5.2. Water Bodies

Water bodies are primarily used to designate fire breaks. Fire is extinguished when it comes in contact with the water body. It would also help facilitate the users to get a perception of sources of water that could be used for wildfire suppression. In this model, water bodies are added as a negative cost in wildfire spread. Once the fire meets a water

body, it is contained and is not allowed to move forward. Although there might be the cases of spotting where flakes would cross the water body and start a fire at another end, such type of calculation is considered out of scope for this research. Dataset consisting of water bodies is downloaded from USGS national map is of medium resolution downloaded in .gdb format and has North American 1983 geographic projection system.

4.2.6. Summary of acquired data sets:

Dataset	Format	Scale/ Resolution	Projection	Temporal Resolution	Data Owner
Topography					
DEM	GeoTIFF	1 Arc Second	WGS 1984 GCS	Produced on Oct. 17, 2011	USGS
Vegetation					
Fuel Load	.LCP	30m	USA contiguous Albers Equal area projection	Last updated 2014	LANDFIRE Program
Satellite Image	HDF	Band2: 250m Band7: 500m	Universal Transverse Mercator	Every 2 days	NASA
Weather					
Relative Humidity (%)	GRIB2	5079.406m	Lambert Conformal Conic	8 times a day (Every 3 hrs)	NWS
Temperature (^{O}K)	GRIB2	5079.406m	Lambert Conformal Conic	8 times a day (Every 3 hrs)	NWS
Wind Speed (m/s)	GRIB2	5079.406m	Lambert Conformal Conic	8 times a day (Every 3 hrs)	NWS
Wind Direction (^{O})	GRIB2	5079.406m	Lambert Conformal Conic	8 times a day (Every 3 hrs)	NWS
Natural Environment					
Water bodies	ESRI File GDB	1:24000	North American Datum 1983 GCS	Last Updated 2016	USGS
Infrastructure					
Roads	ESRI Shapefile	1:20000	North American Datum 1983 GCS	Last Updated 2016	BLM

Table 4- 3 Summary of acquired data sets

4.2.7. Acquisition of data in near real time:

Since it is critical to use updated data for proper fire modelling, the dynamically updated datasets are acquired in near real time. Weather data such as wind direction, wind

speed, humidity and temperature are downloaded every 3 hours. The downloading of data is started by considering the model executing time in UTC. As soon as the model execution time step exceeds the currently available data previously downloaded, it downloads the updated data and uses the new one instead of the previous one.

For the case of live fuel moisture calculation using MODIS data, MODIS data is downloaded every two days automatically by a background job and stores the file using standard naming convention including date of production and validity. This date and validity of file are used by the model script to determine which data is updated, and best fits the modelling requirements. If the script doesn't find the updated MODIS image, it logs the error and uses the last available data for further processing. Other datasets are considered static for the time being and are just acquired once.

4.3. Data processing and Validation:

The datasets acquired both in near real time, and one-time basis needs to be processed before they could be used for wildfire modelling. The first task in data processing is the projection and transformation of the datasets. All the datasets used for this research are projected and/or transformed to Albers Conic Equal Area projection using North American Datum (NAD) 1983 as specified in Appendix B-3 using GDAL, OGR simple freature libraries for Python. All the datasets downloaded are inspected for their advertised resolution. For the dynamic datasets, the temporal resolution was of primary concern. The metadata for each downloaded image/data was extracted using the model script to determine the validity of the dataset being used. These metadata were again stored into the processed file to be utilised by the modelling script later on.

4.4. Development of WebGIS based wildfire model

4.4.1. System Architecture:

The overall system architecture adopted for the development of the system is given below:

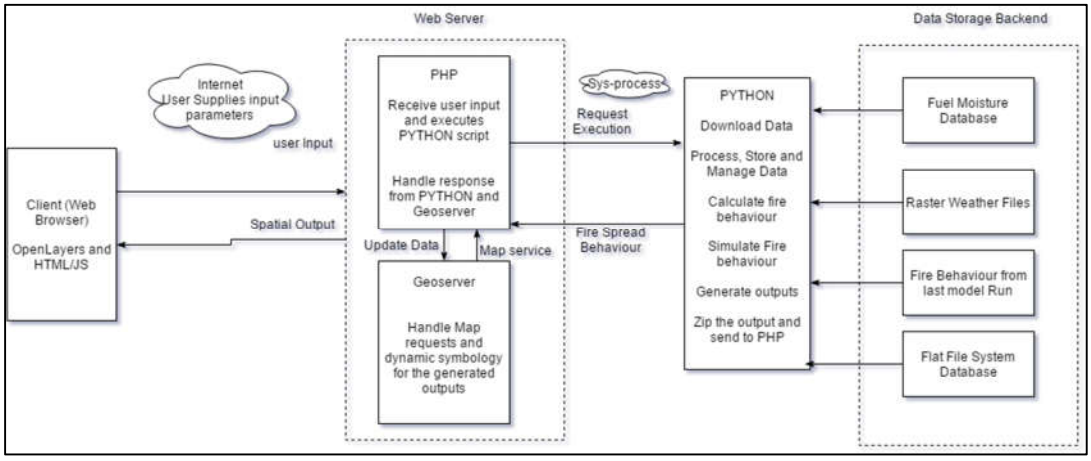


Figure 4- 1 System Architecture

This system uses client-server technology to provide wildfire modelling and simulation capabilities to the users. The users comprise of HTML web page that can run on any W3C compliant web browser. The server resides on Apache web server which runs PHP script as well as runs GeoServer using Jetty. The user requests are sent to the web server via the internet. Custom server scripts are created to handle and preprocess the user inputs. Those server scripts are created using PHP programming language and allow us to communicate with the system scripts which consists the main business logic of the research. The web server also houses GeoServer which is specifically designed to handle large volumes of geospatial data requests simultaneously. GeoServer allows us to present the results of modelling to the users easily. GeoServer is an Open-Source software that could provide Web Mapping service(WMS), Web Feature service(WFS) services to the users. One of the major reason for adding GeoServer to the application stack is to reduce the amount of data being transferred to the users so that the users could access the functionality even with low bandwidth. PHP scripts after determining and processing the user inputs execute the Python Scripts which stores the fire modelling algorithm. The python script is made in such a way that it could be run using system process command utilising the parameters supplied by the user. In this case, the parameterization of the python script is handled by the PHP script. Python script as shown in Appendix - C performs several steps to calculate fire spread extent.

The initial step for the Python script is to download the updated data and process them. Those data are stored and managed properly before using them to calculate fire behaviour variables such as fire intensity and rate of spread. Using such fire behaviour variables, fire spread is simulated, and the outputs are generated. As the model progresses, several files are created with each one for one simulation step. These records tend to be enormous and need to be properly compressed before sending back to the PHP script. Those compressed data are also stored in the form of the flat file system to be used for future retrieval.

While performing the simulation, Python script regularly reads and writes information from and to the fuel moisture database as well as the raster weather files and other secondary information stored in flat file system. The flat file system opts in this case to make future extensions possible when the data volume might increase.

Upon receiving the fire spread results, the PHP scripts invokes a script to update GeoServer indexes and send the updated results to the client. The client script utilises Open-source Javascript mapping library known as OpenLayers to present an interactive and dynamic web mapping experience to the user. While developing the client application, the usability of the system in different user environment such as smartphones, tablets, Personal computers, Laptops is also taken into consideration. To make the web page automatically adapt to different sizes of screen, responsive template from Bootstrap is used. The client application also uses Jquery UI and Jquery libraries to enhance user experience in the form of time slider and customised buttons.

The overall flow of the system execution is illustrated below:

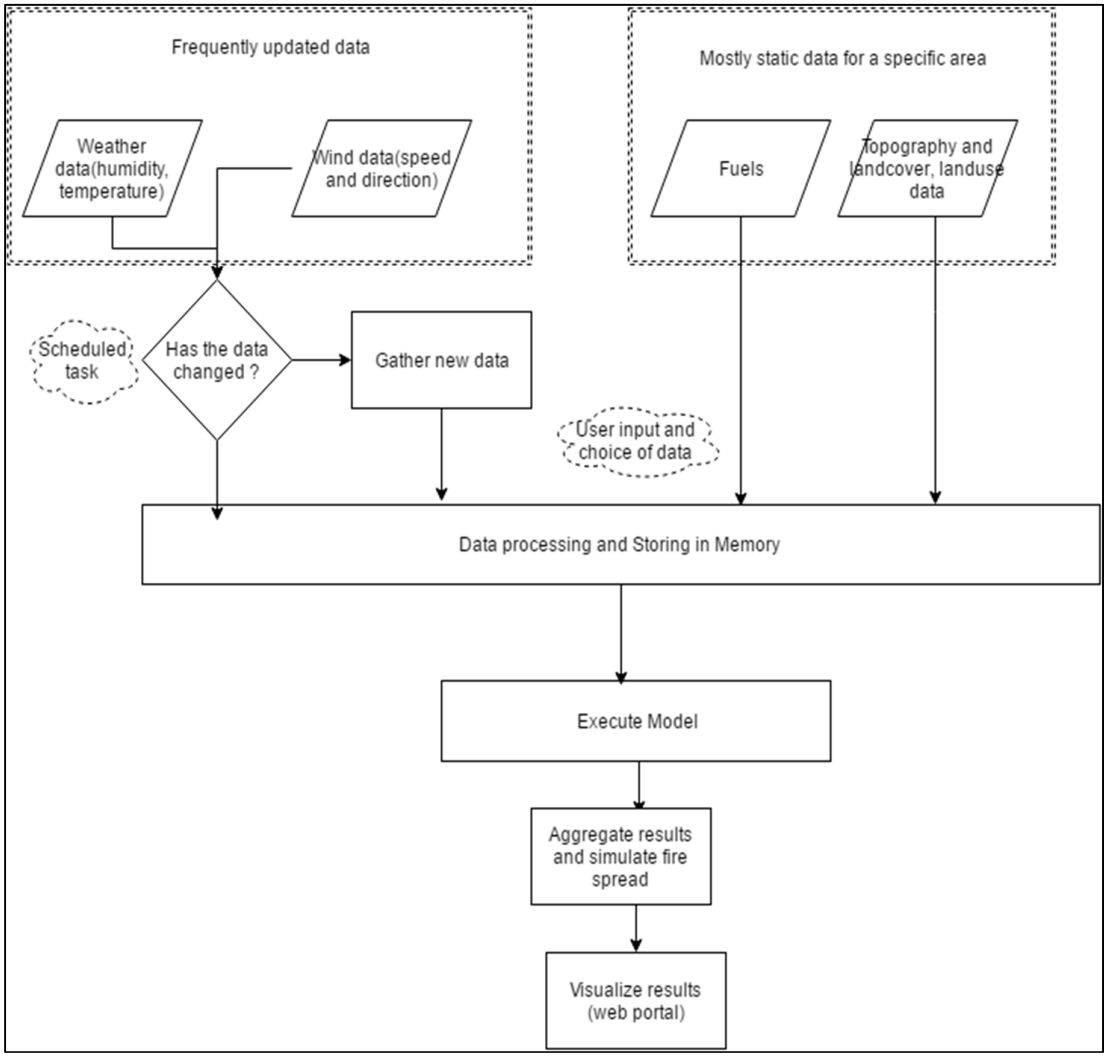


Figure 4- 2 Overview of system processes

The first process after the user selects the fire ignition location and the time and steps to run the simulation are the data acquisition. The model first inspects it needs to download the updated data or not. If the model finds out there is new data published; it grabs the updated data and stores with the appropriate filename. As discussed in the earlier section, the datasets that are being updated on a regular basis are weather data such as humidity and temperature and wind data. The fuel moisture data is also acquired at this stage and is processed and stored in memory for faster execution. The static datasets such as topography fuel models are updated manually by the developer. Those datasets are then used to execute the wildfire model which is discussed in detail in coming sections. The

results from those models are aggregated to simulate wildfire behaviour. Those simulated results are then visualised to the user using WebGIS based portal.

4.4.2. Integration of wildfire modelling system in Python:

For the purpose of developing wildfire model, Python was chosen as the scripting language. Python was primarily because of its ease of use and my previous experience. Python is officially the base programming language used by industry leading GIS software namely ArcGIS, QGIS, GRASS and SAGA GIS amongst others. The algorithm used for developing python model is given below:

4.4.2.1. Acquire inputs from the users

At first, the user input such as fire start perimeter, simulation time and simulation steps are obtained and stored as variables.

4.4.2.2. Loading of required libraries:

For the development and execution of wildfire model, several libraries are used. The descriptions of the libraries used are given below:

- fnmatch: Required for wildcard matching of filenames
- sip: Handles interaction between C, C++ libraries and python, needed to set the API level of qgis correctly
- os: All the interaction with operating system and file stored there, read/write functions
- sys: required to receive inputs from the user and exit the program once executed
- is: All the core QGIS functionalities such as geoprocessing, feature editing, reading and visualisation
- processing: library to access QGIS processing capabilities and run them from python code

- winddata_downloader: Custom developed library to download automatically, mosaic, project and store updated dynamic data discussed in earlier section
- gdal: GDAL library to read, process and write raster datasets
- geojson: Library to handle GeoJSON obtained from the user and convert to appropriate feature information
- generateellipse: Custom developed a library to generate ellipse using the elliptical parameters calculated using Rothermel model and Huygen's principle
- shapefile_methods: Custom developed methods to handle projection, renaming, storing and reading shapefile information.
- bishrant-rothermel: custom developed library that utilises Rothermel equations and dynamic fuel values to calculate fire behaviour parameters along with elliptical parameters.
- numpy: Numerical python package that handles large mathematical calculations and integrates very well with raster information read using gdal.

4.4.2.3. Calculation of Fire Spread perimeter:

The steps used for calculating fire spread perimeter is listed below:

- *Calculate fire front coordinates*: The ignition input provided by the user is split into points that reflect the fire start positions. For reducing the data volume and speeding the computation, the points are considered every 5 meters. These fire-front coordinates serve as rear focal point for the fire ellipses.
- *Determine Rate of spread*: Rate of spread is calculated for each fire front coordinate using Rothermel equation incorporated into package bishrant-rothermel. The rate of spread obtained from Rothermel equation is converted into metric units as Rothermel equation produces output in English system.

- *Determine elliptical parameters*: The size of the ellipse is calculated using the rate of spread obtained from the previous step and the wind components used to calculate Length to breadth ratio.
- *Generate points of the ellipse*: The parameters of ellipse size and its orientation and position of the rear focal point is used to create points of the ellipse. For reducing the data volume, 40 points that define the shape of the ellipse are produced for each one.
- *Merge ellipses*: All the original ellipses generated from earlier step are combined into a single fire front.
- *Send result and clean memory*: After obtaining fire front polygon, it is converted to GeoJSON object and sent back to the requesting client which is PHP script. Then all the temporary files are deleted and the memory allocated is freed up for other processes.
- *Check result and run iteration*: If the user's simulation step is more than one, the last fire perimeter becomes new fire front, and all the previous steps are executed in iteration until the desired number of steps is reached.

4.4.3. Visualisation of results:

After generating outputs using Python, the results are used by PHP and GeoServer along with OpenLayers to display interactive web map with the modelling result. The users are also provided with several controls such as zooming to the full extent of United States, going back/forward in navigation history, identify features to get detail about them, ability to switch between different base maps, zooming and panning and viewing simple help. The map interface also provides the users with a time slider which reflects its change on the simulation result being displayed on the map. The users don't need to refresh the map if they want to start a new simulation. They just need to digitise their fire start polygon, input time and step for simulation and the result of simulation will be presented to them on the same portal.

Chapter 5: Evaluation and Discussion

5.1. Introduction

Using the methodologies and tools discussed in earlier chapters, a WebGIS based wildfire simulation system was developed. Fire spread is calculated using least number of inputs from the users. The users could provide the starting ignition position of fire and the number of simulation steps along with the size of the individual steps. As a response, the users could see a time series based simulation results which are interactive as well. For predicting the spread of fires, the latest wind and environmental conditions are used. As mentioned earlier, the updated data is automatically used by the system reducing the effort on the user side.

This chapter is focused on the evaluation of the results obtained by the wildfire model and the WebGIS interface developed as a part of this research. For the assessment of results, FARSITE software was chosen. Same inputs were used to run a simulation on custom developed wildfire model as well as FARSITE.

5.2. Choice of FARSITE for comparison:

As mentioned in earlier chapters, FARSITE is the wildfire modelling system developed for simulating the spread of wildfire, especially for the United States. It has been termed as best software available for wildfire modelling because of its ability to incorporate several aspects of wildfire behaviour into a single modelling program (Finney, 1998). It utilises several inputs such as weather, fuel data, the terrain to predict fire spread across space and time. It utilises several widely adopted and intensively tested models such as Rothermel (1972) for surface fire spread, Van Wagner(1977) model for crown fire modelling and Albini (1979) model for determining spotting behaviour. Finney and Andrews (1999) states that FARSITE is capable of producing maps of fire perimeter as well as fire behaviour in both vector and raster data formats that could be used for further processing in GIS software. Numerous use cases of FARISTE have been existent in both State, as well as Private

organisations dealing with fire. The most notable ones are prioritisation of fuel treatment plans, reconstruction after fire incident and the most important one, fire growth simulation (Finney, & Andrews, 1998). FARSITE has been used as a primary resource by Firefighting agencies to study wildfire suppression options and devise tactics for attack (Finney, & Andrews, 1999). FARSITE has been used by National Park Service(NPS) since 1995 for supporting their wildfire management decision. The notable applications are fire growth predictions performed at Yosemite National Park in California and Glacier National Park in Montana which started burning in summer of 1994(Finney, & Ryan, 1995).

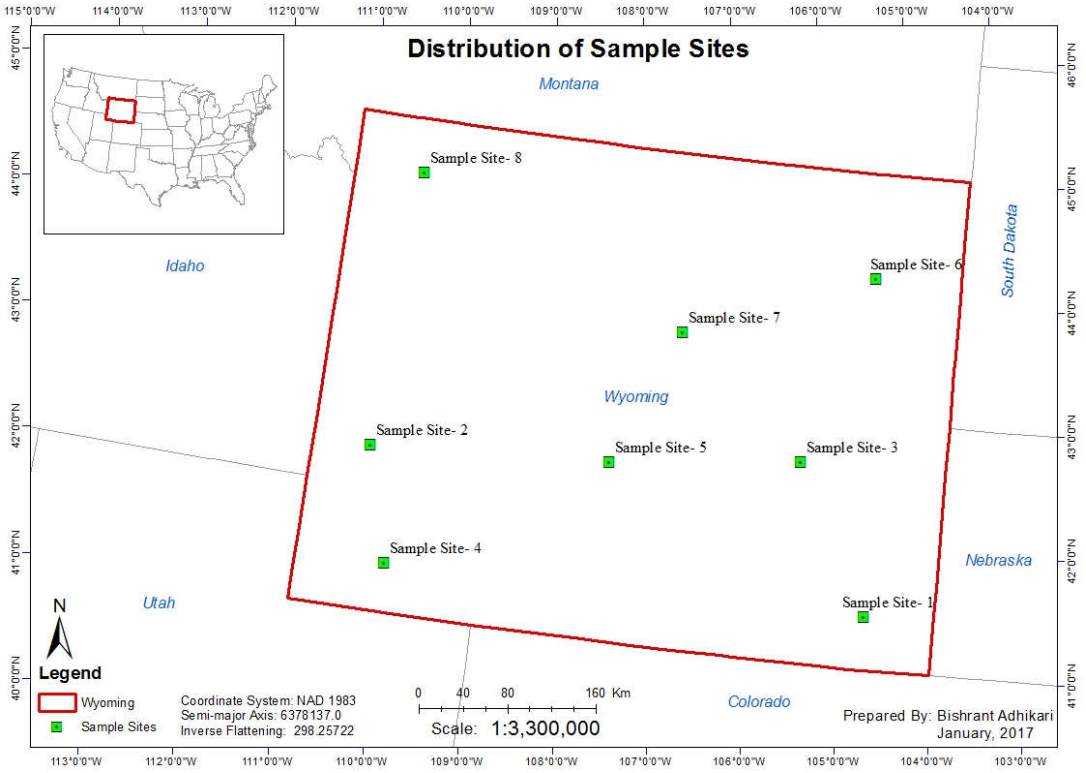
Due to its ability to simulate past fires as well as current and potential fires, it has placed itself in uniquely advantageous position than its competitors such as BehavePlus, FlamMap and FSPro. FARSITE has been used as the primary wildfire modelling system for several federal agencies in United States such as US Forest Service, National Park Service, US Department of Agriculture and several other federal land management agencies such as BLM (Finney, Brittain, & Seli, 2004). In addition to these, FARSITE has also been proved very successful in assessing effects of prescribed burns and estimating the fire behaviour for executing fire management activities. Other successful applications of FARSITE in wildfire domain are, Mistry and Berardi (2005)'s application for dry season fire modelling in Brazilian savanna, Yang et. al. (2008)'s use of FARSITE for comparison of fire characteristics amongst different vegetation types in Mark Twain National Forest. WU et. al. (2012) after using FASTIE to simulate wildfire in Fenglin National Nature Reserve in China, notes the ability of FARSITE to simulate fire spread across large spatial as well as temporal scales and reflect the current fire behaviour.

Research done by Noonan and Tueller (2008) by applying FARSITE for assessment of fuel treatment showed that FARSITE fire area simulator is very useful in predicting wildfire behaviour in sagebrush shrubland (Noonan, & Tueller, 2008). This research successfully tested the use of custom fuel models and variable weather conditions into FARSITE to simulate wildfire spread. Another study was undertaken by Ryu, Chen, Zheng, & Lacroix (2007) evaluated the capability of FARSITE for modelling surface wildfire under varied

landscape and varying levels of fuel loading in Wisconsin, USA. The researchers found the wildfire spread behaviour predicted by FARSITE reflected the variations present in the input datasets such as topography, fuel bed and weather (Ryu, Chen, Zheng, & Lacorix, 2007).

Despite its limitations such as the assumption of homogenous fuel conditions and utilisation of generalised weather conditions such as wind and humidity, it has been found to produce reasonably accurate results in homogenous conditions and complex conditions (Anderson, Catchpole, DeMestre, & Parkes, 1982; French, 1992). Despite these studies, there have also been cases where FARSITE over predicts fire spread which is mostly likely because of limitations imposed by fuel moisture data and weather inputs (Finney, 1998; Rothermel, 1972). For this purpose, FARSITE was chosen as a standard for comparison of modelling result because of its ability to use gridded weather data currently being used in my model as well as good accuracies obtained by previous researchers.

Eight different samples sites were chosen for the testing of the modelling results. The sites are selected so that they are geographically distributed all around Wyoming and also cover all the major fuel types. These sites would allow us to inspect the model results in geographically diverse as well as for different fuel natures. The distribution of the sample sites is given in map shown below:



Map 5- 1 Distribution of sample sites

The dominant fuel types of each site along with a brief description of each fuel model is given below:

Site ID	Dominant Fuel Model	Description of Fuel Model
1	GR2	Moderately coarse continuous grass, average depth about 1 foot
2	SH2	Moderate fuel load depth about 1 foot, no grass fuel present
3	GS2	Shrubs are 1 to 3 feet high, moderate grass load
4	TU1	Fuelbed is low, load of grass and/or shrub with litter
5	NB2 and NB9	Ice/Snow and Bare ground respectively
6	GR2	Moderately coarse continuous grass, average depth about 1 foot.
7	TL3	Moderate load conifer litter.
8	SB1	Fine fuel load is 10 to 20 tonnes/acre, weighted toward fuels 1 to 3 inches diameter class; depth is less than 1 foot.

Table 5- 1 Sample site and fuel types

Source: Adapted from: Scott, & Burgan(2005)

For each site, the model is run, and the similar conditions are supplied as input in FARSITE. The results from both model and FARSITE are presented here. Also, the result, when viewed on the web interface, is shown. The total area predicted each of these

processes as well as the area that is common in both predictions is also presented for each sample site.

5.3. Simulation Results

5.3.1. Sample Site-1

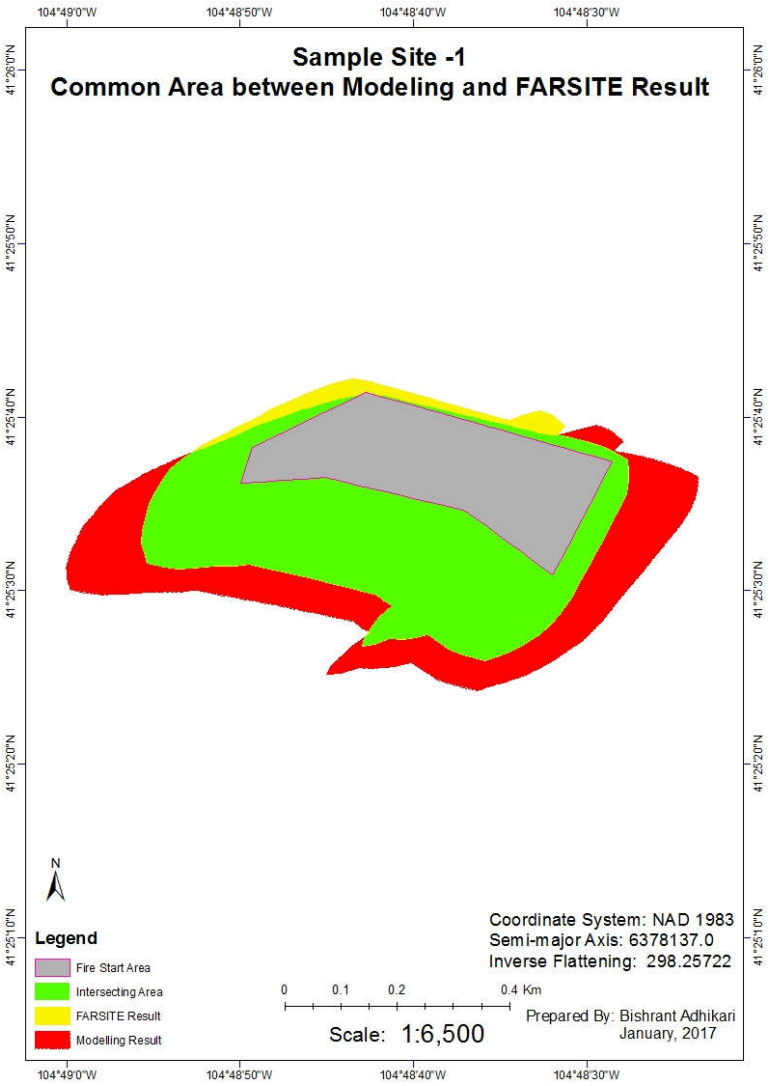
Summary of Sample site and modelling conditions:

SN	Parameter	Value
1	Dominant Vegetation Fuel Model	GR2
2	Initial Burn area	0.07 Sq. Km
3	Average Wind Speed	30 mph
4	Wind Direction	210 degrees
5	Average Fuel Moisture (1h, 10h, 100hr)	5%, 10%, 12%
6	Humidity Min, Max	25% - 50%
7	Temperature Min, Max (o Celsius)	30 - 50
8	Simulation iteration Time	2 Hours
9	Simulation Steps	10
10	Total Simulation Time	20 Hours

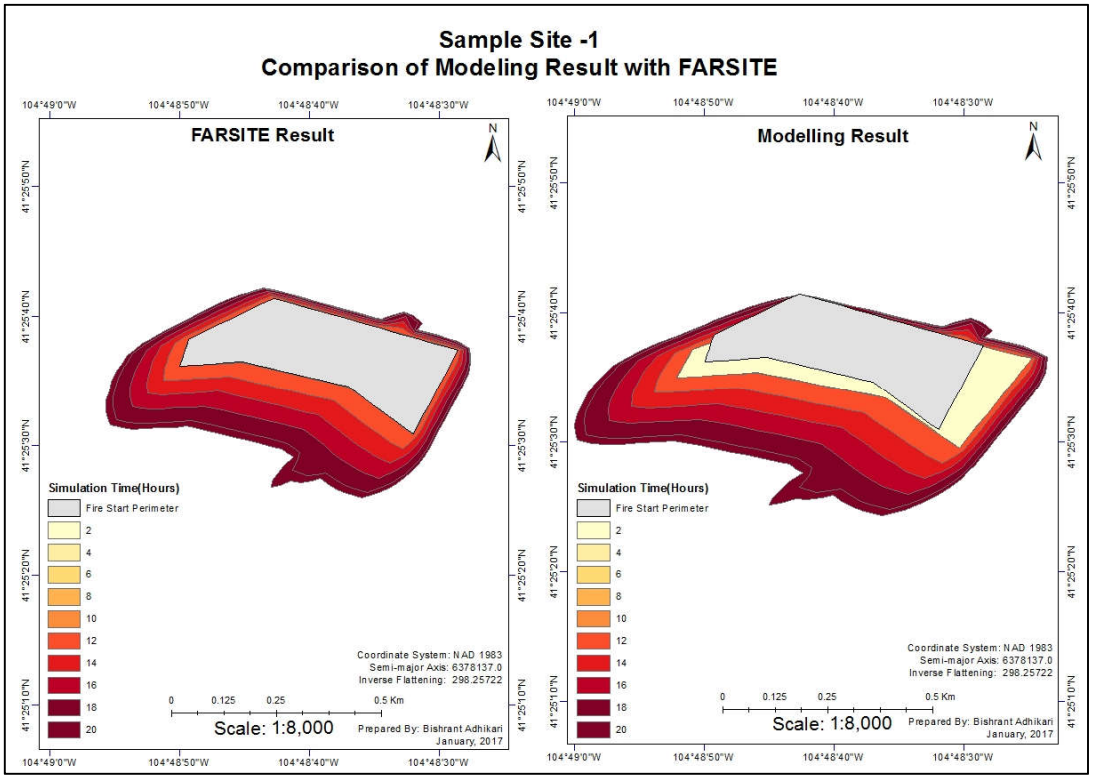
Table 5- 2 Sample site-1 summary and modelling conditions

5.3.1.1. Simulation Results

The results from modelling and its comparison with results from FARSITE are shown below:



Map 5- 2 Intersecting area between FARSITE and model results, Site-1



Map 5- 3 Comparison of Modelling result with FARSITE result: Site-1

Following were the results from this simulation run:

Site 1	**Area(Square Km)**
Total area predicted by Model	0.20
Total area predicted by FARSITE	0.13
Intersecting area in both	0.129

Table 5- 3 Sample site-1 comparative results

The results as visualised in WebGIS portal interface is given below:

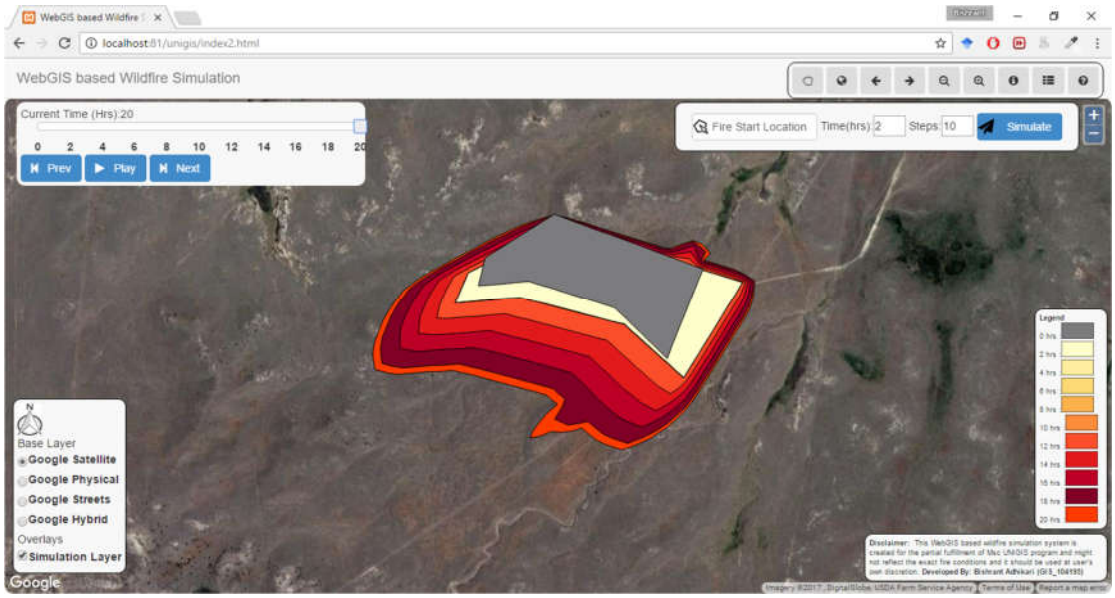


Figure 5- 1 Modelling result in WebGIS interface: Site-1

5.3.1.2. Observation:

Both developed model and FARSITE software propagated the fire in the direction of the wind. The fuel facilitated a good burn in both conditions. It was observed that the burned area given by the model was greater than the area predicted by FARSITE.

5.3.2. Sample Site-2:

Summary of sample site and modelling conditions:

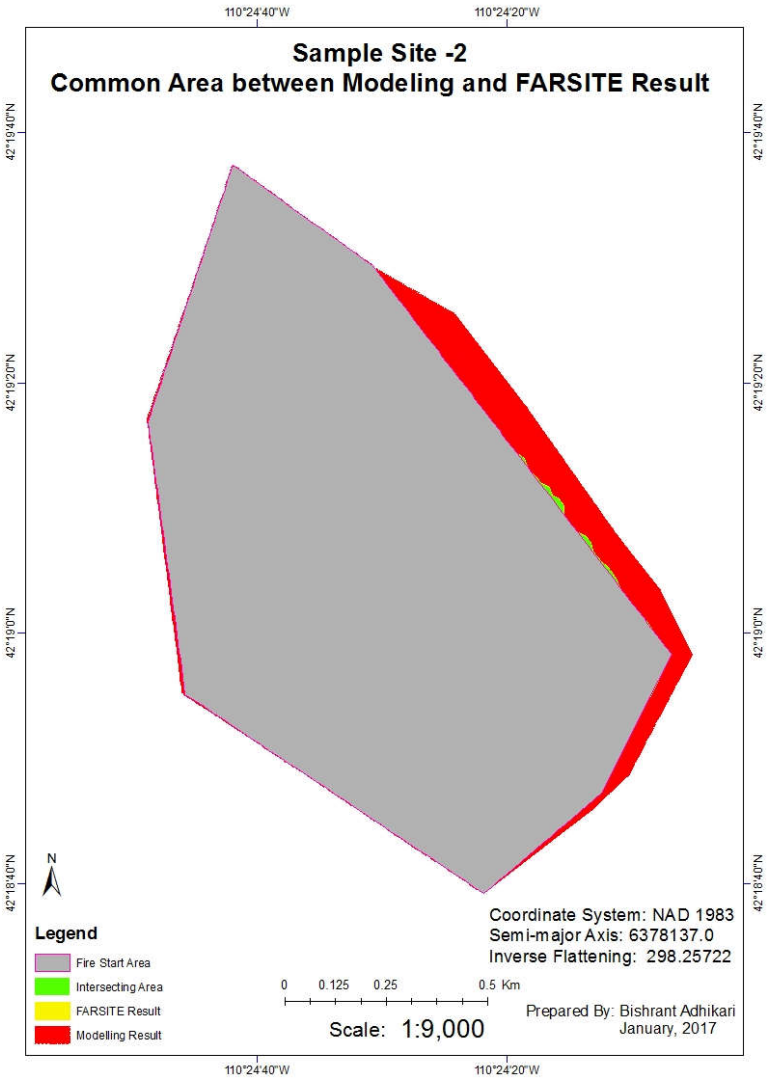
SN	Parameter	Value
1	Dominant Vegetation Fuel Model	SH2
2	Initial Burn area	0.97 Sq. Km
3	Average Wind Speed	45 mph
4	Wind Direction	90 degrees
5	Average Fuel Moisture (1h, 10h, 100hr)	10%, 13%, 25%
6	Humidity Min, Max	50% - 70%
7	Temperature Min, Max (° Celsius)	25 - 35
8	Simulation iteration Time	2 Hours
9	Simulation Steps	4

10	Total Simulation Time	8 Hours

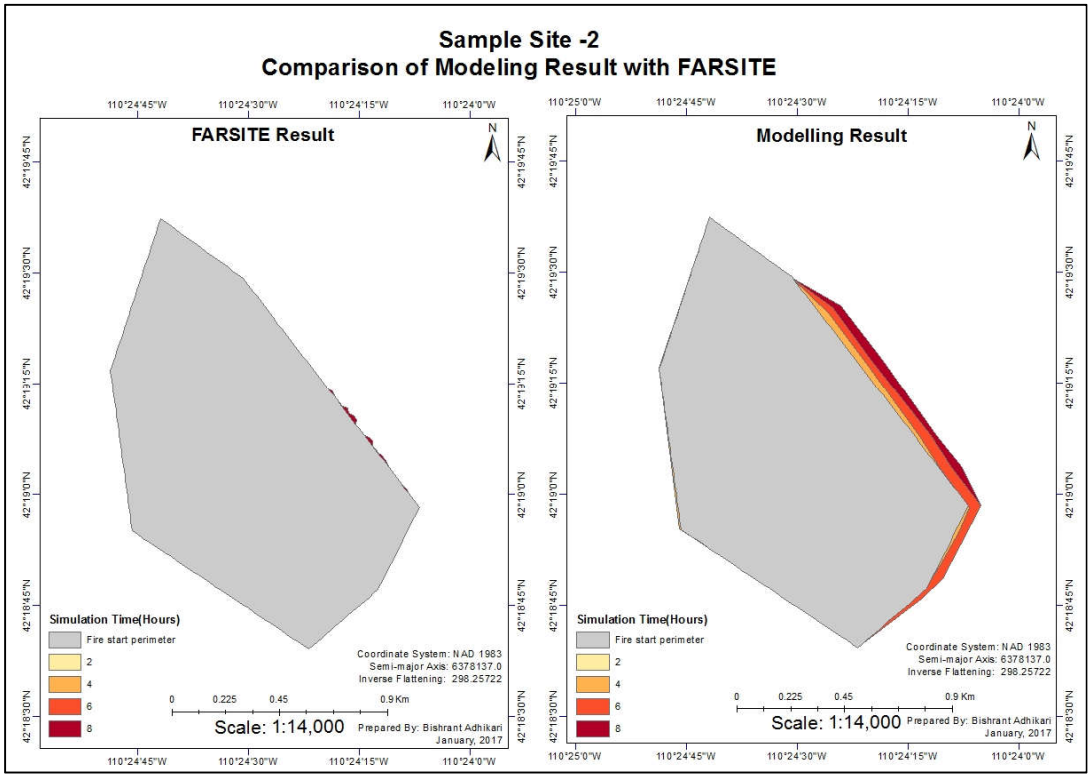
Table 5- 4 Sample site-2 summary and modelling conditions

5.3.2.1. Simulation Results

The results from modelling and its comparison with results from FARSITE are shown below:



Map 5- 4 Intersecting area between FARSITE and model results, Site-2



Map 5- 5 Comparison of Modelling result with FARSITE result

Following were the results from this simulation run:

Site 2	Area (Square Km)
Total area predicted by Model	0.08
Total area predicted by FARSITE	0.002
Intersecting area in both	0.002

Table 5- 5 Sample site-2 comparative results

The results as visualised in WebGIS portal interface is given below:

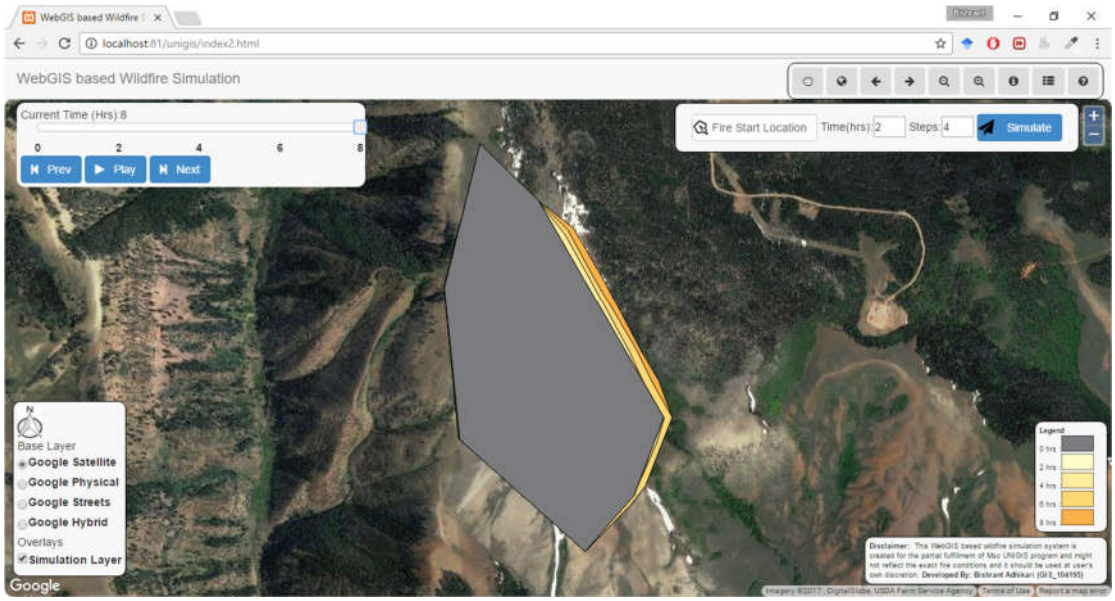


Figure 5- 2 Modelling result in WebGIS interface: Site-2

5.3.2.2. Observation:

The moderately dense shrub site eight did not facilitate fire spread. Even under strong wind conditions, the fire is pretty much contained within initial ignition area. FARSITE predicted very nominal fire spread whereas my model predicted some fire spread in the direction of the wind.

5.3.3. Sample Site-3

Summary of Sample site and modelling conditions:

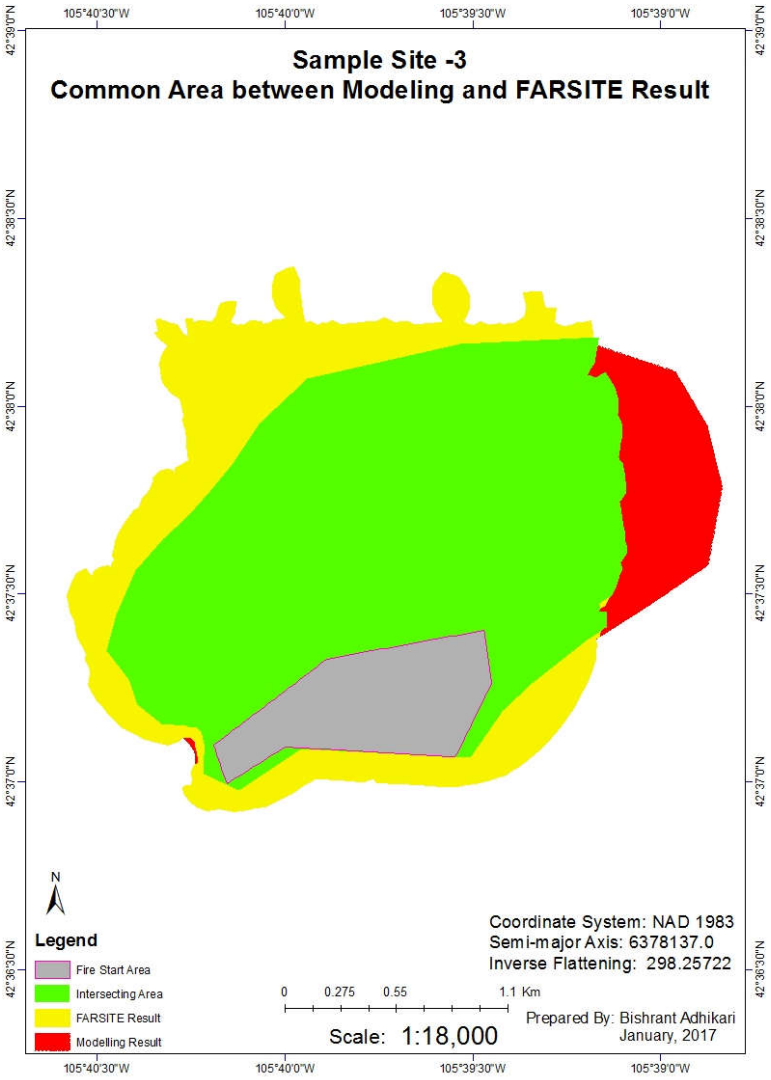
SN	Parameter	Value
1	Dominant Vegetation Fuel Model	GS2
2	Initial Burn area	0.41 Sq. Km
3	Average Wind Speed	40 mph
4	Wind Direction	30 degrees
5	Average Fuel Moisture (1h, 10h, 100hr)	5%, 12%, 19%
6	Humidity Min, Max	20% - 30%
7	Temperature Min, Max (o Celsius)	38 - 45

8	Simulation iteration Time	2 Hours
9	Simulation Steps	10
10	Total Simulation Time	20 Hours

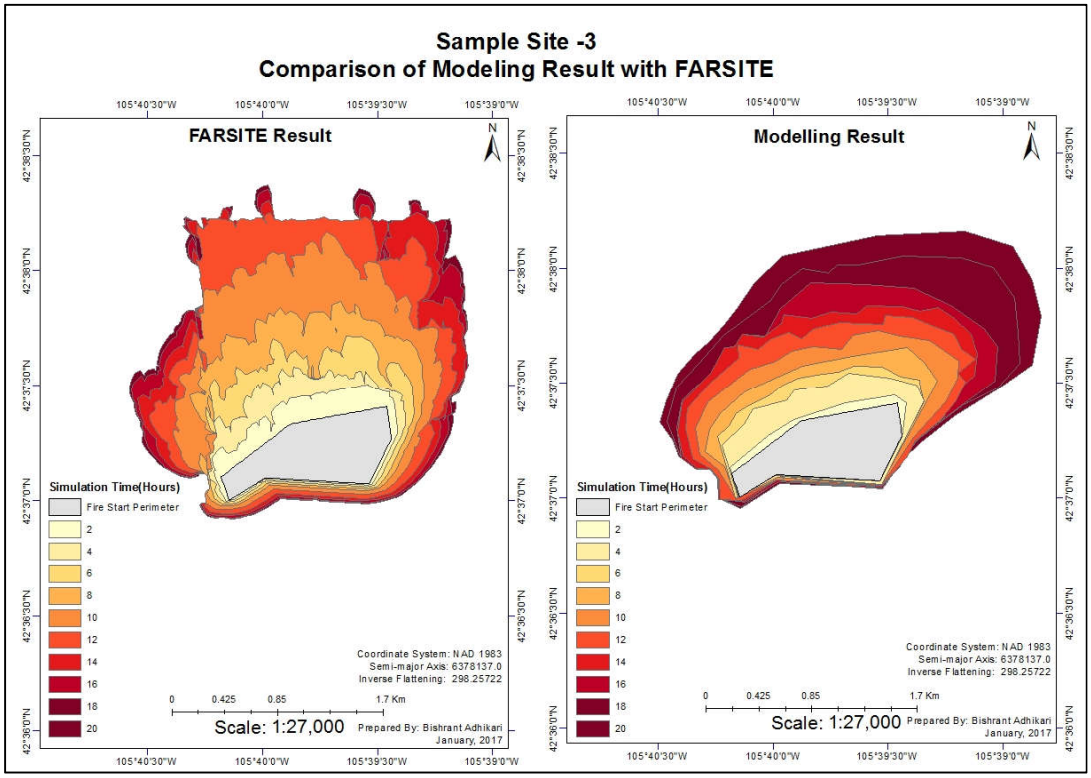
Table 5- 6 Sample site-3 summary and modelling conditions

5.3.3.1. Simulation Results

The results from modelling and its comparison with results from FARSITE are shown below:



Map 5- 6 Intersecting area between FARSITE and model results, Site-3



Map 5- 7 Comparison of model result with FARSITE, Sample site-3

Following were the results from this simulation run:

Site 3	Area (Square Km)
Total area predicted by Model	2.93
Total area predicted by FARSITE	3.50
Intersecting area in both	2.52

Table 5- 7 Sample site-3 comparative results

The results as visualised in WebGIS portal interface is given below:

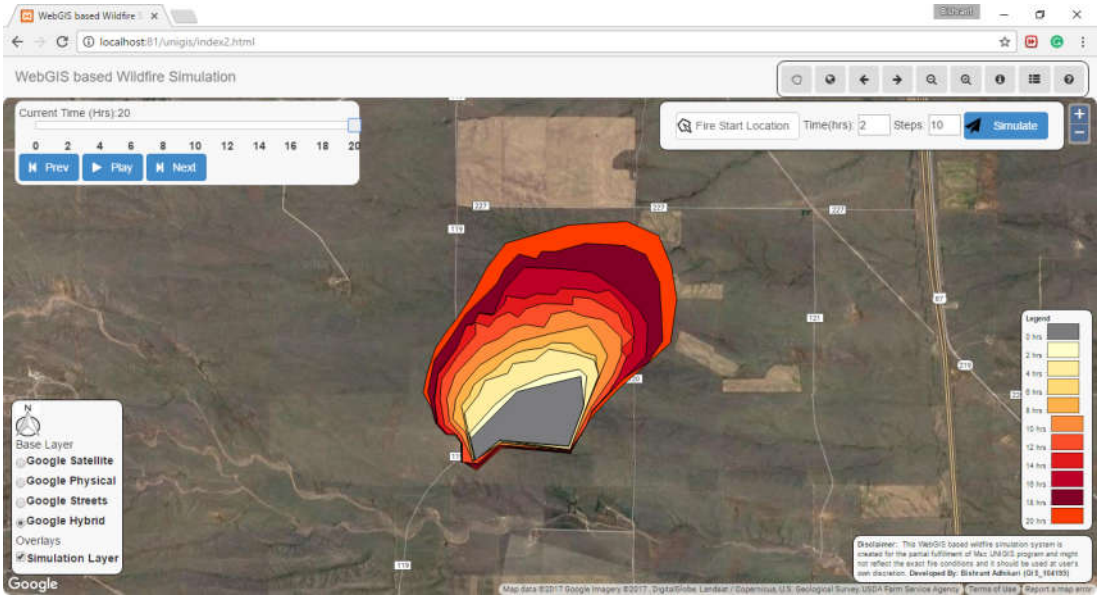


Figure 5- 3 Modelling result in WebGIS interface: Site-3

5.3.3.2. Observation:

In sample site -4, the modelling results were slighter lesser as compared to the results obtained from FARSITE. However, the overall shape of results from both the simulations is almost the same.

5.3.4. Sample Site-4

Summary of Sample site and modelling conditions:

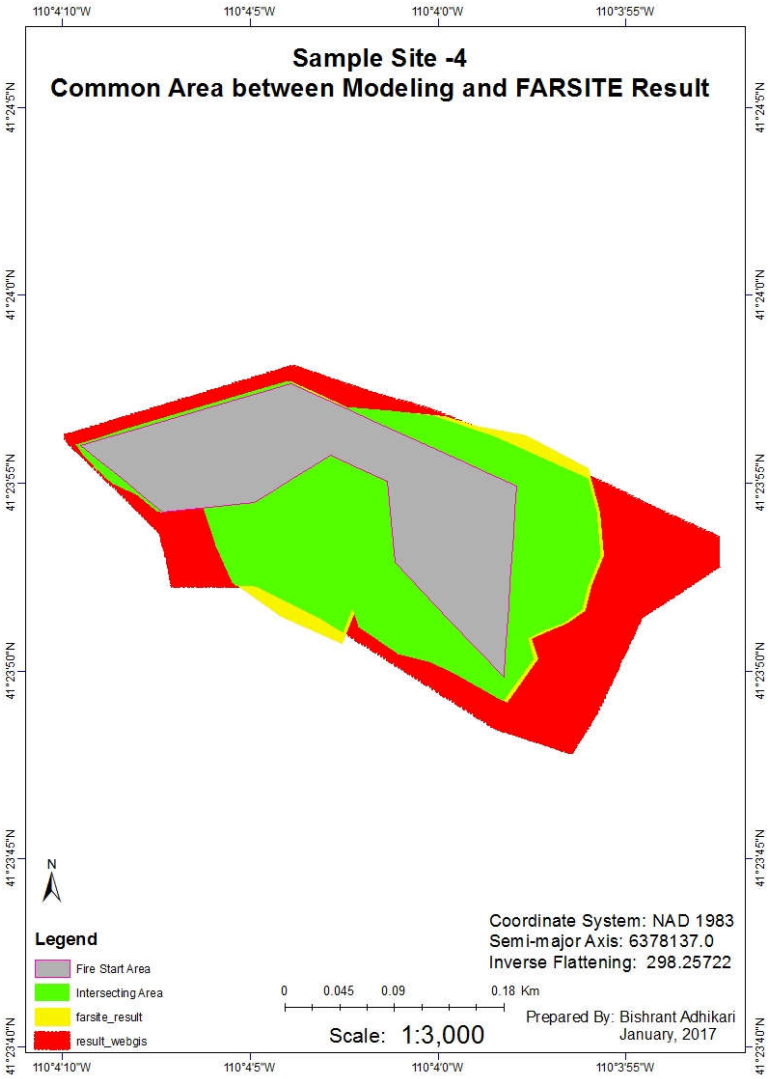
SN	Parameter	Value
1	Dominant Vegetation Fuel Model	TU1
2	Initial Burn area	0.02159 Sq. Km
3	Average Wind Speed	10 mph
4	Wind Direction	120 degrees
5	Average Fuel Moisture (1h, 10h, 100hr)	2%, 7%, 17%
6	Humidity Min, Max	55% - 65%
7	Temperature Min, Max (o Celsius)	22 - 31
8	Simulation iteration Time	2 Hours

9	Simulation Steps	10
10	Total Simulation Time	20 Hours

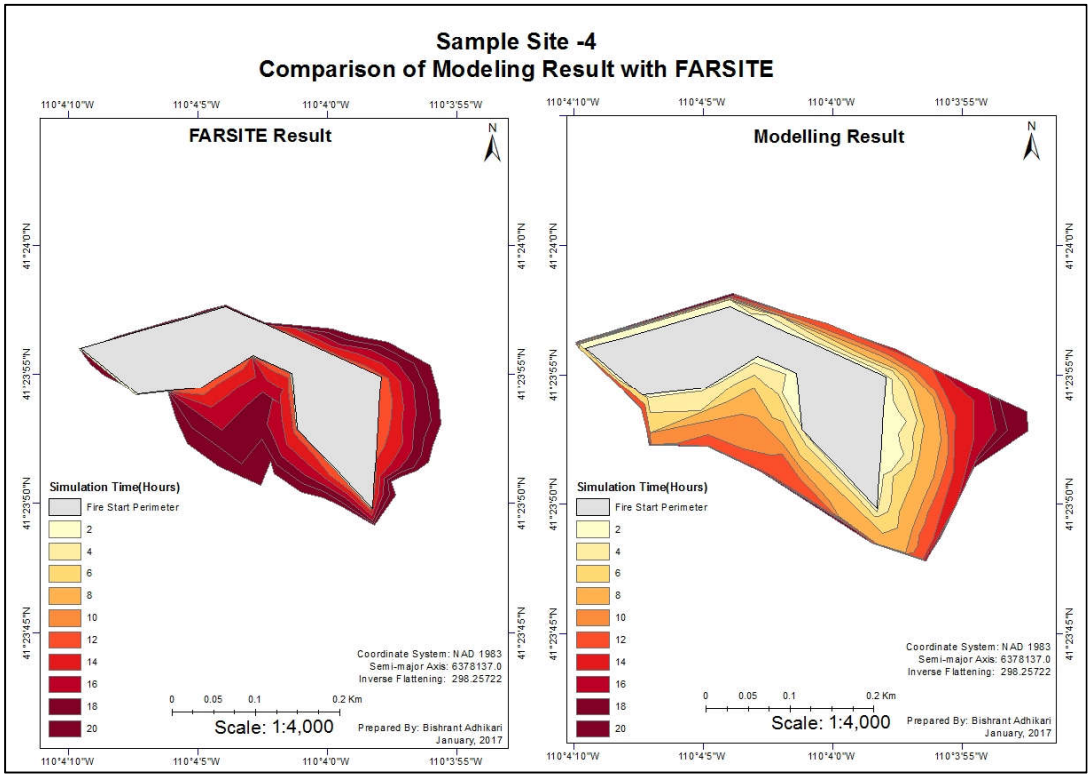
Table 5- 8 Sample site-4 summary and modelling conditions

5.3.4.1. Simulation Results

The results from modelling and its comparison with results from FARSITE are shown below:



Map 5- 8 Intersecting area between FARSITE and model results, Site-4



Map 5- 9 Comparison of Modelling result with FARSITE result: Site-4

Following were the results from this simulation run:

Site 4	Area (Square Km)
Total area predicted by Model	0.064801
Total area predicted by FARSITE	0.04792
Intersecting area in both	0.046481

Table 5- 9 Sample site-4 comparative results

The results as visualised in WebGIS portal interface is given below:

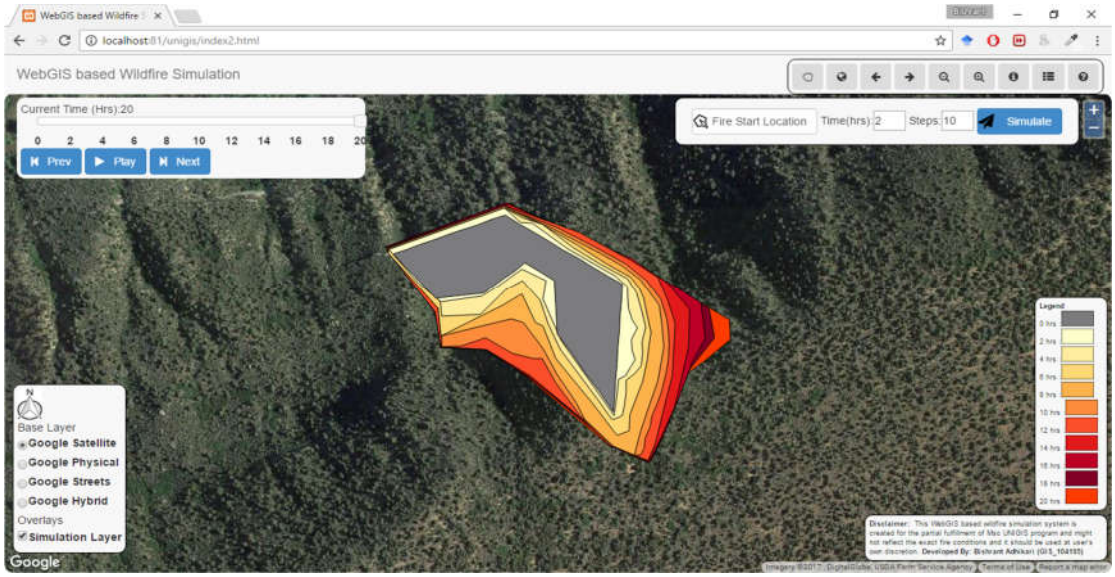


Figure 5- 4 Modelling result in WebGIS interface: Site-4

5.3.4.2. Observation:

In sample site 4, the results obtained from both model and FARSITE were almost same with slightly higher values obtained in modelling.

5.3.5. Sample Site-5

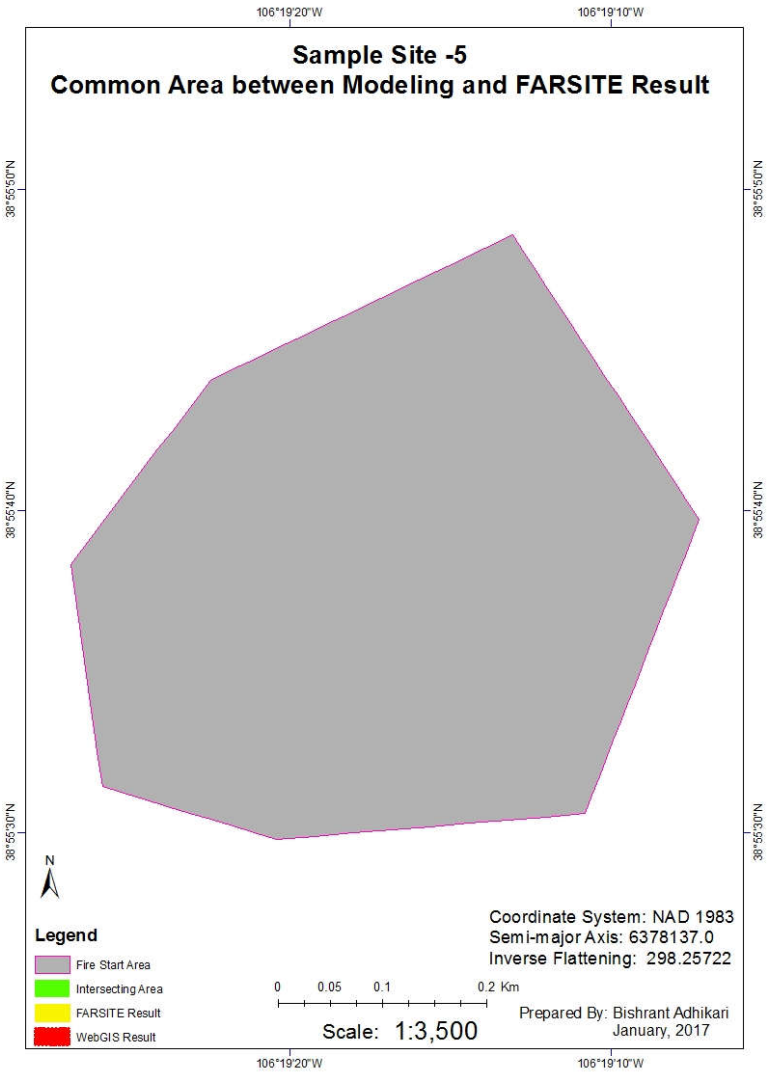
Summary of Sample site and modelling conditions:

SN	Parameter	Value
1	Dominant Vegetation Fuel Model	NB2
2	Initial Burn area	0.19 Sq. Km
3	Average Wind Speed	25 mph
4	Wind Direction	185 degrees
5	Average Fuel Moisture (1h, 10h, 100hr)	20%, 24%, 29%
6	Humidity Min, Max	22% - 29%
7	Temperature Min, Max (o Celsius)	32 - 36
8	Simulation iteration Time	1 Hour
9	Simulation Steps	5
10	Total Simulation Time	5 Hours

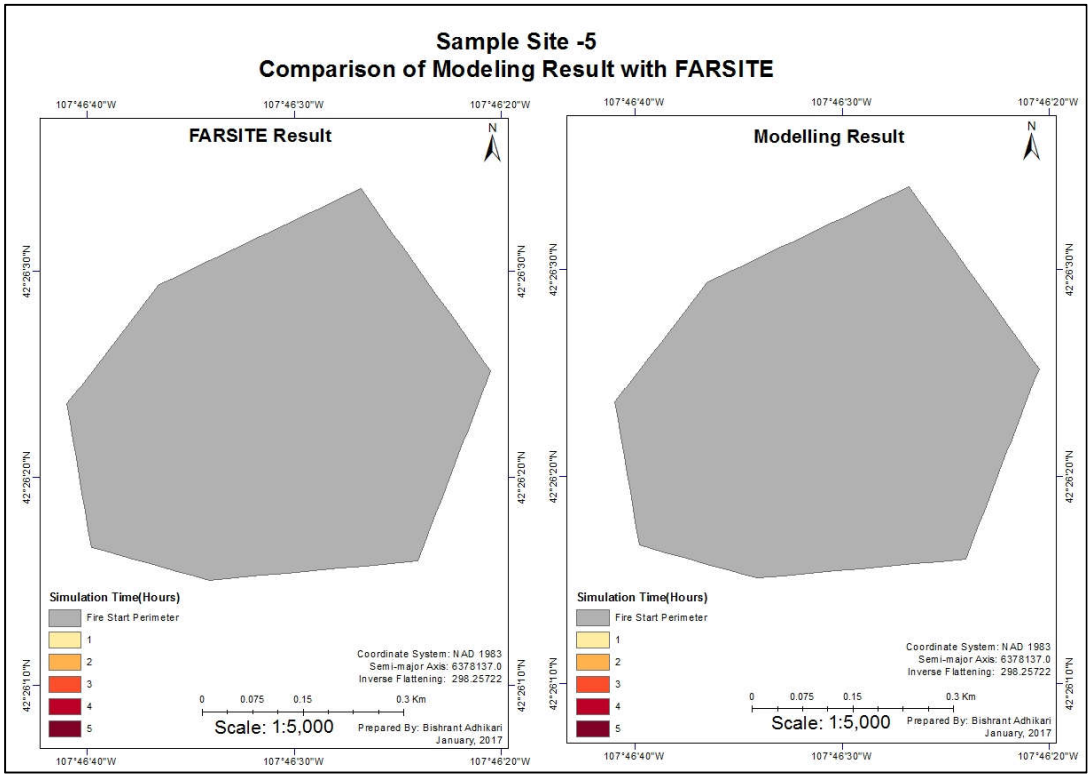
Table 5- 10 Sample site-5 summary and modelling conditions

5.3.5.1. Simulation Results

The results from modelling and its comparison with results from FARSITE are shown below:



Map 5- 10 Intersecting area between FARSITE and model results, Site-5



Map 5- 11 Comparison of Modelling result with FARSITE result: Site-5

Following were the results from this simulation run:

Site 5	Area (Square Km)
Total area predicted by Model	0
Total area predicted by FARSITE	0
Intersecting area in both	No fire spread in both cases

Table 5- 11 Sample site-5 comparative results

The results as visualised in WebGIS portal interface is given below:

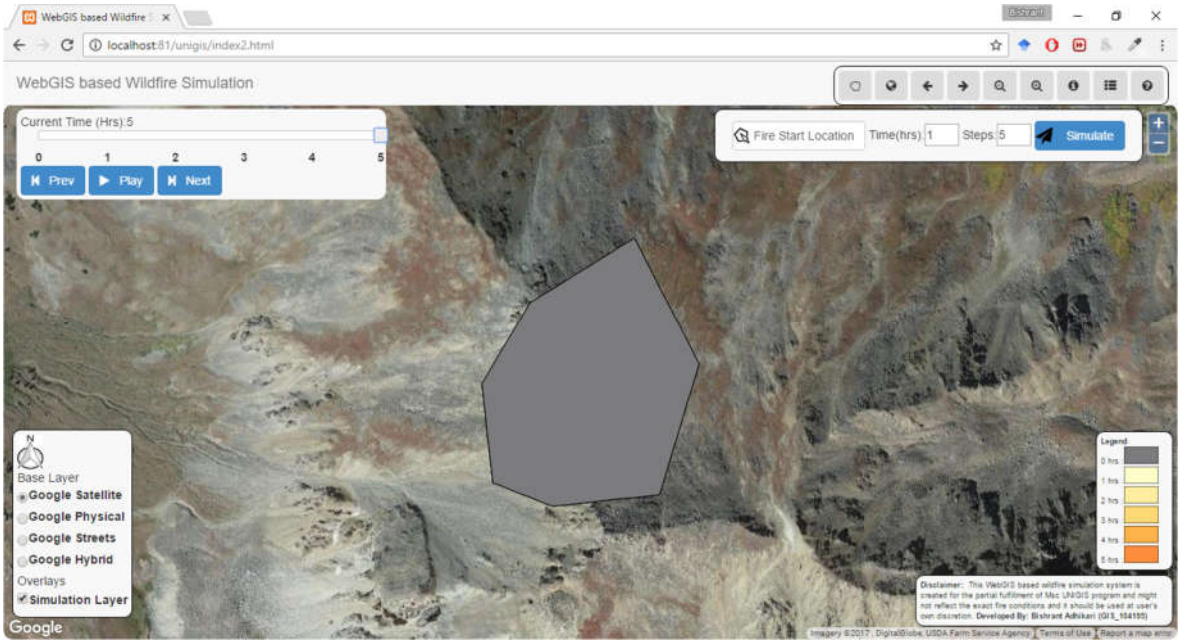


Figure 5- 5 Modelling result in WebGIS interface: Site-5

5.3.5.2. Observation:

Both developed model and FARSITE software did not propagate the fire forward because of lack of proper fuel for burning. Even though other conditions such as humidity, temperature as well as the wind were favourable for fire spread, fire is entirely contained within the initially designated ignition area.

5.3.6. Sample Site-6

Summary of Sample site and modelling conditions:

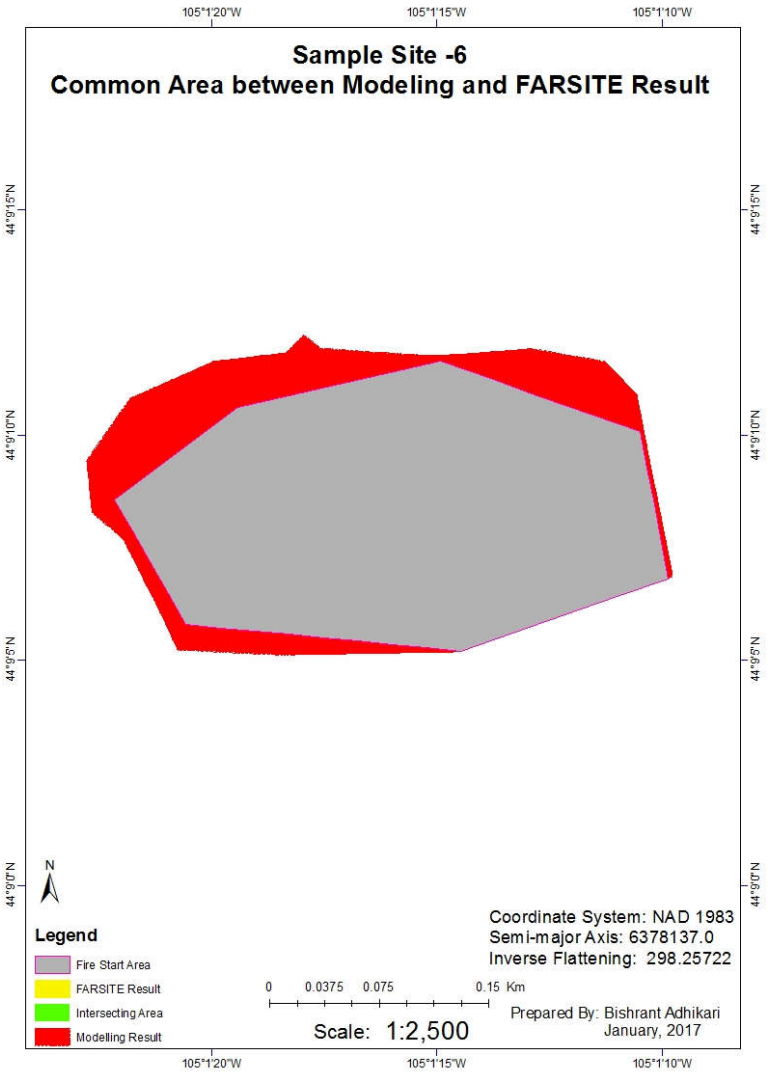
SN	Parameter	Value
1	Dominant Vegetation Fuel Model	GR2
2	Initial Burn area	0.19 Sq. Km
3	Average Wind Speed	25 mph
4	Wind Direction	185 degrees
5	Average Fuel Moisture (1h, 10h, 100hr)	20%, 24%, 29%

6	Humidity Min, Max	22% - 29%
7	Temperature Min, Max (° Celsius)	32 - 36
8	Simulation iteration Time	1 Hour
9	Simulation Steps	5
10	Total Simulation Time	5 Hours

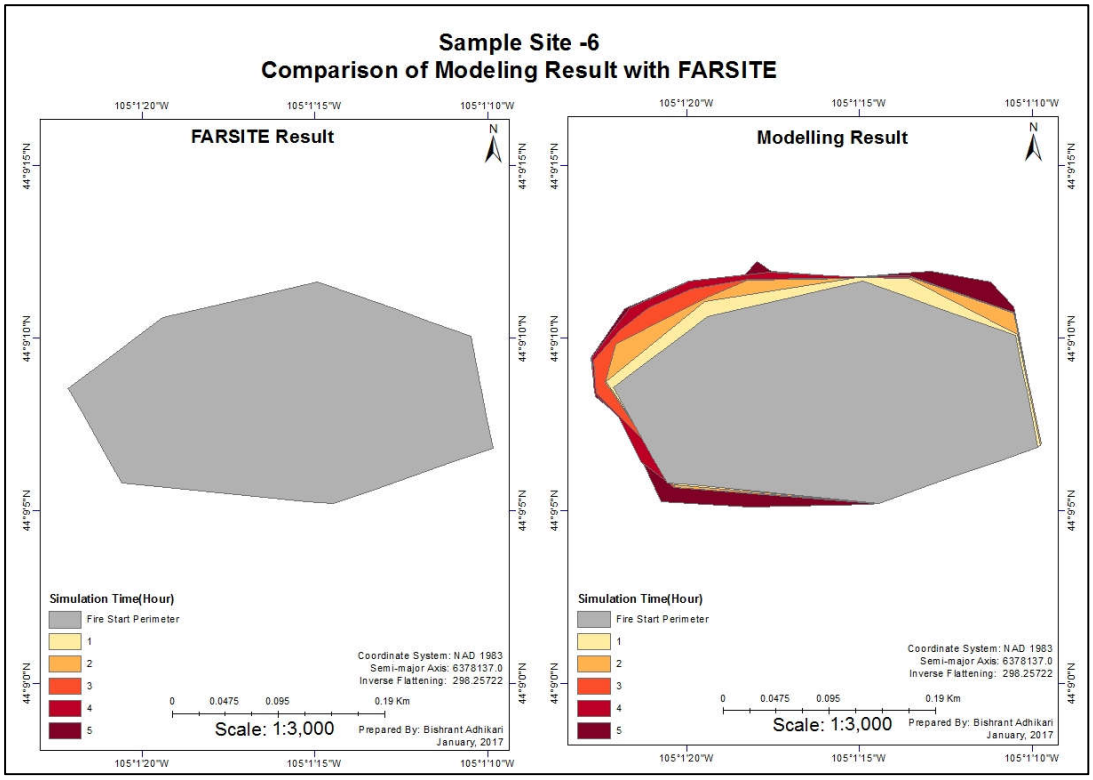
Table 5- 12 Sample site-6 summary and modelling conditions

5.3.6.1. Simulation Results

The results from modelling and its comparison with results from FARSITE are shown below:



Map 5- 12 Intersecting area between FARSITE and model results, Site-6



Map 5- 13 Comparison of Modelling result with FARSITE result: Site-6

Following were the results from this simulation run:

Site 6	Area (Square Km)
Total area predicted by Model	0.11
Total area predicted by FARSITE	0
Intersecting area in both	0

Table 5- 13 Sample site-6 comparative results

The results as visualised in WebGIS portal interface is given below:

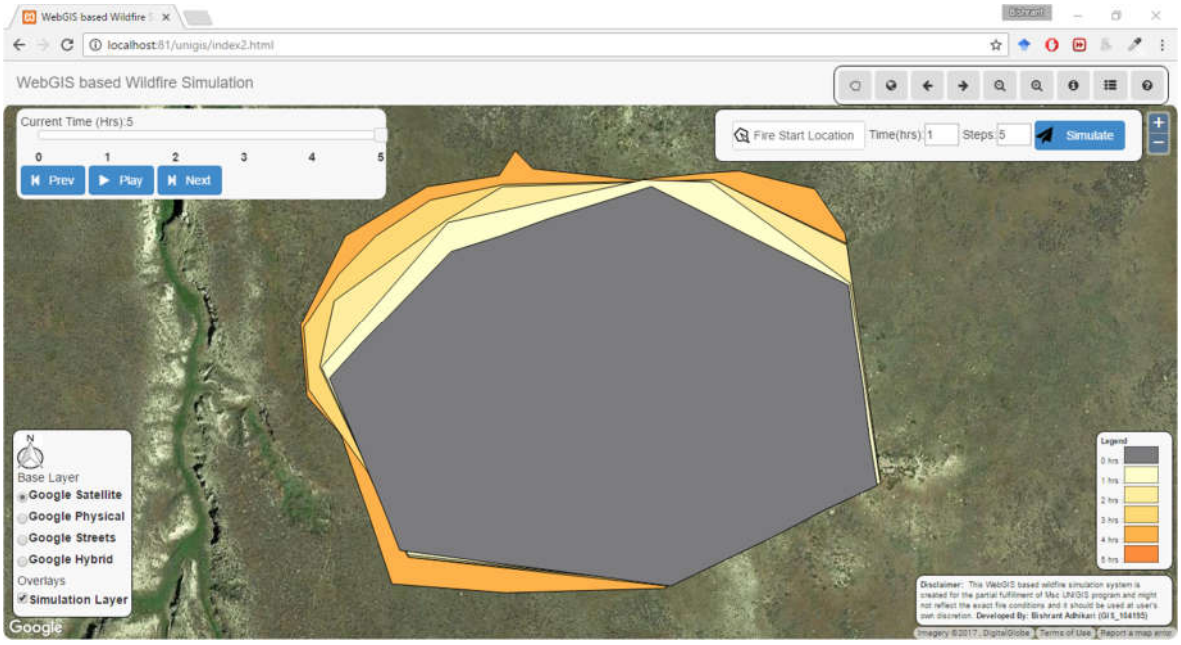


Figure 5- 6 Modelling result in WebGIS interface: Site-6

5.3.6.2. Observation:

In site 6, FARSITE predicted that the fire would not propagate further. However, my model predicted that the fire would spread slightly. The observed fire spread might be because of suitable fuel type and ignition situations.

5.3.7. Sample Site-7

Summary of Sample site and modelling conditions:

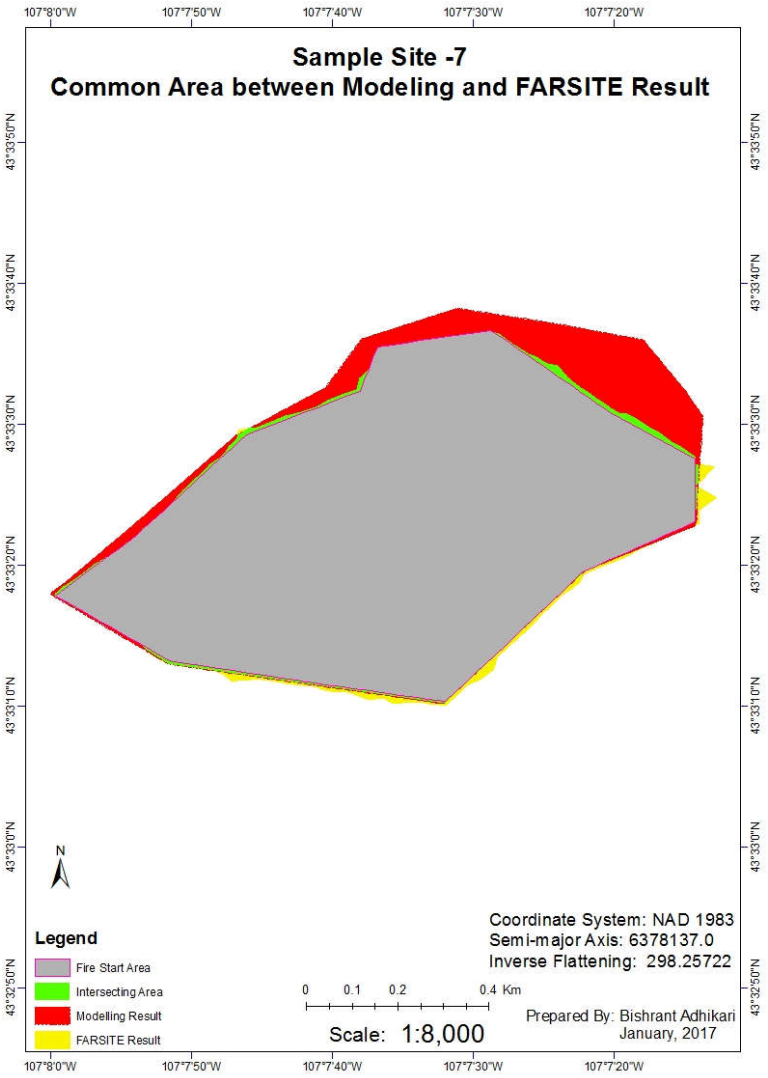
SN	Parameter	Value
1	Dominant Vegetation Fuel Model	TL3
2	Initial Burn area	0.476 Sq. Km
3	Average Wind Speed	27 mph
4	Wind Direction	40 degrees
5	Average Fuel Moisture (1h, 10h, 100hr)	18%, 25%, 27%
6	Humidity Min, Max	40% - 48%

7	Temperature Min, Max (o Celsius)	26 - 32
8	Simulation iteration Time	1 Hour
9	Simulation Steps	5
10	Total Simulation Time	5 Hours

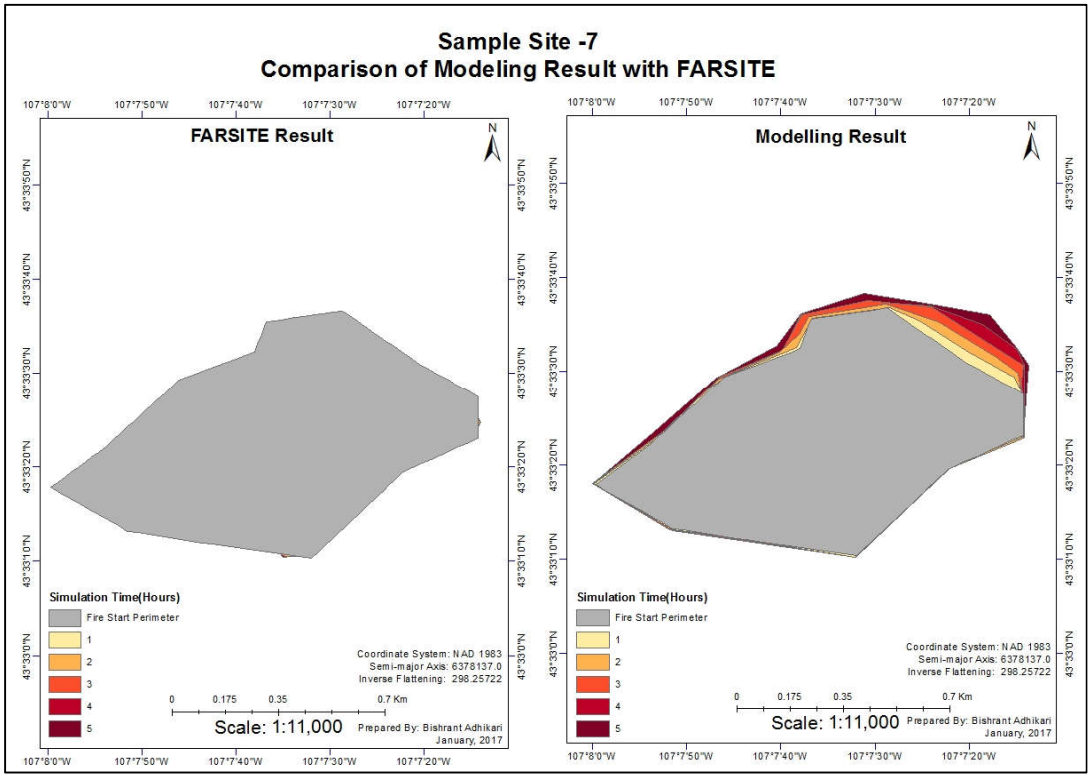
Table 5- 14 Sample site-7 summary and modelling conditions

5.3.7.1. Simulation results

The results from modelling and its comparison with results from FARSITE are shown below:



Map 5- 14 Intersecting area between FARSITE and model results, Site-7



Map 5- 15 Comparison of Modelling result with FARSITE result: Site-7

Following were the results from this simulation run:

Site 7	Area (Square Km)
Total area predicted by Model	0.07
Total area predicted by FARSITE	0.025
Intersecting area in both	0.01

Table 5- 15 Sample site-7 comparative results

The results as visualized in WebGIS interface is shown below:

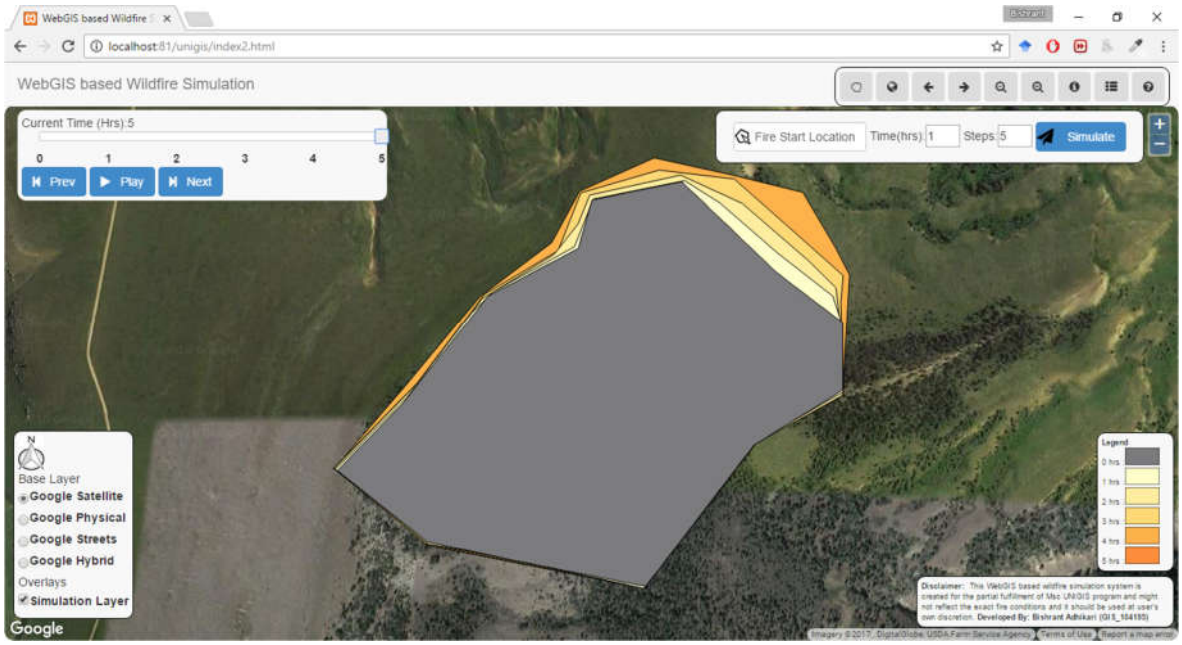


Figure 5- 7 Modelling result in WebGIS interface: Site-7

5.3.7.2. Observation:

In site 7, FARSITE predicted that the fire would propagate forward very slightly. Although my model did not predict too much fire spread, it is slightly greater than the prediction done by FARSITE.

5.3.8. Sample Site-8

Summary of Sample site and modelling conditions:

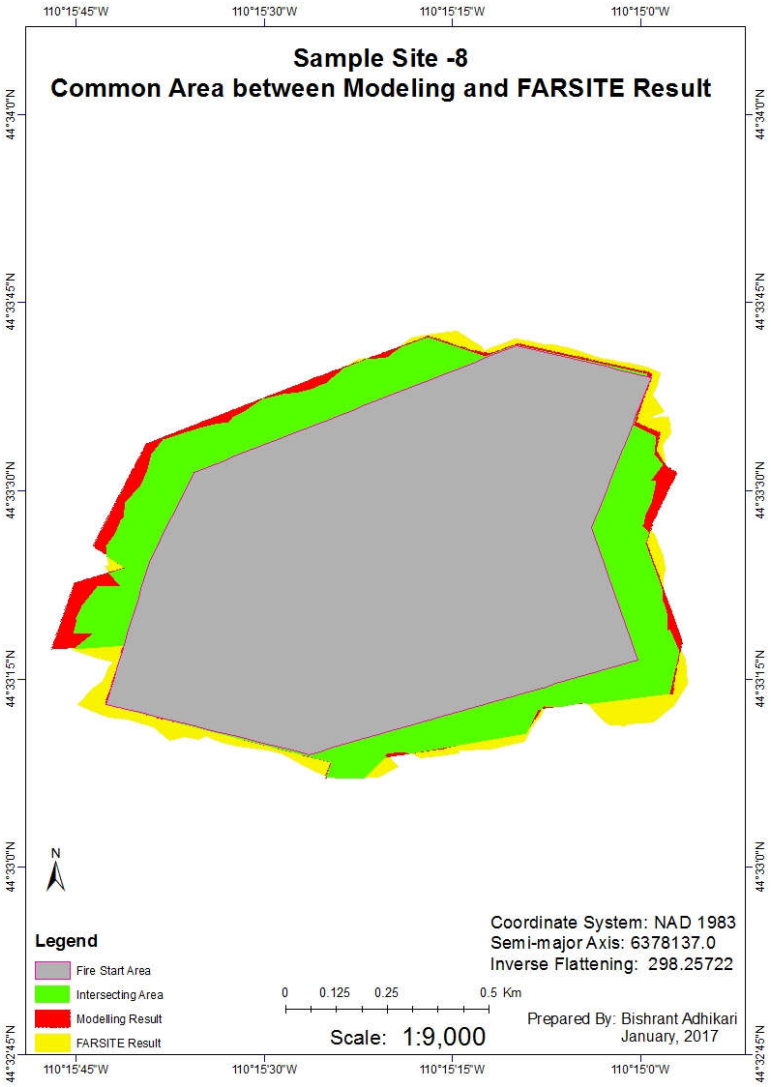
SN	Parameter	Value
1	Dominant Vegetation Fuel Model	SB1
2	Initial Burn area	0.645 Sq. Km
3	Average Wind Speed	No Wind
4	Wind Direction	No Wind
5	Average Fuel Moisture (1h, 10h, 100hr)	2%, 6%, 13%
6	Humidity Min, Max	5% - 22%
7	Temperature Min, Max (o Celsius)	21 - 36
8	Simulation iteration Time	2 Hours

9	Simulation Steps	5
10	Total Simulation Time	10 Hours

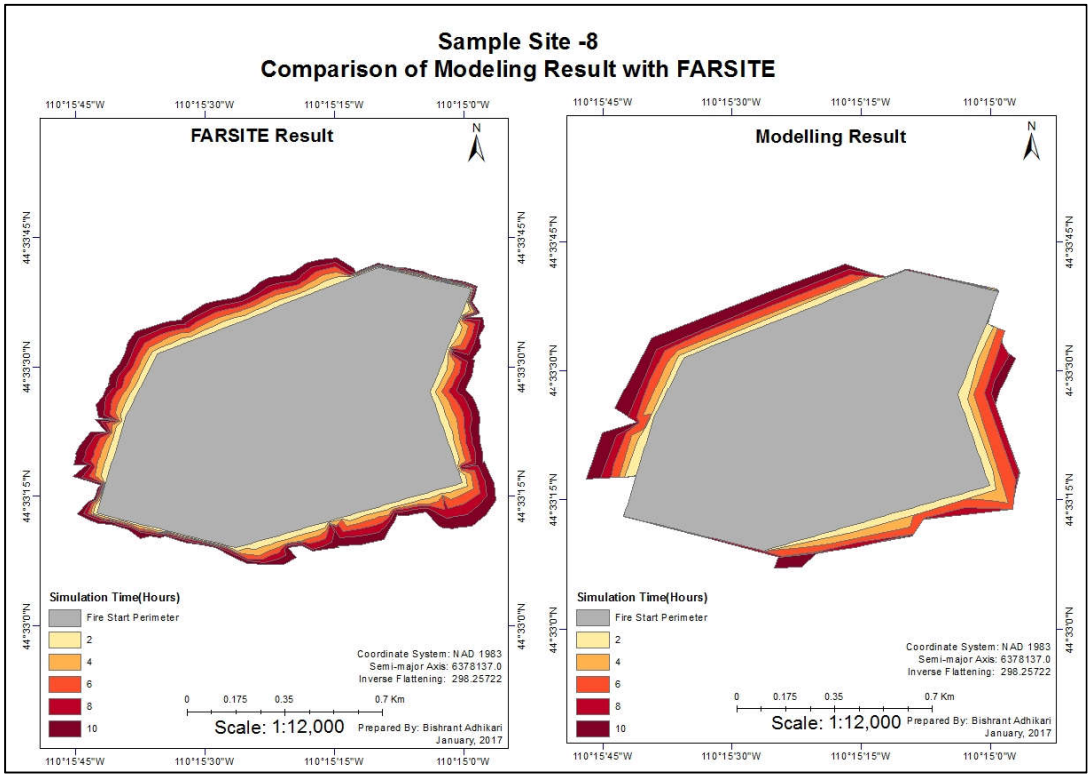
Table 5- 16 Sample site-8 summary and modelling conditions

5.3.8.1. Simulation Results

The results from modelling and its comparison with results from FARSITE are shown below:



Map 5- 16 Intersecting area between FARSITE and model results, Site-8



Map 5- 17 Comparison of Modelling result with FARSITE result: Site-8

Following were the results from this simulation run:

Site 8	Area (Square Km)
Total area predicted by Model	0.205
Total area predicted by FARSITE	0.230
Intersecting area in both	0.185

Table 5- 17 Sample site-8 comparative results

The results as visualized in WebGIS interface is shown below:

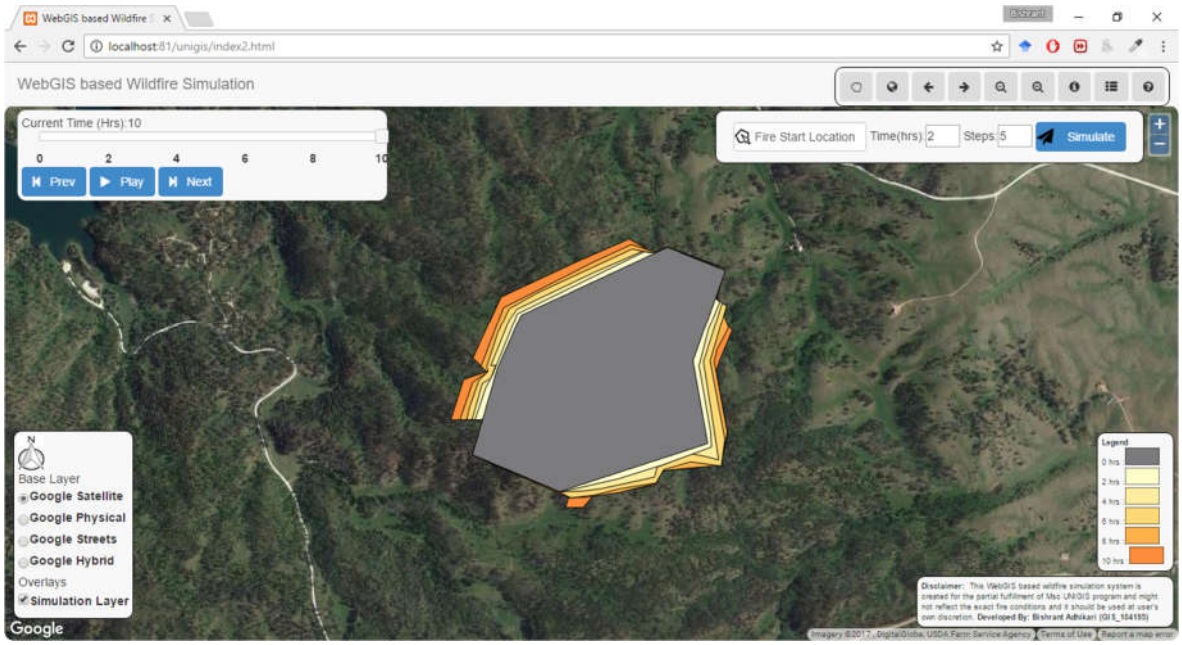


Figure 5- 8 WebGIS simulation result for Site-8

5.3.8.2. Observation:

Site 8 environmental conditions were chosen to simulate no wind conditions. During the weather with no wind blowing, the fire is supposed to spread in all direction. However, due to fuel conditions, the fire did not achieve an overall circular shape, but only two sides were propagating while other two were relatively contained. FARSITE predicted fire spread in all direction despite considerably low in North-east and South-west directions. My model, however, predicted that the fire would not spread at all in those directions because of lack of fuel and fire is mostly spreading in South-East and North-West.

5.3.9. Summary of observation from sample sites

From the observation recorded in eight different sample sites, a summary of the results from model and FARSITE is tabulated below:

Site ID	Fire Start Area (Sq. m)	Area predicted by Model (Sq. Km)	Area predicted by FARSITE (Sq. Km)	Common area in both	Difference with FARSITE (%)
1	0.07	0.2	0.13	0.129	53.86
2	0.97	0.08	0.02	0.02	300
3	0.41	2.93	3.50	2	-22.62
4	0.021	0.065	0.048	0.047	35.41
5	0.19	0	0	No Fire	Same
6	0.19	0.11	0	0	N/A(*)
7	0.476	0.07	0.025	0.01	180
8	0.645	0.205	0.230	0.185	-10.87

Table 5- 18 Summary of modelling results from sample sites

*: Fire spread predicted by model as opposed to no fire spread by FARSITE

Negative sign(-) refers to under prediction by model and Positive Difference refers to over prediction by the model

5.4. Visualisation of results with time slider

The results obtained from the models were presented to the users as a time series animation. The users could view the results of their query in real time in the form of an interactive map. The users could perceive the spread of fire using the time slider provided to them. This information is accompanied by legend and choice of changing the base layer to understand fire behaviour better. The time series result for Sample site -3 of the earlier section is given in Appendix-D with images for each time stamp from the beginning of the simulation to the end of it.

5.5. Discussion

From the eight sample sites chosen for comparing the results of the model, it has been observed that in general, this model is over predicting the wildfire spread at most of the cases. In some cases, the prediction is severely out of prediction made by FARSITE. At two sites (site-3 and site -8) it has been observed that the model under-predicted the fire perimeter. However, for the sample site 5, the result from both FARSITE and my model were consistent as both predicted no fire spread. Site-2 comparison result showed model

prediction were 300% more than the FARSITE result but on the actual ground area its only 0.06 sq.km. The higher percentage is because of smaller area predicted by FARSITE. Overall, the prediction of the model could be regarded as reasonable for the initial model development and difference of the nature of calculations being used. The simplification brought on to the developed model could account for this difference. These discrepancies could be attributed to lack of comprehensive fire modelling techniques in my model. As crown fire model and spotting are not included in my model, they could also account for this difference in prediction. Also, FARSITE uses its internal algorithm to calculate live fuel moisture whereas my model uses the fuel moisture value calculated using MODIS image. Also, the limitation of computation resources has forced me to reduce the number of points per unit length of fire front to be considered for fire propagation. Currently, the point density for fire front is only 5 per meters which could also bring unexpected fire behaviour. Despite all these, the shape attained by the fire perimeter is almost similar in both conditions which show the closer agreement of the modelling results with FARSITE. The problem observed here is mostly related to scale which might be reduced considerably by including Van Wagner's and Albini's model for incorporating crown fire model and spotting phenomenon into the current model. Those works are planned for future improvement of the model and were left in this study because of lack of time and computation resources.

Chapter 6: Conclusion and Recommendation

6.1. Conclusion

This chapter presents a summarized conclusion of the findings and observations made during and after the completion of this research. Upon building a WebGIS based wildfire modeling system it has successfully achieved the primary and secondary objectives stated in Chapter 1. The sub-objectives mentioned in Chapter 1 which collectively helped to achieve the primary objective are discussed below.

6.1.1. To develop wildfire behavior model

A wildfire behavior model was developed successfully using Rothermel (1972) equation for modelling surface fires. The modifications put forth by Albini (1982) were also used to improvise Rothermel's model. The model was successful in calculating several wildfire behavior variables such as Rate of Spread and wildfire intensity. These inputs could be used for calculating burning index that determines the risk of burning for any specific location.

6.1.2. To develop wildfire spread model

A GIS based wildfire spread model was developed using the spatial inputs such as topography, wind, weather, fuel moisture along with its rate of spread and fire intensity. It has been observed that the fire spread direction is primarily guided by the wind direction. The amount of backfire is also calculated dynamically thus instead of propagating fires equally in all directions or only in the direction of wind, the fires propagate in both directions but with varying magnitude. The utilization of custom developed python scripts were successful in simulating wildfire spread without the need of costly desktop GIS software.

6.1.3. Verification and comparison of results with standardized model results

The combined results from wildfire model were compared with the results obtained from FARSITE. Upon comparison, it has been found that the results attained a varying success rates. At most of the cases the model prediction were greater than the spread predicted by FARSITE. However, the general shape of the wildfire spread was almost similar in both cases. The omission of spotting and crown fire model is considered responsible for the mismatch in modelling results. The inclusion of both of those models could bring the modelling results into closer to the FARSITE results. It has been observed that the fire behavior in both cases is same as both of them do not spread fire in non-burnable fuels such as snow or bare land.

6.1.4. Visualization of the modelling and simulation results

The simulation results are presented to the users in the form of WebGIS based portal that facilitates time slider for animating through the modelling results. The responsive user interface coupled with interactive map interface provides the users the ability to choose the fire start location and time along with steps to run the simulation. The results are updated in real time using the updated data. The users are also provided with several controls such as pan, zoom, identify among other for interacting with map.

6.1.5. Discussion

Upon completion of this research, it has been practically proved that the complex wildfire spread models and their equations could be implemented in WebGIS environment to provide real-time wildfire modelling capabilities to the users. This system seemingly simple from the front-end hides the complexity of modelling dynamic environmental phenomenon such as wildfire using custom develop scripts and tools. The WebGIS based wildfire modelling is supposed to make a substantial impact on improving the efficiency of wildfire management activities. Through its easier to user interface, this system allows even the user unware of core fire behavior knowledge to study the future spread behavior of

wildfire. Personal users as well as insurance companies could evaluate the risk perceived by an specific area and the properties there from a fire event at a certain area. Fire fighters could readily use this application to plan their fire management strategies.

The utilization of Geocomputation and custom programming using open source libraries and packages has allowed us to quantify fire spread in the form of rate of spread, flame length and fire intensity in near real time. The automation of data download, processing and utilization of forecasted data has increased the usability of the model for studying fire behavior in near future. The automated re-initialization of the model as soon as the updated input data is received allows the model to better portray the current environment which greatly influences where and how fast a wildfire is going to spread. The integration of server side computing with rich client based interface has made it possible for the system to run on mobile devices and perform with greater efficiency than client based applications. Thus, this architecture has been found successful in handling the temporal dimension of dynamic environmental process such as wildfire.

6.2. Problems encountered and Limitations

Beyond the problems expected and discussed in Chapter 1, some unforeseen problems were encountered during the completion of this research. The major problem was the computation resources that were available for performing the simulation. For making a better estimate of wildfire spread behavior smaller computation intervals and dense points were required. However due to limited computation resources, the point density in ellipse had to be fixed at 50 per ellipse which might not truly reflect real world phenomenon. Also minimum time step of five minutes had to be chosen to reduce the need of very intensive computation. In addition to this, the fire perimeter needed to be simplified thus removing minor peaks and crests for the sake of computation efficiency.

The equations developed by Rothermel were developed in English units so all the inputs needed to be converted to English units first and the results needed to be converted to metric units during final output. It also added extra overhead to the computation especially

for larger rasters. Also because of the limitation of availability of data, the raster resolution had to be kept at 30 meters for DEM and Landscape files. This resolution often resulted in generalization of smaller roads and small water bodies thus allowing fire to spread forward disregarding the fire break from roads and water bodies. The resolution of weather data such as wind and humidity were very coarser than other inputs and thus needed to be resampled to 30 meters without increasing the actual accuracy of the data. Thus, the variations in wildfire behavior brought about by local wind conditions could not be fully investigated.

Finally, the results had to be compared with the output from FARSITE under similar conditions because of lack of time series based fire perimeter and fire ignition perimeter data. Assumption has been made that FARSITE produces fairly accurate results under homogenous fuel conditions. Also, the fire is assumed to behave only as surface fire, its crown fire and spotting nature is not considered for this research because of limitation of time and resources.

6.3. Recommendations and Future improvements

After completing this research despite several limitations and problems, following are the recommendations and future improvement deemed appropriate:

6.3.1. Utilization of better resolution input data

The input data used for this project had the highest resolution of 30 meters which is not sufficient for properly approximating wildfire spread. Specially, the weather data had a resolution of around 5 kilometers which neglected small scale variation caused from wind and weather. Better resolution weather data along with satellite images could allow the model to perform better. Also, if the real-time weather data feeds are available, they could also greatly improve the real-time nature of the model and its results.

6.3.2. Incorporation of Crown and Spotting fire models

An assumption has been made in this research that fire will behave as surface fire which might not be the case in real world. The incorporation of crown fire model will allow the fire to propagate along tree tops whereas spotting fire models would allow the fires to start a new fire with the conflagrations and flakes originating from existing fire perimeter. During the high winds, the spotting behavior is very common thus bringing new dimension in fire behavior and fire spread. If the spotting and crown fire model are implemented, it could improve the predictive ability of wildfire by a larger margin.

6.3.3. Testing with real wildfire data

The results of this model have been tested with the findings from FARSITE however the real accuracy of the output could be ascertained using the real wildfire perimeter data. The real wildfire data should be of good temporal resolution at least equal to the simulation time step being used for modelling.

6.3.4. Utilization of parallel computing

The model has been tested in a portable laptop with limited computation resources. However, the real potential of the model could be tested and improvements could be made in resources consumption if a better computing platform is used. Parallel processing could be used to allocate different tasks to different memory locations for improvement of performance.

References

Alesheikh, A. A., Helali, H., & Behroz, H. A. (2002). Web GIS: technologies and its applications. In Symposium on geospatial theory, processing and applications (Vol. 15).

Albini, F. A. (1976). Estimating wildfire behavior and effects.

Albright, D., & Meisner, B. N. (1999). Classification of fire simulation systems. Fire management notes.

Anderson, D. H., Catchpole, E. A., De Mestre, N. J., & Parkes, T. (1982). Modelling the spread of grass fires. The Journal of the Australian Mathematical Society. Series B. Applied Mathematics, 23(04), 451-466.

Andrews, P. L. (1986). BEHAVE: fire behavior prediction and fuel modeling system-BURN subsystem, Part 1.Retrieved from http://digitalcommons.usu.edu/barkbeetles/151/, retrieved on: 01-19-2017

Andrews, P. (2010). Do you BEHAVE?-Application of the BehavePlus fire modeling system. In Proceedings of 3rd fire behavior and fuels conference, Birmingham, AL: International Association of Wildland Fire. CD-ROM. Retrieved from http://www.fs.fed.us/rm/pubs_other/rmrs_2010_andrews_p001.pdf, retrieved on: 01-19-2017

Altintas I., Block J., de Callafon R., Crawl D., Cowart C., Gupta A., Nguyen M., Braun H.W., Schulze J., Gollner M., Trouve A., Smarr L. (2015). Towards an integrated cyberinfrastructure for scalable data-driven monitoring, dynamic prediction and resilience of wildfires. Procedia Computer Science, 51, 1633-1642. In Proceedings of the Workshop on Dynamic Data-Driven Application Systems (DDDAS) at the 15th International Conference on Computational Science (ICCS 2015). doi:10.1016/j.procs.2015.05.296.

Anderson, D. H., Catchpole, E. A., De Mestre, N. J., & Parkes, T. (1982). Modelling the spread of grass fires. The Journal of the Australian Mathematical Society. Series B. Applied Mathematics, 23(04), 451-466.

Andrews, P. L. (1986). BEHAVE: fire behavior prediction and fuel modeling system-BURN subsystem, Part 1. Retrieved from http://digitalcommons.usu.edu/barkbeetles/151/, retrieved on: 01-19-2017

Beer, T. (1990). Percolation theory and fire spread. Combustion science and technology, 72(4-6), 297-304.

Brimicombe, A. (2009). GIS, environmental modeling and engineering. CRC Press.

Burgan, R. E., & Rothermel, R. C. (1984). BEHAVE: fire behavior prediction and fuel modeling system--FUEL subsystem.

Burgan, R. E., & Hartford, R. A. (1997). Live vegetation moisture calculated from NDVI and used in fire danger rating. In Proceedings of the 13th Fire and Forest Meteorology Conference (pp. 225-231).

Byram, G. M. (1959). Combustion of forest fuels. In 'Forest fire: control and use'. (Ed. KP Davis) pp. 61–89.

Câmara, G., & Monteiro, A. M. V. (2001). Geocomputation techniques for spatial analysis: are they relevant to health data? Cadernos de Saúde Pública, 17(5), 1059-1071.

Carmo, M., Moreira, F., Casimiro, P., & Vaz, P. (2011). Land use and topography influences on wildfire occurrence in northern Portugal. Landscape and Urban Planning, 100(1), 169-176.

Ceccato, P., Flasse, S., Tarantola, S., Jacquemoud, S., & Grégoire, J. M. (2001). Detecting vegetation leaf water content using reflectance in the optical domain. Remote sensing of environment, 77(1), 22-33.

Cheney, N. P., Gould, J. S., & Catchpole, W. R. (1998). Prediction of fire spread in grasslands. International Journal of Wildland Fire, 8(1), 1-13.

Cheney, P., & Sullivan, A. (Eds.). (2008). Grassfires: fuel, weather and fire behaviour. CSIRO PUBLISHING.

Chuvieco, E. (2003). Wildland fire danger: estimation and mapping: the role of remote sensing data (Vol. 4). World Scientific.

Clark, T. L., Coen, J. L., Radke, L., Reeder, M., & Packham, D. (1998). Coupled atmosphere fire dynamics. In Proc. III Int. Conference on Forest Fire Research/14th Conference on Fire and Forest Meteorology (Vol. 1). 67-82

Clarke, K.C., Parks, B.O. and Crane, M.P. (2002). Geographic Information Systems and Environmental Modeling, 1st edition, New Jersey: Prentice Hall

Dennison, P. E., Roberts, D. A., Peterson, S. H., & Rechel, J. (2005). Use of normalized difference water index for monitoring live fuel moisture. International Journal of Remote Sensing, 26(5), 1035-1042.

Dragicevic, S. (2004). The potential of Web-based GIS. Journal of Geographical Systems, 6(2), 79-81.

Favier, C. (2004). Percolation model of fire dynamic. Physics Letters A, 330(5), 396-401.

Finney, M. A., & Ryan, K. C. (1995). Use of the FARSITE fire growth model for fire prediction in US National Parks. In International emergency management and engineering conference, edited by JD Sullivan, JL Wybo, and L. Buisson. Paris, France: International Emergency Management and Engineering Society

Finney, M. A., & Andrews, P. L. (1998). Application and status of the FARSITE fire area simulator.

Finney, M. A., & Andrews, P. L. (1999). FARSITE—a program for fire growth simulation. Fire management notes, 59(2), 13-15.

Finney, M. A. (2003). Calculation of fire spread rates across random landscapes. International Journal of Wildland Fire, 12(2), 167-174.

Finney, M. A. (2004). FARSITE: Fire area simulator: model development and evaluation. Ogden, UT: US Department of Agriculture, Forest Service, Rocky Mountain Research Station.

Finney, M. A., Brittain, S., & Seli, R. C. (2004). FlamMap Spatial Analysis of Fire Potential. Missoula, MT: Rocky Mountain Research Station, USDA Forest Service.

Finney, M. A. (2006). An overview of FlamMap fire modeling capabilities. Retrieved from http://www.treesearch.fs.fed.us/pubs/25948, retrieved on: 01-19-2017

French, I. A., Anderson, D. H., & Catchpole, E. A. (1990). Graphical simulation of bushfire spread. Mathematical and Computer Modelling, 13(12), 67-71.

Global Fire Monitoring Center (GFMC). (2016). Fire Management Glossaries, International Strategy for Disaster Reduction, Retrieved from: http://www.fire.uni-freiburg.de/literature/show_complete.php, retrieved on: 01-19-2017

Glasa, J., & Halada, L. (2008). On elliptical model for forest fire spread modeling and simulation. Mathematics and Computers in Simulation, 78(1), 76-88.

Google. (2014, November 24). The Final Countdown for NPAPI. Retrieved from https://blog.chromium.org/2014/11/the-final-countdown-for-npapi.html, retrieved on: 01-19-2017

Hao, X., & Qu, J. J. (2005). Real-Time Live Fuel Moisture Retrieval with MODIS Measurements. In submitted to East FIRE Conference Proceeding.

Hargrove, W. W., Gardner, R. H., Turner, M. G., Romme, W. H., & Despain, D. G. (2000). Simulating fire patterns in heterogeneous landscapes. Ecological modelling, 135(2), 243-263.

Hungerford, R. D., Harrington, M. G., Frandsen, W. H., Ryan, K. C., & Niehoff, G. J. (1991). Influence of fire on factors that affect site productivity. In Proceedings of the symposium on management and productivity of western-montane forest soils (pp. 32-50).

Johnston, P., Milne, G., & Kelso, J. (2006). A heat transfer simulation model for wildfire spread. Forest Ecology and Management, 234(1), S78.

Kalabokidis, K., Athanasis, N., Gagliardi, F., Karayiannis, F., Palaiologou, P., Parastatidis, S., & Vasilakos, C. (2013). Virtual Fire: A web-based GIS platform for forest fire control. Ecological Informatics, 16, 62-69.

Kennard, D. and Fowler, C. (2008). Fire Behaviour. [Online]. Available: http://www.forestencyclopedia.net, retrieved on: 01-19-2017

LANDFIRE. (2015). LANDFIRE 2015 Program Report

Mansor, S., Abu Shariah, M., Billa, L., Setiawan, I., & Jabar, F. (2004). Spatial technology for natural risk management. Disaster Prevention and Management: An International Journal, 13(5), 364-373.

Martínez, J., Vega-Garcia, C., & Chuvieco, E. (2009). Human-caused wildfire risk rating for prevention planning in Spain. Journal of Environmental Management, 90(2), 1241-1252.

Martin, J., & Hillen, T. (2016). The Spotting Distribution of Wildfires. Applied Sciences, 6(6), 177.

Mathiyalagan, V., Grunwald, S., Reddy, K. R., & Bloom, S. A. (2005). A WebGIS and geodatabase for Florida's wetlands. Computers and electronics in agriculture, 47(1), 69-75.

McArthur, A. G. (1966). Weather and grassland fire behaviour. Forestry and Timber Bureau, Department of national Development, Commonwealth of Australia.

McCarthy MA. (1997). Fire modeling and biodiversity. Proceedings of the Conference of the Ecological Society of Australia, Albury, Australia: Charles Sturt University

McCormick, R. J. (2002). On Developing a Meso-theoretical Viewpoint of Complex Systems by Exploring the Use of Artificial Neural Networks in Modeling Wildfires. In ForestSAT Symposium, Edinburgh.

McGrattan, K., Hostikka, S., & Floyd, J. E. (2010). Fire dynamics simulator (version 5), user's guide. NIST special publication, 1019(5), 1-186.

Microsoft. (2010, December 21). Virtual Fire System Aids Firefighters in Wildfire Combat and Prevention. Retrieved from https://www.microsoft.com/en-

us/research/blog/virtual-fire-system-aids-firefighters-in-wildfire-combat-and-prevention/, retrieved on: 01-19-2017

National Park Service (NPS). (2016). Wildfire causes, Retrieved from: https://www.nps.gov/fire/wildland-fire/learning-center/fire-in-depth/wildfire-causes.cfm, retrieved on: 01-19-2017

National Wildfire Coordinating Group (NWCG). (2006). NWCG Fireline Handbook, Appendix B, pp B25-B28. Retrieved from: https://www.nwcg.gov/sites/default/files/products/appendixB.pdf, retrieved on: 01-19-2017

Nelson Jr, R. M. (2000). Prediction of diurnal change in 10-h fuel stick moisture content. Canadian Journal of Forest Research, 30(7), 1071-1087.

Noonan, E. K., & Tueller, P. T. (2008). The Application of FARSITE for Assessing a Mechanical Fuel Treatment in the Eastern Sierra Nevada1. Managing Fire and Fuels in the Remaining Wildlands and Open Spaces of the Southwestern United States, 295.

Ntaimo, L., Hu, X., & Sun, Y. (2008). DEVS-FIRE: Towards an integrated simulation environment for surface wildfire spread and containment. Simulation, 84(4), 137-155.

NWCG. (2008). Glossary of Wildland Fire Terminology, National Wildfire Coordinating Group, pp 182. Retrieved from: https://gacc.nifc.gov/nrcc/dc/idgvc/dispatchforms/glossary.pdf, retrieved on: 01-19-2017

Omi, P. N., & Martinson, E. J. (2002). Effect of fuels treatment on wildfire severity. Final report submitted to the Joint Fire Science Program Governing Board.

Pastor, E., Zarate, L., Planas, E., & Arnaldos, J. (2003). Mathematical models and calculation systems for the study of wildland fire behaviour. Progress in Energy and Combustion Science, 29(2), 139-153.

Peng, Z. R. (1999). An assessment framework for the development of Internet GIS. Environment and Planning B: Planning and Design, 26(1), 117-132.

Perry, G. L. W. (1998). Current approaches to modelling the spread of wildland fire: a review. Progress in Physical Geography, 22(2), 222-245.

Plewe, B. (1997). GIS online: Information retrieval, mapping, and the Internet. OnWord Press.

Rios, O., Jahn, W., & Rein, G. (2014). Forecasting wind-driven wildfires using an inverse modelling approach. Natural Hazards and Earth System Sciences, 14(6), 1491.

Roberts, D. A., Dennison, P. E., Peterson, S. H., & Rechel, J. (2006). Use of Normalized Difference Water Index for monitoring live fuel moisture.

Rothermel, R. C. (1972). A mathematical model for predicting fire spread in wildland fuels.

Rothermel, R. C. (1983). How to predict the spread and intensity of forest and range fires. The Bark Beetles, Fuels, and Fire Bibliography, 70.

Ryu, S. R., Chen, J., Zheng, D., & Lacroix, J. J. (2007). Relating surface fire spread to landscape structure: an application of FARSITE in a managed forest landscape. Landscape and Urban Planning, 83(4), 275-283.

Scott, J. H. (1999). NEXUS: a system for assessing crown fire hazard. Fire Management Notes.

Skidmore, A. (2002). Environmental Modelling with GIS and Remote Sensing: Introduction. Environmental Modelling with GIS and Remote Sensing. London: Taylor and Francis.

Sohlberg, R., Descloitres, J., & Bobbe, T. (2001). MODIS Land rapid response: operational use of terra data for USFS wildfire management. The Earth Observer, 13(5), 8-10.

Sneeuwjagt, R. J., Peet, G. B., & Beggs, B. J. (1979). Forest fire behaviour tables for Western Australia. Forests Department, Western Australia.

Sullivan, A. L. (2009a). Wildland surface fire spread modelling, 1990–2007. 2: Empirical and quasi-empirical models. International Journal of Wildland Fire, 18(4), 369–386.

Sullivan, A. L. (2009b). Wildland surface fire spread modelling, 1990–2007. 3: Simulation and mathematical analogue models. International Journal of Wildland Fire, 18(4), 387–403. https://doi.org/10.1071/WF06144

Taylor, S. W., & Alexander, M. E. (2006). Science, technology, and human factors in fire danger rating: the Canadian experience. International Journal of Wildland Fire, 15(1), 121-135.

Trollope, W. S. W., Trollope, L. A., & Hartnett, D. C. (2002). Fire behaviour a key factor in the fire ecology of African grasslands and savannas. Forest Fire Research and Wildland Fire Safety, Millpress, Rotterdam.

Vakalis, D., Sarimveis, H., Kiranoudis, C. T., Alexandridis, A., & Bafas, G. (2004). A GIS based operational system for wildland fire crisis management II. System architecture and case studies. Applied Mathematical Modelling, 28(4), 411-425.

Wagner, C. V. (1977). Conditions for the start and spread of crown fire. Canadian Journal of Forest Research, 7(1), 23-34.

Wildland Fire Management (WFM). (2012). Introduction to Fire Behavior Modeling, retrieved from: https://www.frames.gov/files/8413/4643/5159/Intro_to_Fire_Behavior_Modeling_Guide_2012.06.25.pdf, retrieved on: 01-19-2017

Yassemi, S., Dragićević, S., & Schmidt, M. (2008). Design and implementation of an integrated GIS-based cellular automata model to characterize forest fire behaviour. Ecological modelling, 210(1), 71-84.

Zheng, K. G., Rahim, S. T., & Pan, Y. H. (2000). Web GIS: Implementation issues. Chinese Geographical Science, 10(1), 74-79.

Appendix

Appendix A-1

National Fire Danger Rating System, Parameters for Fuel models

Fuel model	Ratios and Fuel loadings[1] 1-h[2]	10-h[3]	100-h[4]	1000-h[5]	Wood[6]	Herb[7]	Depth (ft)[8]	MXD (pct)[9]	HD&HL (Btu/lb)[10]	SCM[11]	WNDFC[12]
A Western grasses (annual)	3000 0.20	---- ----	---- ----	---- ----	---- ----	3000 0.30	0.80	15	8000	300	0.6
B California chaparral	700 3.50	109 4.00	30 0.50	---- ----	1250 11.50	---- ----	4.50	15	9500	58	0.5
C Pine-grass savanna	2000 0.40	109 1.00	---- ----	---- ----	1500 0.50	2500 0.80	0.75	20	8000	32	0.4
D Southern rough	1250 2.00	109 1.00	---- ----	---- ----	1500 3.00	1500 0.75	2.00	30	9000	25	0.4
E Hardwood litter (winter)	2000 1.50	109 0.50	30 0.25	---- ----	1500 0.50	2000 0.50	0.40	25	8000	25	0.4
F Intermediate brush	700 2.50	109 2.00	30 1.50	---- ----	1250 9.00	---- ----	4.50	15	9500	24	0.5
G Short needle (heavy dead)	2000 2.50	109 2.00	30 5.00	8 12.0	1500 0.50	2000 0.50	1.00	25	8000	30	0.4
H Short needle (normal dead)	2000 1.50	109 1.00	30 2.00	8 2.00	1500 0.50	2000 0.50	0.30	20	8000	8	0.4
I Heavy slash	1500 12.00	109 12.00	30 10.00	8 12.00	---- ----	---- ----	2.00	25	8000	65	0.5
J Intermediate slash	1500 7.00	109 7.00	30 6.00	8 5.50	---- ----	---- ----	1.30	25	8000	44	0.5
K Light slash	1500 2.50	109 2.50	30 2.00	8 2.50	---- ----	---- ----	0.60	25	8000	23	0.5
L Western grasses (perennial)	2000 0.25	109 1.50	---- ----	---- ----	---- ----	2000 0.50	1.00	15	8000	178	0.6
N Sawgrass	1600 1.50	109 3.00	---- ----	---- ----	1500 2.00	2000 0.50	3.00	25	8700	167	0.6
O High pocosin	1500 2.00	109 1.00	30 3.00	8 2.00	1500 7.00	---- ----	4.00	30	9000	99	0.5
P Southern pine plantation	1750 1.00	109 2.50	30 0.50	---- ----	1500 0.50	---- ----	0.40	30	8000	14	0.4
Q Alaskan black spruce	1500 2.00	109 0.50	30 2.00	8 1.00	1200 4.00	1500 0.50	3.00	25	8000	59	0.4
R Hardwood litter (summer)	1500 0.50	109 0.50	30 0.50	---- ----	1500 0.50	2000 0.50	0.25	25	8000	6	0.4
S Tundra	1500 0.50	109 0.50	30 0.50	8 0.50	1200 0.50	1500 0.50	0.40	25	8000	17	0.6
T Sagebrush-grass	2500 1.00	109 1.50	---- ----	---- ----	1500 2.50	2000 0.50	1.25	15	8000	73	0.6
U Western pines	1750 1.50		30 1.00	---- ----	1500 0.50	2000 0.50	0.50	20	8000	16	0.4

[1]For each fuel model, the top value is surface-area-to-volume ratio (ft^{-1}), and the bottom value is fuel loading (tons/acre).
[2]1-hour timelag dead fuel moisture class.
[3]10-hour timelag dead fuel moisture class.
[4]100-hour timelag dead fuel moisture class.
[5]1000-hour timelag dead fuel moisture class.
[6]Live fine woody fuel class.
[7]Live fine herbaceous fuel class.
[8]Effective fuel bed depth.
[9]Assigned dead fuel moisture of extinction.
[10]Dead and live fuel heat of combustion.
[11]Assigned spread component value when all ignitions become reportable fires.
[12]Wind reduction factor from 20-foot standard height to the midflame height.

Source: Cohen, J. D., & Deeming, J. E. (1985). The national fire-danger rating system: basic equations

Appendix A-2

Albini's 13 Fuel Model Parameters

Model	Typical fuel complexes	Surface-to-volume ratio (ft^{-1})/Loading (lb/ft^2): Dead fuel: 1-h	Dead fuel: 10-h	Dead fuel: 100-h	Live fuel (Foliage)	Depth (ft)	Moisture of extinction, dead fuel (percent)
		GRASS AND GRASS-DOMINATED					
1	Short grass (1 ft)	3500/.034	--	--	--	1.0	12
2	Timber (grass and understory)	3000/.092	109/.046	30/.023	1500/.023	1.0	15
3	Tall grass (2.5 ft)	1500/.138	--	--	--	2.5	25
		CHAPARRAL AND SHRUBFIELDS					
4	Chaparral (6 ft)	2000/.230	109/.184	30/.092	1500/.230	6.0	20
5	Brush (2 ft)	2000/.046	1091.023	--	1500/.092	2.0	20
6	Dormant brush, hardwood slash	1750/.069	109/.115	30/.092	--	2.5	25
7	Southern rough	1750/.052	109/.086	30/.069	1550/.017	2.5	40
		TIMBER LITTER					
8	Closed timber litter	2000/.069	109/.046	30/.115	--	0.2	30
9	Hardwood litter	2500/.134	109/.019	301.007	--	.2	25
10	Timber (litter and understory)	2000/.138	109/.092	30/.230	1500/.092	1.0	25
		LOGGING SLASH					
11	Light logging slash	1500/.069	109/.207	30/.253	--	1.0	15
12	Medium logging slash	1500/.184	109/.644	30/.759	--	2.3	20
13	Heavy logging slash	1500/.322	109/1.058	30/1.288	--	3.0	25

Source: Albini, F. A. (1976). Estimating wildfire behavior and effects.

Appendix A-3

Scott and Burgan's 40 Fuel Model Parameters

Fuel model code	Fuel load (t/ac) 1-hr	10-hr	100-hr	Live herb	Live woody	Fuel model type[a]	SAV ratio (1/ft)[b] Dead 1-hr	Live herb	Live woody	Fuel bed depth (ft)	Dead fuel extinction moisture (percent)	Heat content BTU/lb)[c]
GR1	0.10	0.00	0.00	0.30	0.00	dynamic	2200	2000	9999	0.4	15	8000
GR2	0.10	0.00	0.00	1.00	0.00	dynamic	2000	1800	9999	1.0	15	8000
GR3	0.10	0.40	0.00	1.50	0.00	dynamic	1500	1300	9999	2.0	30	8000
GR4	0.25	0.00	0.00	1.90	0.00	dynamic	2000	1800	9999	2.0	15	8000
GR5	0.40	0.00	0.00	2.50	0.00	dynamic	1800	1600	9999	1.5	40	8000
GR6	0.10	0.00	0.00	3.40	0.00	dynamic	2200	2000	9999	1.5	40	9000
GR7	1.00	0.00	0.00	5.40	0.00	dynamic	2000	1800	9999	3.0	15	8000
GR8	0.50	1.00	0.00	7.30	0.00	dynamic	1500	1300	9999	4.0	30	8000
GR9	1.00	1.00	0.00	9.00	0.00	dynamic	1800	1600	9999	5.0	40	8000
GS1	0.20	0.00	0.00	0.50	0.65	dynamic	2000	1800	1800	0.9	15	8000
GS2	0.50	0.50	0.00	0.60	1.00	dynamic	2000	1800	1800	1.5	15	8000
GS3	0.30	0.25	0.00	1.45	1.25	dynamic	1800	1600	1600	1.8	40	8000
GS4	1.90	0.30	0.10	3.40	7.10	dynamic	1800	1600	1600	2.1	40	8000
SH1	0.25	0.25	0.00	0.15	1.30	dynamic	2000	1800	1600	1.0	15	8000
SH2	1.35	2.40	0.75	0.00	3.85	N/A	2000	9999	1600	1.0	15	8000
SH3	0.45	3.00	0.00	0.00	6.20	N/A	1600	9999	1400	2.4	40	8000
SH4	0.85	1.15	0.20	0.00	2.55	N/A	2000	1800	1600	3.0	30	8000
SH5	3.60	2.10	0.00	0.00	2.90	N/A	750	9999	1600	6.0	15	8000
SH6	2.90	1.45	0.00	0.00	1.40	N/A	750	9999	1600	2.0	30	8000
SH7	3.50	5.30	2.20	0.00	3.40	N/A	750	9999	1600	6.0	15	8000
SH8	2.05	3.40	0.85	0.00	4.35	N/A	750	9999	1600	3.0	40	8000
SH9	4.50	2.45	0.00	1.55	7.00	dynamic	750	1800	1500	4.4	40	8000
TU1	0.20	0.90	1.50	0.20	0.90	dynamic	2000	1800	1600	0.6	20	8000
TU2	0.95	1.80	1.25	0.00	0.20	N/A	2000	9999	1600	1.0	30	8000
TU3	1.10	0.15	0.25	0.65	1.10	dynamic	1800	1600	1400	1.3	30	8000
TU4	4.50	0.00	0.00	0.00	2.00	N/A	2300	9999	2000	0.5	12	8000
TU5	4.00	4.00	3.00	0.00	3.00	N/A	1500	9999	750	1.0	25	8000
TL1	1.00	2.20	3.60	0.00	0.00	N/A	2000	9999	9999	0.2	30	8000
TL2	1.40	2.30	2.20	0.00	0.00	N/A	2000	9999	9999	0.2	25	8000
TL3	0.50	2.20	2.80	0.00	0.00	N/A	2000	9999	9999	0.3	20	8000
TL4	0.50	1.50	4.20	0.00	0.00	N/A	2000	9999	9999	0.4	25	8000
TL5	1.15	2.50	4.40	0.00	0.00	N/A	2000	9999	1600	0.6	25	8000
TL6	2.40	1.20	1.20	0.00	0.00	N/A	2000	9999	9999	0.3	25	8000
TL7	0.30	1.40	8.10	0.00	0.00	N/A	2000	9999	9999	0.4	25	8000
TL8	5.80	1.40	1.10	0.00	0.00	N/A	1800	9999	9999	0.3	35	8000
TL9	6.65	3.30	4.15	0.00	0.00	N/A	1800	9999	1600	0.6	35	8000
SB1	1.50	3.00	11.00	0.00	0.00	N/A	2000	9999	9999	1.0	25	8000
SB2	4.50	4.25	4.00	0.00	0.00	N/A	2000	9999	9999	1.0	25	8000
SB3	5.50	2.75	3.00	0.00	0.00	N/A	2000	9999	9999	1.2	25	8000
SB4	5.25	3.50	5.25	0.00	0.00	N/A	2000	9999	9999	2.7	25	8000

[a] Fuel model type does not apply to fuel models without live herbaceous load.
[b] The value 9999 was assigned in cases where there is no load in a particular fuel class or category
[c] The same heat content value was applied to both live and dead fuel categories.

Source: Scott, J. H., & Burgan, R. E. (2005). Standard fire behavior fuel models: a comprehensive set for use with Rothermel's surface fire spread model. The Bark Beetles, Fuels, and Fire Bibliography, 66.

Appendix A-4

Description of Scott and Burgan's 40 Fuel model classification

Model Code	Description
Nearly pure grass and/or forb type (Grass)	
GR1	Grass is short, patchy, and possibly heavily grazed. Spread rate moderate; flame length low.
GR2	Moderately coarse continuous grass, average depth about 1 foot. Spread rate high; flame length moderate
GR3	Very coarse grass, average depth about 2 feet. Spread rate high; flame length moderate.
GR4	Moderately coarse continuous grass, average depth about 2 feet. Spread rate very high; flame length high
GR5	Dense, coarse grass, average depth about 1 to 2 feet. Spread rate very high; flame length high.
GR6	Dryland grass about 1 to 2 feet tall. Spread rate very high; flame length very high.
GR7	Moderately coarse continuous grass, average depth about 3 feet. Spread rate very high; flame length very high.

GR8	Heavy, coarse, continuous grass 3 to 5 feet tall. Spread rate very high; flame length very high.
GR9	Very heavy, coarse, continuous grass 5 to 8 feet tall. Spread rate extreme; flame length extreme
Mixture of grass and shrub, up to about 50 percent shrub coverage (Grass Shrub)	
GS1	Shrubs are about 1 foot high, low grass load. Spread rate moderate; flame length low.
GS2	Shrubs are 1 to 3 feet high, moderate grass load. Spread rate high; flame length moderate
GS3	Moderate grass/shrub load, average grass/shrub depth less than 2 feet. Spread rate high; flame length moderate.
GS4	Heavy grass/shrub load, depth greater than 2 feet. Spread rate high; flame length very high
Shrubs cover at least 50 percent of the site; grass sparse to nonexistent (Shrub)	
SH1	Low shrub fuel load, fuelbed depth about 1 foot; some grass may be present. Spread rate very low; flame length very low
SH2	Moderate fuel load (higher than SH1), depth about 1 foot, no grass fuel present. Spread rate low; flame length low.
SH3	Moderate shrub load, possibly with pine overstory or herbaceous fuel, fuel bed depth 2 to 3 feet. Spread rate low; flame length low.
SH4	Low to moderate shrub and litter load, possibly with pine overstory, fuel bed depth about 3 feet. Spread rate high; flame length moderate.
SH5	Heavy shrub load, depth 4 to 6 feet. Spread rate very high; flame length very high.
SH6	Dense shrubs, little or no herb fuel, depth about 2 feet. Spread rate high; flame length high.
SH7	Very heavy shrub load, depth 4 to 6 feet. Spread rate lower than SH5, but flame length similar. Spread rate high; flame length very high.
SH8	Dense shrubs, little or no herb fuel, depth about 3 feet. Spread rates high; flame length high
SH9	Dense, finely branched shrubs with significant fine dead fuel, about 4 to 6 feet tall; some herbaceous fuel may be present. Spread rate high, flame length very high.
Dead and down woody fuel (litter) beneath a forest canopy (Timber Litter)	
TL1	Light to moderate load, fuels 1 to 2 inches deep. Spread rate very low; flame length very low.
TL2	Low load, compact. Spread rate very low; flame length very low
TL3	Moderate load conifer litter. Spread rate very low; flame length low
TL4	Moderate load, includes small diameter downed logs. Spread rate low; flame length low
TL5	High load conifer litter; light slash or mortality fuel. Spread rate low; flame length low.
TL6	Moderate load, less compact. Spread rate moderate; flame length low
TL7	Heavy load, includes larger diameter downed logs. Spread rate low; flame length low.
TL8	moderate load and compactness may include small amount of herbaceous load. Spread rate moderate; flame length low.
TL9	Very high load broadleaf litter; heavy needle-drape in otherwise sparse shrub layer. Spread rate moderate; flame length moderate.
Grass or shrubs mixed with litter from forest canopy (Timber-Understory)	
TU1	Fuelbed is low load of grass and/or shrub with litter. Spread rate low; flame length low
TU2	Fuelbed is moderate litter load with shrub component. Spread rate moderate; flame length low.
TU3	Fuelbed is moderate litter load with grass and shrub components. Spread rate high; flame length moderate
TU4	Fuelbed is short conifer trees with grass or moss understory. Spread rate moderate; flame length moderate.
TU5	Fuelbed is high load conifer litter with shrub understory. Spread rate moderate; flame length moderate.
Activity fuel (slash) or debris from wind damage (blowdown) (Slash-Blowdown)	
SB1	Fine fuel load is 10 to 20 tons/acre, weighted toward fuels 1 to 3 inches diameter class, depth is less than 1 foot. Spread rate moderate; flame length low
SB2	Fine fuel load is 7 to 12 tons/acre, evenly distributed across 0 to 0.25, 0.25 to 1, and 1 to 3 inch diameter classes, depth is about 1 foot. Spread rate moderate; flame length moderate
SB3	Fine fuel load is 7 to 12 tons/acre, weighted toward 0 to 0.25 inch diameter class, depth is more than 1 foot. Spread rate high; flame length high.
SB4	Blowdown is total, fuelbed not compacted, foliage still attached. Spread rate very high; flame length very high.
Insufficient wildland fuel to carry wildland fire under any condition (Non burnable)	
NB1	Urban or suburban development; insufficient wildland fuel to carry wildland fire.
NB2	Snow/ice.
NB3	Agricultural field, maintained in non-burnable condition
NB8	Open water.
NB9	Bare ground

Source: Scott, & Burgan (2005)

Appendix A-5

Reference Fuel Moisture Day(0800-1959)

Dry Bulb Temp (°F)	Relative Humidity (Percent)																			
	0 ↓ 4	5 ↓ 9	10 ↓ 14	15 ↓ 19	20 ↓ 24	25 ↓ 29	30 ↓ 34	35 ↓ 39	40 ↓ 44	45 ↓ 49	50 ↓ 54	55 ↓ 59	60 ↓ 64	65 ↓ 69	70 ↓ 74	75 ↓ 79	80 ↓ 84	85 ↓ 89	90 ↓ 94	95 ↓ 99	100
10-29	1	2	2	3	4	5	5	6	7	8	8	8	9	9	10	11	12	12	13	13	14
30-49	1	2	2	3	4	5	5	6	7	7	7	8	9	9	10	10	11	12	13	13	13
50-69	1	2	2	3	4	5	5	6	6	7	7	8	8	9	9	10	11	12	12	12	13
70-89	1	1	2	2	3	4	5	5	6	7	7	8	8	8	9	10	10	11	12	12	13
90-109	1	1	2	2	3	4	4	5	6	7	7	8	8	8	9	10	10	11	12	12	13
109+	1	1	2	2	3	4	4	5	6	7	7	8	8	8	9	10	10	11	12	12	12

Source: NWCG (2006, p25)

Appendix A-6

Dead fuel moisture content corrections for month of May, June and July

Aspect	%Slope	0800>			1000>			1200>			1400>			1600>			1800>		
		B	L	A	B	L	A	B	L	A	B	L	A	B	L	A	B	L	A
UNSHADED – LESS THAN 50% SHADING OF SURFACE FUELS																			
N	0-30	2	3	4	1	1	1	0	0	1	0	0	1	1	1	1	2	3	4
	31+	3	4	4	1	2	2	1	1	2	1	1	2	1	2	2	3	4	4
E	0-30	2	2	3	1	1	1	0	0	1	0	0	1	1	1	2	3	4	4
	31+	1	2	2	0	0	1	0	0	1	1	1	2	2	3	4	4	5	6
S	0-30	2	3	3	1	1	1	0	0	1	0	0	1	1	1	1	2	3	3
	31+	2	3	3	1	1	2	0	1	1	0	1	1	1	1	2	2	3	3
W	0-30	2	3	4	1	1	2	0	0	1	0	0	1	0	1	1	2	3	3
	31+	4	5	6	2	3	4	1	1	2	0	0	1	0	0	1	1	2	2
SHADED – 50% OR MORE SHADING OF SURFACE FUELS																			
N	all	4	5	5	3	4	5	3	3	4	3	3	4	3	4	5	4	5	5
E	all	4	4	5	3	4	5	3	3	4	3	4	4	3	4	5	4	5	6
S	all	4	4	5	3	4	5	3	3	4	3	3	4	3	4	5	4	5	5
W	all	4	5	6	3	4	5	3	3	4	3	3	4	3	4	5	4	4	5

Source: NWCG (2006, p26)

Appendix A-7

Dead fuel moisture content corrections for month of February, March, April, August, September and October

Aspect	%Slope	0800>			1000>			1200>			1400>			1600>			1800>		
		B	L	A	B	L	A	B	L	A	B	L	A	B	L	A	B	L	A
UNSHADED – LESS THAN 50% SHADING OF SURFACE FUELS																			
N	0-30	3	4	5	1	2	3	1	1	2	1	1	2	1	2	3	3	4	5
	31+	3	4	5	3	3	4	2	3	4	2	3	4	3	3	4	3	4	5
E	0-30	3	4	5	1	2	3	1	1	1	1	1	2	1	2	4	3	4	5
	31+	3	3	4	1	1	1	1	1	1	1	2	3	3	4	5	4	5	6
S	0-30	3	4	5	1	2	2	1	1	1	1	1	1	1	2	3	3	4	5
	31+	3	4	5	1	2	2	0	1	1	0	1	1	1	2	2	3	4	5
W	0-30	3	4	5	1	2	3	1	1	1	1	1	1	1	2	3	3	4	5
	31+	4	5	6	3	4	5	1	2	3	1	1	1	1	1	1	3	3	4
SHADED – 50% OR MORE SHADING OF SURFACE FUELS																			
N	all	4	5	6	4	5	5	3	4	5	3	4	5	4	5	5	4	5	6
E	all	4	5	6	3	4	5	3	4	5	3	4	5	4	5	6	4	5	6
S	all	4	5	6	3	4	5	3	4	5	3	4	5	3	4	5	4	5	6
W	all	4	5	6	4	5	6	3	4	5	3	4	5	3	4	5	4	5	6

Source: NWCG (2006, p27)

Appendix A-8

Dead fuel moisture content corrections for month of November, December and January

UNSHADED – LESS THAN 50% SHADING OF SURFACE FUELS																			
Aspect	%Slope	0800>			1000>			1200>			1400>			1600>			1800>		
		B	L	A	B	L	A	B	L	A	B	L	A	B	L	A	B	L	A
N	0-30	4	5	6	3	4	5	2	3	4	2	3	4	3	4	5	4	5	6
	31+	4	5	6	4	5	6	4	5	6	4	5	6	4	5	6	4	5	6
E	0-30	4	5	6	3	4	4	2	3	3	2	3	3	3	4	5	4	5	6
	31+	4	5	6	2	3	4	2	2	3	3	4	4	4	5	6	4	5	6
S	0-30	4	5	6	3	4	5	2	3	3	2	2	3	3	4	4	4	5	6
	31+	4	5	6	2	3	3	1	1	2	1	1	2	2	3	3	4	5	6
W	0-30	4	5	6	3	4	5	2	3	3	2	3	3	3	4	4	4	5	6
	31+	4	5	6	4	5	6	3	4	4	2	2	3	2	3	4	4	5	6
SHADED – 50% OR MORE SHADING OF SURFACE FUELS																			
N	all	4	5	6	4	5	6	4	5	6	4	5	6	4	5	6	4	5	6
E	all	4	5	6	4	5	6	4	5	6	4	5	6	4	5	6	4	5	6
S	all	4	5	6	4	5	6	4	5	6	4	5	6	4	5	6	4	5	6
W	all	4	5	6	4	5	6	4	5	6	4	5	6	4	5	6	4	5	6

Source: NWCG (2006, p28)

Appendix A-9

Calculation of sub-components for Rothermel equation

Optimum Packing ratio (β_{op}) = $3.348\ \boldsymbol{\sigma}^{-0.8189}$ [Equation 37, Rothermel (1972)]

Component A of Optimum reaction velocity equation:

$A = 1/(4.77\boldsymbol{\sigma}^{0.1} - 7.27)$ [Equation 39, Rothermel(1972)]

Moisture Damping Coefficient:

$n_M = 1 - 2.59(M_f/M_x) + 5.11\ (M_f/M_x)^2 - 3.52\ (M_f/M_x)^3$ [Equation 29, Rothermel(1972)]

Mineral Damping Coefficient:

$n_S = 0.174\ S_e^{-0.19}$ [Equation 30, Rothermel(1972)]

Propagating Flux ratio:

$\xi = (192 + 0.2595\boldsymbol{\sigma})^{-1} \exp[(0.792 + 0.681\ \boldsymbol{\sigma}^{0.5})(\beta + 0.1)$

[Equation 42, Rothermel(1972)]

Net Fuel Loading:

$W_n = W_o/(1+S_T)$ [Equation 24, Rothermel(1972)]

Ovendry Bulk density:

$\rho_b = W_o/\boldsymbol{\delta}$ [Equation 40, Rothermel(1972)]

Effective heating number:

$\varepsilon = \exp(-138/\boldsymbol{\sigma})$ [Equation 14, Rothermel(1972)]

Heat of preignition in BTU/lb:

$Q_{ig} = 250 + 1116\ M_f$ [Equation 12, Rothermel(1972)]

Packing ratio:

$\beta = \rho_b/\rho_p$ [Equation 31, Rothermel(1972)]

Maximum reaction velocity:

$T_{max} = \sigma^{1.5} (495 + 0.0594\ \sigma^{1.5})^{-1}$ [Equation 36, Rothermel(1972)]

Optimum reaction velocity:

$T = T_{max} (\beta/\beta_{op})\ A \exp[A(1-\beta/\beta_{op})]$ [Equation 38, Rothermel(1972)]

Reaction intensity in BTU ft^{-2} min^{-1}

$I_R = T\ W_n\ h\ n_M\ n_S$ [Equation 27, Rothermel(1972)]

Source: Rothermel (1972)

Appendix B-1

MODIS satellite and band Specifications

Orbit: 705 km, 10:30 a.m. descending node (Terra) or 1:30 p.m. ascending node (Aqua), sun-synchronous, near-polar, circular
Scan Rate: 20.3 rpm, cross track
Swath Dimensions: 2330 km (cross track) by 10 km (along track at nadir)
Telescope: 17.78 cm diam. off-axis, afocal (collimated), with intermediate field stop
Size: 1.0 x 1.6 x 1.0 m
Weight: 228.7 kg
Power: 162.5 W (single orbit average)
Data Rate: 10.6 Mbps (peak daytime); 6.1 Mbps (orbital average)
Quantization: 12 bits
Spatial Resolution: 250 m (bands 1-2), 500 m (bands 3-7), 1000 m (bands 8-36)
Design Life: 6 years

Primary Use	Band	Bandwidth (nm)	Spectral Radiance (W/m^2 -μm-sr)	Required Signal-to-Noise Ratio(SNR)
Land/Cloud/Aerosols Boundaries	1	620 - 670	21.8	128
	2	841 - 876	24.7	201
Land/Cloud/Aerosols Properties	3	459 - 479	35.3	243
	4	545 - 565	29.0	228
	5	1230 - 1250	5.4	74
	6	1628 - 1652	7.3	275
	7	2105 - 2155	1.0	110

Source: MODIS (2016)

Appendix B-2

MODIS Terra Product Descriptions: MOD09

Last update: February 9 2015

Data Set Short Name: MOD09

Data Set Long Name:
MODIS/Terra Atmospherically Corrected Surface Reflectance 5Min L2 Swath 250m 500m 1km

Platform: Terra

Instrument: MODIS

Product Description:
MODIS/Terra Near Real Time (NRT) L2 Surface Reflectance, 5-Min Swath 250m, 500m, and 1km (MOD09). This product is computed from the MODIS Level 1B land bands 1, 2, 3, 4, 5, 6, and 7 (centered at 648 nm, 858 nm, 470 nm, 555 nm, 1240 nm, 1640 nm, and 2130 nm, respectively). The product is an estimate of the surface spectral reflectance for each band as it would have been measured at ground level if there were no atmospheric scattering or absorption. The surface-reflectance product is the input for product generation for several land products: vegetation Indices (VIs), BRDF, thermal anomaly, snow/ice, and Fraction of Photosynthetically Active Radiation/Leaf Area Index (FPAR/LAI).

Production Frequency: 288 files per day

Spatial Coverage: Global

Spatial Resolution: 250 m 500m and 1 km at nadir

File Size (MB): 350

Citation:
Dataset Originator/Creator:

MODIS Land Science Team/MODIS Adaptive Processing System (MODAPS)

Data Authors:
Dataset DOI:
http://dx.doi.org/10.5067/MODIS/MOD09.NRT.006

Source: MODIS(2016)

Appendix B-3

Project projection system

```
PROJCS["NAD83 / Conus Albers",
    GEOGCS["NAD83",
        DATUM["North_American_Datum_1983",
            SPHEROID["GRS 1980",6378137,298.257222101,
                AUTHORITY["EPSG","7019"]],
            TOWGS84[1,1,-1,0,0,0,0],
            AUTHORITY["EPSG","6269"]],
        PRIMEM["Greenwich",0,
            AUTHORITY["EPSG","8901"]],
        UNIT["degree",0.0174532925199433,
            AUTHORITY["EPSG","9122"]],
        AUTHORITY["EPSG","4269"]],
    PROJECTION["Albers_Conic_Equal_Area"],
    PARAMETER["standard_parallel_1",29.5],
    PARAMETER["standard_parallel_2",45.5],
    PARAMETER["latitude_of_center",23],
    PARAMETER["longitude_of_center",-96],
    PARAMETER["false_easting",0],
    PARAMETER["false_northing",0],
    UNIT["metre",1,
        AUTHORITY["EPSG","9001"]],
    AXIS["X",EAST],
    AXIS["Y",NORTH],
    AUTHORITY["EPSG","5070"]]
```

Appendix B-4

Projection system for LANDFIRE data

```
PROJCS["USA_Contiguous_Albers_Equal_Area_Conic_USGS_version",
    GEOGCS["GCS_North_American_1983",
        DATUM["D_North_American_1983",
            SPHEROID["GRS_1980",6378137.0,298.257222101]],
        PRIMEM["Greenwich",0.0],
        UNIT["Degree",0.0174532925199433]],
    PROJECTION["Albers"],
    PARAMETER["False_Easting",0.0],
    PARAMETER["False_Northing",0.0],
    PARAMETER["Central_Meridian",-96.0],
    PARAMETER["Standard_Parallel_1",29.5],
    PARAMETER["Standard_Parallel_2",45.5],
    PARAMETER["Latitude_Of_Origin",23.0],
    UNIT["Meter",1]]
```

Source: LANDFIRE (2015)

Appendix B-5

Projection system for National Weather Service (NWS) weather data

```
PROJCS["NDFD",
    GEOGCS["unnamed",
```

```
    DATUM["unknown",
        SPHEROID["Sphere",6371200,0,
            AUTHORITY["EPSG","7035"]]],
    PRIMEM["Greenwich",0],
    UNIT[,0.0174532925199433]],
PROJECTION["Lambert_Conformal_Conic_2SP"],
PARAMETER["standard_parallel_1",25],
PARAMETER["standard_parallel_2",25],
PARAMETER["latitude_of_origin",25],
PARAMETER["central_meridian",-95],
PARAMETER["false_easting",0],
PARAMETER["false_northing",0],
UNIT["metre",1,
    AUTHORITY["EPSG","9001"]]]
```

Source: National Weather Service (NWS)

Appendix – C

Appendix -C1

Script name: generateellipse/__init__.py

```
import numpy

def generateellipse(semimaj=1, semimin=1, c=1, phi=0, x_cent=0, y_cent=0,
theta_num=50):
    """
        create ellipse geometry
        Parameters:
        -----------------
        semimaj: length of semi-major axis,
        semimin: length of semi-minor axis
        phi:  Orientation of ellipse
        x_cent: Center X of ellipse
        y_cent: Center Y of ellipse
        theta_num: Number of points to generate per ellipse; default 50
        c: distance from focal point to center
        Returns:
        --------------------
        ellipse_x, ellipse_y: coordinates of ellipse
         new_centers: new center after that
        """
    theta = numpy.linspace(0, 2 * numpy.pi, theta_num)
    r = semimaj * semimin / numpy.sqrt((semimin * numpy.cos(theta)) ** 2 +
(semimaj * numpy.sin(theta)) ** 2)
    x = r * numpy.cos(theta)
    y = r * numpy.sin(theta)
    data = numpy.array([x, y])
    R = numpy.array([[numpy.cos(phi), -numpy.sin(phi)], [numpy.sin(phi),
numpy.cos(phi)]])
    data = numpy.dot(R, data)
    # percentage of backfire %
    frnt = c / semimaj
    if (0 <= phi < numpy.pi * 0.5):
        x_cent1 = x_cent + frnt * numpy.cos(phi) * semimaj
        y_cent1 = y_cent + frnt * numpy.sin(phi) * semimaj
    elif (numpy.pi * 0.5 <= phi < numpy.pi):
        x_cent1 = x_cent - frnt * numpy.cos(phi) * semimaj
        y_cent1 = y_cent - frnt * numpy.sin(phi) * semimaj
    elif (numpy.pi <= phi < numpy.pi * 1.5):
        x_cent1 = x_cent + frnt * numpy.cos(phi) * semimaj
        y_cent1 = y_cent + frnt * numpy.sin(phi) * semimaj
    else:
        x_cent1 = x_cent - frnt * numpy.cos(phi) * semimaj
        y cent1 = y cent - frnt * numpy.sin(phi) * semimaj

    # find fire front point (tip of ellipse)
    if (3 * numpy.pi / 2 < phi < 2 * numpy.pi):
        x_front = x_cent - frnt * 2 * numpy.cos(phi) * semimaj
        y_front = y_cent - frnt * 2 * numpy.sin(phi) * semimaj
    elif (0 <= phi < numpy.pi / 2):
        x_front = x_cent + frnt * 2 * numpy.cos(phi) * semimaj
        y_front = y_cent + frnt * 2 * numpy.sin(phi) * semimaj
    elif (numpy.pi / 2 <= phi < numpy.pi):
        x_front = x_cent - frnt * 2 * numpy.cos(phi) * semimaj
        y_front = y_cent - frnt * 2 * numpy.sin(phi) * semimaj
    else:
        x_front = x_cent + frnt * 2 * numpy.cos(phi) * semimaj
        y_front = y_cent + frnt * 2 * numpy.sin(phi) * semimaj
    # new_center.append([x_front,y_front])
    data[0] += x_cent1
    data[1] += y_cent1
    return [data[0], data[1], [x_front, y_front]]
```

```
# create ellipse with center, axis and rotation
def Ellipse(a, b, an, x0, y0):  # an is the rotational angle
    cos_a, sin_a = numpy.cos(an * numpy.pi / 180), numpy.sin(an * numpy.pi /
180)
    the = numpy.linspace(0, 2 * numpy.pi, points)
    x_new, y_new = x0 - cos_a * a, y0 - sin_a * a
    # Here goes the general ellpse, x0, y0 is the origin of the ellipse in xy
plane
    X = a * numpy.cos(the) * cos_a - sin_a * b * numpy.sin(the) + x_new
    Y = a * numpy.cos(the) * sin_a + cos_a * b * numpy.sin(the) + y_new
    return X, Y
```

Appendix -C2
Script name: bishrant-rothermel/__init__.py

```
import math
# Basic variables for rothermel equation
def calculateROS_process(U=0, sav=0, mf=0, mx=0, se=0, wo=0, fuel_depth=0,
slope=0, h=0, st=0, rp=0):
    def calcComponentA(sav):
        # calculates component A of rothermel equation 39
        # sav: fuel particle surface-area-to volumne ratio 1/ft
        A = 1 / (4.774 * pow(sav, 0.1) - 7.27)
        return A
    def calcMaxReactionVelocity(sav):
        # calculates maximum reaction velocity of rothermel equation 36
        # sav: fuel particle surface-area-to volumne ratio 1/ft
        pow sav = pow(sav, 1.5)
        tmax = pow_sav * pow((495 + 0.0594 * pow_sav), -1)
        return tmax
    def calcOptimumPackingRatio(sav):
        # calculates optimum packing ratio of rothermel equation 37
        # sav: fuel particle surface-area-to volumne ratio 1/ft
        optimumpackingratio = 3.348 * pow(sav, -0.8189)
        return optimumpackingratio
    def calcMoistureDampingCoeff(mf, mx):
        # calculates moisture damping coefficient using rothermel equation 29
        # parameters: mf: fuel particle moisture content
        # parameters: mx: moisture of extinction
        r = mf / mx
        moisturecoeff = 1 - 2.59 * (r) + 5.11 * pow(r, 2) - 3.52 * pow(r, 3)
        return moisturecoeff
    def calcMineralDampingCoeff(se):
        # calculates mineral damping coefficient using rothermel equation 30
        # parameters: se: fuel particle effective mineral content
        mineralcoeff = 0.174 * pow(se, -0.19)
        return mineralcoeff
    def calcOvenDryBulkDensity(wo, fuel depth):
        # calculates Ovendry bulk density in lb/ft3 using rothermel equation 40
        # parameters: wo: ovendry fuel loading, Ib/ft.
        # parameters: fuel_depth:  fuel depth, ft
        ovendrybulkdensity = wo / fuel_depth
        return ovendrybulkdensity
    def calcPackingRatio(rb, rp):
        # ---- LOOK FOR PACKING RATIO OBTAINED FROM EXCEL AS WELL ------#
        # calculates packing ratio using rothermel equation 31
        # parameters: rb:  Overldry bulk density, lb/ft3
        # parameters: rp:  ovendry particle density, lb/ft3
        packingratio = rb / rp
        return packingratio
    def calcPropagaingFluxRatio(sav, packingratio):
        # calculates propagating flux ratio using rothermel equation 42
        # parameters: sav: fuel particle surface-area-to volumne ratio 1/ft
        # parameters: packingratio: packing ratio obtained from equation  31
        propfluxratio = (1 / (192 + 0.2595 * sav)) * math.exp((0.792 + 0.681 *
pow(sav, 0.5)) * (packingratio + 0.1))
```

```
        return propfluxratio
    def calcHeatofPreigntion(mf):
        # calculates heat of pre-ignition in B.t.u/lb from rothermel equation
12
        # parameters: mf: fuel particle moisture content
        qig = 250 + 1116 * mf
        return qig
    def calcEffectiveHeatingno(sav):
        # calculates effective heating number using rothermel equation 14
        # parameters: sav: fuel particle surface-area-to volumne ratio 1/ft
        effecheating = math.exp(-138 / sav)
        return effecheating
    def calcSlopeFactor(packingratio, slope):
        # calculates slope factor using rothermel equation 51
        # parameters: packingratio: packing ratio obtained from equation  31
        # parameters: slope:  tan (phi) vertical rise/horizontal distance
        slope_factor = 5.275 * pow(packingratio, -0.3) * (pow(slope, 2))
        return slope_factor
    def calcNetFuelLoading(wo, st):
        # calcualtes Net fuel loading, 1b./ft2 using rothemel equation 24
        # parameters: wo: ovendry fuel loading, Ib. /ft.
        # parameters: st: fuel particle total mineral content
        net_fuel_loading = wo / (1 + st)
        return net_fuel_loading
    def calcWindCoeff(midflame_wspd, sav, packingratio, optimumpackingratio):
        # calculates Wind coefficient using rothermel equation 47
        # parameters
        C = 7.47 * math.exp(-0.133 * pow(sav, 0.55))
        B = 0.02526 * pow(sav, 0.54)
        E = 0.715 * math.exp(-3.59 * sav * pow(10, -4))
        wind_coeff = C * pow(midflame_wspd, B) * pow((packingratio /
optimumpackingratio), -1 * E)
        return wind_coeff
    def calcOptimumReactionVelocity(tmax, packingratio, optimumpackingratio,
A):
        # calculates Optimum reaction velocity, (min)-1 using rothermel
equation 38
        # parameters: tmax: ~ Maximum reaction velocity from eq(36) min.-1
        # A: component A obtained from eq 39
        opt_reaction_vel = tmax * pow((packingratio / optimumpackingratio), A)
* math.exp(
            A * (1 - packingratio / optimumpackingratio))
        return opt_reaction_vel
    def calcReactionIntensity(opt_reaction_vel, net_fuel_loading, h, nm, ns):
        # parameters: h: fuel particle low heat content B.t.u/lb
        # parameters: nm: Moisture damping coefficient from eq 29
        # ns: Mineral Damping coefficient from eq 30
        reaction_intensity = opt_reaction_vel * net_fuel_loading * h * nm * ns
        return reaction_intensity
    def calcROS(reaction_intensity, propfluxratio, slope_factor, wind_coeff,
ovendrybulkdensity, effecheating, qig):
        # calculates Rate of spread, ft ./min using rothermel equation 52
        ros = (reaction_intensity * propfluxratio * (1 + wind_coeff +
slope_factor)) / (
            ovendrybulkdensity * effecheating * qig)
        return ros
    A = calcComponentA(sav)
    tmax = calcMaxReactionVelocity(sav)
    optimumpackingratio = calcOptimumPackingRatio(sav)
    nm = calcMoistureDampingCoeff(mf, mx)
    ns = calcMineralDampingCoeff(se)
    rb = calcOvenDryBulkDensity(wo, fuel_depth)
    packingratio = calcPackingRatio(rb, rp)
    propfluxratio = calcPropagaingFluxRatio(sav, packingratio)
    qig = calcHeatofPreigntion(mf)
    effecheating = calcEffectiveHeatingno(sav)
    slope_factor = calcSlopeFactor(packingratio, slope)
    net_fuel_loading = calcNetFuelLoading(wo, st)
```

```
    wind_coeff = calcWindCoeff(U, sav, packingratio, optimumpackingratio)
    opt_reaction_vel = calcOptimumReactionVelocity(tmax, packingratio,
optimumpackingratio, A)
    reaction_intensity = calcReactionIntensity(opt_reaction_vel,
net_fuel_loading, h, nm, ns)
    ros = calcROS(reaction_intensity, propfluxratio, slope_factor, wind_coeff,
rb, effecheating, qig)
    return ros
if __name__ == '__main__':
    # 10-hr dead fuel SAV is 109 1/ft, and 100-hr SAV is 30 1/ft.
    # Total mineral content is 5.55 percent; effective (silica-free) mineral
content is 1.00 percent.
    # Ovendry fuel particle density is 32 lb/ft3.
    # wo ovendry fuel loading, Ib. /ft.  --> Fine Fuel looading (Excel)
    # fuel_depth, ft.  --> fuel bed depth (Excel)
    # a, fuel particle surface-area-to-volume ratio, 1/ft. --> charac_SAV
(Excel)
    # h, fuel particle low heat content, B. t.u./lb.  --> heat content (Excel)
    # p  ovendry particle density, lb./ft.3  --> 32 lb/ft3 (Scott paper)
    # Mf, fuel particle moisture content, lb. moisture   --> predicted from
excel comparison table
    # ST, fuel particle total mineral content, lb. min,erals --> 5.55 percent
(Scott paper)
    # Se, fuel particle effective mineral content, lb. silica- fi-ee minerals
--> 1.00 percent (Scott Paper)
    # U,  wind velocity at midflame height, ft ./min. --> from raster
    # tan 4 , slope, vertical rise/horizontal distance --> from raster
    # Mx, moisture content of extinction. --> dead_moisture_ext (Excel)

    # run main function
    R = calculateROS_process(U=0, sav=0, mf=0, mx=0, se=0, wo=0, fuel_depth=0,
slope=0, h=0, st=0, rp=0)
def calcellipseparam(R0, U0):
    # convert ROS and U from ft/min to meter/sec
    R = R0 * 0.00508
    U = U0 * 0.00508
    LB = 0.936 * math.exp(0.2566 * U) + 0.461 * math.exp(-0.1548 * U) - 0.397
    # print LB
    HB = (LB + pow((pow(LB, 2) - 1), 0.5)) / (LB - pow((pow(LB, 2) - 1), 0.5))
    a = 0.5 * (R + R / HB) / (LB)
    b = (R + R / HB) / 2.0
    c = b - R / HB
    return a, b, c

# a, b, c = calcellipseparam(R, U)
import numpy

def generateellipse(semimaj=1, semimin=1, phi=0, x_cent=0, y_cent=0,
theta_num=4, c=1):
    """
        create ellipse geometry
        Parameters:
        -----------------
        semimaj: length of semi-major axis,
        semimin: length of semi-minor axis
        phi:  Orientation of ellipse
        x_cent: Center X of ellipse
        y_cent: Center Y of ellipse
        theta_num: Number of points to generate per ellipse; default 60
        c: distance from focal point to center
        Returns:
        --------------------
        ellipse_x, ellipse_y: coordinates of ellipse
         new_centers: new center after that


        """
    theta = numpy.linspace(0, 2 * numpy.pi, theta_num)
```

```
    r = semimaj * semimin / numpy.sqrt((semimin * numpy.cos(theta)) ** 2 +
(semimaj * numpy.sin(theta)) ** 2)
    x = r * numpy.cos(theta)
    y = r * numpy.sin(theta)
    data = numpy.array([x, y])
    R = numpy.array([[numpy.cos(phi), -numpy.sin(phi)], [numpy.sin(phi),
numpy.cos(phi)]])
    data = numpy.dot(R, data)
    # percentage of backfire 15%
    frnt = c / semimaj
    # frnt = 0.85
    if (0 <= phi < numpy.pi * 0.5):
        x_cent1 = x_cent + frnt * numpy.cos(phi) * semimaj
        y_cent1 = y_cent + frnt * numpy.sin(phi) * semimaj
    elif (numpy.pi * 0.5 <= phi < numpy.pi):
        x_cent1 = x_cent - frnt * numpy.cos(phi) * semimaj
        y_cent1 = y_cent - frnt * numpy.sin(phi) * semimaj
    elif (numpy.pi <= phi < numpy.pi * 1.5):
        x_cent1 = x_cent + frnt * numpy.cos(phi) * semimaj
        y_cent1 = y_cent + frnt * numpy.sin(phi) * semimaj
    else:
        x_cent1 = x_cent - frnt * numpy.cos(phi) * semimaj
        y_cent1 = y_cent - frnt * numpy.sin(phi) * semimaj

    # find fire front point (tip of ellipse)
    if (3 * numpy.pi / 2 < phi < 2 * numpy.pi):
        x_front = x_cent - frnt * 2 * numpy.cos(phi) * semimaj
        y_front = y_cent - frnt * 2 * numpy.sin(phi) * semimaj
    elif (0 <= phi < numpy.pi / 2):
        x_front = x_cent + frnt * 2 * numpy.cos(phi) * semimaj
        y_front = y_cent + frnt * 2 * numpy.sin(phi) * semimaj
    elif (numpy.pi / 2 <= phi < numpy.pi):
        x_front = x_cent - frnt * 2 * numpy.cos(phi) * semimaj
        y_front = y_cent - frnt * 2 * numpy.sin(phi) * semimaj
    else:
        x_front = x_cent + frnt * 2 * numpy.cos(phi) * semimaj
        y_front = y_cent + frnt * 2 * numpy.sin(phi) * semimaj
    # new_center.append([x_front,y_front])

    data[0] += x_cent1
    data[1] += y_cent1
    return [data[0], data[1], [x_front, y_front]]


def calcEllipsepoint(a, b, c, r):
    phi = r * math.pi / 180
    cos_phi = round((numpy.cos(phi)), 10)
    sin_phi = round(math.sin(phi), 10)
    xs = 0
    ys = 0
    print a, b, c
    x1_upper = pow(a, 2) * cos_phi * (xs * sin_phi + ys * cos_phi) - pow(b, 2)
* sin_phi * (xs * cos_phi - ys * sin_phi)
    lower = pow((pow(b, 2) * pow((xs * cos_phi + ys * sin_phi), 2) - pow(a, 2)
* pow((xs * sin_phi - ys * cos_phi), 2)),
                1)
    # print x1_upper
    if (x1_upper == 0.0 or lower == 0.0):
        x1_ratio = 0
    else:
        x1_ratio = x1_upper / lower
    x1 = x1_ratio + c * sin_phi
    print x1
    y1_upper = -pow(a, 2) * sin_phi * (xs * sin_phi + ys * cos_phi) - pow(b, 2)
* cos_phi * (
        xs * cos_phi - ys * sin_phi)
    if (y1_upper == 0.0 or lower == 0.0):
        y1_ratio = 0
```

```
    else:
        y1_ratio = y1_upper / lower
    y1 = y1_ratio + c * cos_phi
    print y1_ratio
    # return x1, y1
    # print calcEllipsepoint(a,b,c,wdir)
```

Appendix -C3

Script name: winddata_downloader/__init__.py

```
import time
start_time = time.time()
import urllib
from datetime import datetime
import gdal
import sqlite3
import math
import bishrant_rothermel
import time
import os
import sip
import sys
sip.setapi('QString', 2)
sip.setapi("QVariant", 2)
sip.setapi("QDate", 2)
sip.setapi("QDateTime", 2)
sip.setapi("QTextStream", 2)
sip.setapi("QTime", 2)
sip.setapi("QUrl", 2)
qgis_path = "C://OSGeo4W64"
from qgis.core import *
QgsApplication.setPrefixPath(qgis_path, True)
qgs = QgsApplication(sys.argv, True)
qgs.initQgis()
sys.path.append('C:\\OSGeo4W64\\apps\\qgis\\python\\plugins')
from processing.core.Processing import Processing
Processing.initialize()
from processing.tools import *
os.chdir("C:\\Fire Simulation\\")
QgsApplication.setPrefixPath(qgis_path, True)
qgs = QgsApplication(sys.argv, True)
qgs.initQgis()
sys.path.append('C:\\OSGeo4W64\\apps\\qgis\\python\\plugins')
from processing.core.Processing import Processing
import processing
Processing.initialize()
from processing.tools import *
from qgis.core import (QgsVectorLayer)
from qgis.analysis import QgsGeometryAnalyzer
os.chdir("C:\\Fire_Simulation\\")
# storing center calculation here as well to save memory
def fn_generate_centers(geojson_str):
    feat = []
    geojson_line = QgsVectorLayer(geojson_str, "json", "ogr")
    def geojsontoshp(geojson_str, centervectorname):
        # convert geojson to shapefile
        # @param: geojson_str: geojson string
        # @param: centervectorname: name of output vector shapefile
        temp_center = QgsVectorLayer(geojson_str, "json", "ogr")
        temp_center.setCrs(QgsCoordinateReferenceSystem(4326,
QgsCoordinateReferenceSystem.EpsgCrsId))
        QgsGeometryAnalyzer().dissolve(temp_center, centervectorname,
onlySelectedFeatures=False, uniqueIdField=-1,
                                       p=None)
    geojsontoshp(geojson_str, "c:\\Fire_Simulation\\data\\centerline0.shp")
    processing.runalg("qgis:reprojectlayer",
QgsVectorLayer("c:\\fire_simulation\\data\\centerline0.shp", "sd", 'ogr'),
```

```
                      "EPSG:5070", "c:\\fire_simulation\\data\\centerline.shp")

    def createPointsAt(distance, geom):
        total = geom.length()
        dist = distance
        while dist < total:
            pt = geom.interpolate(dist)
            feat.append([pt.asPoint().x(), pt.asPoint().y()])
            # print pt.asPoint().toString()
            dist = dist + distance
    def pointsAlongLine(distance):
        # Create a new memory layer and add a distance attribute
        # vl = QgsVectorLayer("c:\\fire_simulation\\Point.shp", "distance
nodes", "ogr")
        # pr = vl.dataProvider()
        # pr.addAttributes( [ QgsField("distance", QVariant.Int) ] )
        layer = QgsVectorLayer("c:\\fire_simulation\\data\\centerline.shp",
"centers", "ogr")
        # layer = iface.mapCanvas().currentLayer()
        # Loop though all the selected features
        total = layer.getFeatures()

        for feature in layer.getFeatures():
            geom = feature.geometry()
            createPointsAt(distance, geom)
    pointsAlongLine(5)
    print feat
    return feat
    print 'in test 1, unproductive'
"""downloads wind data when requested, mosaics them, projects them to specified
CRS"""
# @todo create metadata table for unused layers and store timestamp and data
validity in flat file sys --> RESOLVED PARTIALLY
os.chdir("c:\\Fire_simulation\\data\\wind\\")
# use extent of grids created in postgis for cutting rasters
# max swath is 20 degrees
# variables to hold memory rasters
def download_dynamic_rasters():
    ras_dict = {'wdir': {}, 'wspd': {}, 'rhm': {}, 'temp': {}}
    # grid for wyoming
    grids = [[-112.5, 39.5, -102.0, 46.5, 1]]
    # standard url:
http://graphical.weather.gov/xml/gribcut.php?var=wdir&lat1=30&lon1=-
100&lat2=50&lon2=-80
    baseurl = "http://graphical.weather.gov/xml/gribcut.php?"
    # variables to download for
    variables = ['wdir', 'wspd', 'rhm', 'temp']
    filename_array = []
    for vars in variables:
        time = datetime.utcnow()
        # classify hour in 3 hour interval as NOAA
        hours = time.hour
        cc = [0, 3, 6, 9, 12, 15, 18, 21]
        class_hour = {0: [0, 1, 2], 3: [3, 4, 5], 6: [6, 7, 8], 9: [9, 10, 11],
12: [12, 13, 14], 15: [15, 16, 17],
                      18: [18, 19, 20], 21: [21, 22, 23]}
        new_hr = int(hours / 3) * 3
        hr_str = "%02d" % (new_hr,)
        filename = vars + "_" + time.strftime("%Y%m%d") + "_" + hr_str + ".tif"
        filename_array.append(filename)
        if os.path.isfile(filename):
            continue
        else:
            # loop through all variables and all extent polygons
            for polygons in grids:
                # append extent and variables to baseurl
                newurl = baseurl + "var=%s&lat1=%s&lon1=%s&lat2=%s&lon2=%s" % (
                    vars, polygons[1], polygons[0], polygons[3], polygons[2])
```

```
                # output filename
                obj_name = vars + str(polygons[4]) + ".grb"
                # grab the file
                print newurl
                urllib.urlretrieve(newurl, obj_name)

                # totalbands= gdal_frband.GetRasterCount()
                # convert GRIB2 to GTiff and store in memory, using first
layer; closest forecast
                ras_dict[vars][polygons[4] - 1] = gdal.Translate("",
gdal.Open(obj_name), format='MEM',

bandList=range(1, 30))

            gdal.Warp(filename, [ras_dict[vars][0]], dstSRS='EPSG:5070')
            # ras_dict[vars][1], ras_dict[vars][2], ras_dict[vars][3],
ras_dict[vars][4], ras_dict[vars][5]],

            # store the metadata in multi band raster:
            main_file = gdal.Open(filename)
            wdir_grib1 = gdal.Open("c:\\Fire_simulation\\data\\wind\\" + vars +
"1.grb")
            # for i in range(1, 30):
            #
(main_file.GetRasterBand(i)).SetMetadata((wdir_grib1.GetRasterBand(i)).GetMetad
ata())
            del main_file
            del wdir_grib1
            # remove the temp grb files
            # for ii in range(1, 2):
            #   os.remove(vars + str(ii) + ".grb")

    # return filenames of the rasters
    return filename_array
    # mosaic them and reproject to WGS84


# gdal.Warp("wdir.tif",
    #           [ras_dict['wdir'][0], ras_dict['wdir'][1], ras_dict['wdir'][2],
ras_dict['wdir'][3], ras_dict['wdir'][4],
    #            ras_dict['wdir'][5]], dstSRS='EPSG:6703')
    # # gdal.Warp("wspd.tif",
   # #           [ras_dict['wspd'][0], ras_dict['wspd'][1],
ras_dict['wspd'][2], ras_dict['wspd'][3], ras_dict['wspd'][4],
    # #            ras_dict['wspd'][5]])
    #
    # #store the metadata in multi band raster:
    # wdir_ras= gdal.Open("wdir.tif")
    # wdir_grib1= gdal.Open("wdir1.grb")
    # for i in range (1, 11):
    #
(wdir_ras.GetRasterBand(i)).SetMetadata((wdir_grib1.GetRasterBand(i)).GetMetada
ta())
    # del wdir_ras
    # del wdir_grib1

    # handle multi band data for wind speed as well
    # wspd_ras= gdal.Open("wspd.tif")
    # wspd_grib1= gdal.Open("wspd1.grb")
    # for i in range (1, 11):
    #
(wspd_ras.GetRasterBand(i)).SetMetadata((wspd_grib1.GetRasterBand(i)).GetMetada
ta())
    # del wspd_ras
    # del wspd_grib1
    #
    # for i in range(1,6):
    #     os.remove("wspd%s.grb" % i)
```

```
    #     os.remove("wdir%s.grb" % i
def calculate_fire_vars(x=0, y=0):
    # connect to sqllite database to fetch fuel data
    conn = sqlite3.connect("C:\\Fire_Simulation\\data\\landfire_fuels.db")
    def conv_to_f_string(celcius):
        return str(int(9.0 / 5.0 * celcius + 32))
    dynamic_files = download_dynamic_rasters()
    # print dynamic_files
    wdir_raster = QgsRasterLayer(dynamic_files[0], "wdir")
    wspd_raster = QgsRasterLayer(dynamic_files[1], "wspd")
    rhm_raster = QgsRasterLayer(dynamic_files[2], "rhm")
    temp_raster = QgsRasterLayer(dynamic_files[3], "temp")
    wdir = (wdir_raster.dataProvider().identify(QgsPoint(x, y),
QgsRaster.IdentifyFormatValue)).results()[1]
    wspd = (wspd_raster.dataProvider().identify(QgsPoint(x, y),
QgsRaster.IdentifyFormatValue)).results()[1]
    rhm = (rhm_raster.dataProvider().identify(QgsPoint(x, y),
QgsRaster.IdentifyFormatValue)).results()[1]
    temp = (temp_raster.dataProvider().identify(QgsPoint(x, y),
QgsRaster.IdentifyFormatValue)).results()[1]
    print x, y
    # find fuel moisture and its correction using sqlite tables
    cur = conn.cursor()
    sql = "select RH5_" + str(int(rhm / 5)) + " from fuel_moisture_humidity
where (temp_lower <= " + conv_to_f_string(
        temp) + " and temp_upper > " + conv_to_f_string(temp) + " )"
    cur.execute(sql)
    moisture_uncorrected = cur.fetchone()[0]

    # data from landfire
    landfire_raster = QgsRasterLayer("C:\\Fire_Simulation\\wy_landfire.tif",
"landfire")
    landfire = (landfire_raster.dataProvider().identify(QgsPoint(x, y),
QgsRaster.IdentifyFormatValue)).results()

    # extract slope, elevation and aspect and fuel type classification
    elev, slope, aspect, fuel_class = landfire[1], landfire[2], landfire[3],
int(landfire[4])
    if aspect < 0:
        aspect = 0
    # find aspect classification
    if (aspect >= 345 and aspect <= 360) or (aspect < 45):
        aspect_str = "N"
    elif (45 <= aspect < 135):
        aspect_str = "E"
    elif (135 <= aspect < 225):
        aspect_str = "S"
    else:
        aspect_str = "W"
    #find time of day and month to apply corrections to fuel moisture content
    time = datetime.utcnow()
    # classify hour in 3 hour interval as NOAA
    hours = time.hour
    month = time.month
    # categorize hour
    # if H_8L  H_8A   H_10B  H_10L  H_10A  H_12B  H_12L  H_12A  H_14B  H_14L
H_14A  H_16B  H_16L  H_16A  H_18B  H_18L  H_18A
    if hours < 8:
        hours_str = "H_8L"
    elif 8 <= hours < 10:
        hours_str = "H_10L"
    elif 12 <= hours < 14:
        hours_str = "H_12L"
    elif 14 <= hours < 16:
        hours_str = "H_14L"
    elif 16 <= hours < 18:
        hours_str = "H_16L"
    else:
```

```
        hours_str = "H_18L"

    month_str = "%02d" % (month,)

    sql1 = "select " + hours_str + " from fuel_moisture_corrections where
(slope_lower <= " + str(
        slope) + " and slope_upper > " + str(slope) + \
           " and aspect  = '" + aspect_str + "' and months like '%" + month_str
+ "%' and shaded = 'no' )"
    cur.execute(sql1)
    moisture_correction = cur.fetchone()[0]

    # identify fuel class dynamically using db and landfire inputs
    # check if the fuel type is NB (non burnable) values 90-99
    if (90 <= fuel_class <= 99) or (fuel_class == 0):
        return []
        pass
    else:
        # variables to be extracted from database
        # wo ovendry fuel loading, lb. /ft.  --> Fine Fuel looading (Excel)
        # a, fuel particle surface-area-to-volume ratio, 1/ft. --> charac_SAV
(Excel)
        # h, fuel particle low heat content, B. t.u./lb.  --> heat content
(Excel)
        # fuel_depth, ft.  --> fuel bed depth (Excel)
        sql_fuelmodel = "select dead_moisture_ext, fine_fuel_load, charac_SAV,
heat_content, FL_bed_depth  from fire_fuel_model where model_no =" + str(
            fuel_class)

        cur.execute(sql_fuelmodel)
        fuel_info = cur.fetchone()

        mx = fuel_info[0] / 100.0  # Mx, moisture content of extinction. -->
dead_moisture_ext (Excel)
        wo = fuel_info[1]  # ovendry fuel loading, lb. /ft.  --> Fine Fuel
looading (db)
        sav = fuel_info[2]
        h = fuel_info[3]  # fuel particle low heat content, B. t.u./lb.  -->
heat content (db)
        fuel_depth = fuel_info[4]  # fuel bed depth (db)

        # From Scott's paper
        # 10-hr dead fuel SAV is 109 1/ft, and 100-hr SAV is 30 1/ft.
        # Total mineral content is 5.55 percent; effective (silica-free)
mineral content is 1.00 percent.
        # Ovendry fuel particle density is 32 lb/ft3.
        se = 0.01  # from scott paper         # ST, fuel particle total mineral
content, lb. min,erals --> 5.55 percent (Scott paper)
        st = 0.0555  # ST, fuel particle total mineral content, lb. min,erals -
-> 5.55 percent (Scott paper)
        rp = 32  # ovendry particle density, lb./ft.3  --> 32 lb/ft3 (Scott
paper)
        st = 0.0555  # ST, fuel particle total mineral content, lb. min,erals -
-> 5.55 percent (Scott paper)
        # Mf, fuel particle moisture content, lb. moisture   --> predicted from
excel comparison table
        mf = (
             moisture_uncorrected + moisture_correction) / 100.0  # moisture
content apply correction and convert to fraction of 100
        # wdir,
        # return [ wspd*88, sav, mf, mx, se, wo, fuel_depth,
math.tan(slope*math.pi/180), h, st,  rp ]

        # print wspd*88, sav, mf, mx, se, wo, fuel_depth,
math.tan(slope*math.pi/180), h, st,  rp
        ros = bishrant_rothermel.calculateROS_process(wspd * 88, sav, mf, mx,
se, wo, fuel_depth,
                                                      math.tan(slope * math.pi
```

```
/ 180), h, st, rp)
        a, b, c = bishrant_rothermel.calcellipseparam(ros, wspd * 88)
        return [a, b, c, wdir]
if __name__ == "__main__":
    calculate_fire_vars(x=0, y=0)
    QgsApplication.exitQgis()
```

Appendix -C4

Script name: run_simulation.py

```
import fnmatch
import os
import sip
import sys
sip.setapi('QString', 2)
sip.setapi("QVariant", 2)
sip.setapi("QDate", 2)
sip.setapi("QDateTime", 2)
sip.setapi("QTextStream", 2)
sip.setapi("QTime", 2)
sip.setapi("QUrl", 2)
qgis_path = "C://OSGeo4W64"
import winddata_downloader
from qgis.core import *
QgsApplication.setPrefixPath(qgis_path, True)
qgs = QgsApplication(sys.argv, True)
qgs.initQgis()
sys.path.append('C:\\OSGeo4W64\\apps\\qgis\\python\\plugins')
from processing.core.Processing import Processing
import processing
Processing.initialize()
from processing.tools import *
from qgis.core import (QgsVectorLayer)
os.chdir("C:\\Fire_Simulation\\")
import geojson
os.chdir("C:\\Fire_Simulation\\")
from geojson import Polygon, Feature, FeatureCollection
from qgis.analysis import QgsGeometryAnalyzer
os.chdir("C:\\Fire Simulation\\data\\")
points = 50  # Number of points whicnh needs to construct the elipse
import json
new_center = []
# coordinate system object
crs = QgsCoordinateReferenceSystem(5070,
QgsCoordinateReferenceSystem.PostgisCrsId)
# @todo get pixel size dynamically
# script to generate ellipse dynamically
from generateellipse import *
# shapefile deletion and projection functions
from shapefile_methods import *
def simulator(geojson_str, simulation_time, simulation_steps):
    # Required by Qt4 to initialize the UI
    new_centers = []
    # processing.runalg("qgis:reprojectlayer",
QgsVectorLayer(geojson_str,"sd",'ogr'), "EPSG:5070",
"c:\\fire_simulation\\data\\centerline.shp")
    centers1 = winddata_downloader.fn_generate_centers(geojson_str)
    def runsimulation(centers_arg, iteration):
        """
        Executes simulation for specified center coordinate array
        Parameters:
        -----------------
        centers_arg: 2D array of center points format: [[1,1],[2,4]]
        iteration: iteration step
        Returns:
        --------------------
        Final raster layer
```

```
        New updated point for center
        """
        def generateEllipsegeojson(centers_arg):
            """
            Generates ellipse for center points
            Parameters:
            -----------------
            centers_arg: 2D centers array
            Returns:
            --------------------
            shape_geojson: geojson of new ellipses
            new_centers: updated center for ext iteration
            """
            crs = {"type": "name", "properties": {"name":
"urn:ogc:def:crs:EPSG::5070"}}
            featurearray = []
            new_centers = []
            for center_i in centers_arg:
                # print center_i
                # extract wind direction and speed dynamically
                fire_vars =
winddata_downloader.calculate_fire_vars(center_i[0], center_i[1])
                try:
                    # check if they are moving outside of our study area
                    # if so delete that index and move to next one
                    if (len(fire_vars) != 0):
                        # ellipse parameters are for meter/sec convert to
minutes and multiply by simulation time step
                        a, b, c, wind_dir = fire_vars[0] * 60 *
simulation_time, fire_vars[1] * 60 * simulation_time, \
                                            fire_vars[2] * 60 *
simulation_time, fire_vars[3]
                        # calculate ellipse size from windspeed
                        semi_major, semi_minor = a, b

                        ellipse = generateellipse(semimaj=semi_major,
semimin=semi_minor, c=c, x_cent=center_i[0],
                                                  y_cent=center_i[1],
phi=wind_dir * numpy.pi / 180)
                        geom = Polygon([[(ellipse[0][iii], ellipse[1][iii]) for
iii in range(len(ellipse[0]))]])
                        if geojson.is_valid(geom)[
                            "message"] == "The first and last positions in
LinearRingmust be equivalent":
                            geom['coordinates'][0][-1] =
geom['coordinates'][0][0]
                        new_centers.append(ellipse[2])
                        feat = Feature(geometry=geom, properties={"id1": 0})
                        featurearray.append(feat)
                    pass
                except:
                    # remove the entry if invalid
                    centers_arg.pop(centers_arg.index(center_i))
                    print "fail"
                    # Continue to next iteration.
                    continue
            # create final geojson object

            shape_geojson = geojson.dumps(FeatureCollection(featurearray,
crs=crs), sort_keys=True)
            return shape_geojson, new_centers

        ellipse_geojson, new_centers = generateEllipsegeojson(centers_arg)

        # if we have reached the end of centers so that we are not getting any
valid polygons
        # signal the exit of this loop and subsequent iterations
        if len(json.loads(ellipse_geojson)['features']) == 0:
```

```
            centers1 = []
            return 0, 0

        layer2 = QgsVectorLayer(ellipse_geojson, "json", "ogr")
        layer2.setCrs(QgsCoordinateReferenceSystem(5070,
QgsCoordinateReferenceSystem.EpsgCrsId))
        temp_ellipsename = 'temp_ellipse_' + str(iteration) + '.shp'

        # remove shapefiles that might be already there

        # deleteshapefile(temp_ellipsename)

        # dissolve ellipses to single to reduce computation for overlapped
areas
        QgsGeometryAnalyzer().dissolve(layer2, temp_ellipsename,
onlySelectedFeatures=False, uniqueIdField=-1, p=None)

        # ellipse_shp = QgsVectorLayer(temp_ellipsename, "ellipse", "ogr")

        # create projection file for each shapefile created
        createprojection(temp_ellipsename)
        # del ellipse_shp

        #
QgsVectorFileWriter.writeAsVectorFormat(QgsVectorLayer("c:\\fire_simulation\\da
ta\\ellipse_wgs.shp", "json", "ogr"), "export", "utf-8", None, 'GeoJson')
        # geo_final_str=open("export","r").read()
        wgs_ellipsename = "wgs_" + temp_ellipsename
        ees = QgsVectorLayer(temp_ellipsename, "sd", "ogr")
        processing.runalg("qgis:reprojectlayer", ees, "EPSG:4326",
wgs_ellipsename)
        del ees
        geoj_filename = "wgs_" + str(iteration) + ".geojson"
        # store geojson in htdocs unigis/data

        geoj_filepath = "C:\\xampp\\htdocs\\unigis\\data\\"
        QgsVectorFileWriter.writeAsVectorFormat(QgsVectorLayer(wgs_ellipsename,
"json", "ogr"),
                                                geoj_filepath + geoj_filename,
"utf-8", None, 'GeoJson',

layerOptions=['COORDINATE_PRECISION=6'])

        f_geo = open(geoj_filepath + geoj_filename, "r+")
        rr = (f_geo.read()).replace('\n', ' ')

        f_geo.seek(0)
        f_geo.write(rr)
        f_geo.close()

        # print geo_final_str
        # print geojson.dumps(geo_final_str)
        return [new_centers, geoj_filename]

    return_values = []
    for iter in range(simulation_steps):
        # run the simulation specified number of times and exit whenever we get
signal from loop
        if new_centers == 0:
            break
        else:
            if iter == 0:
                new_centers, geo_final_str = runsimulation(centers1, iter)
                return_values.append([iter, geo_final_str])
                # print new_center
            else:
                # print new_centers
                new_centers, geo_final_str = runsimulation(new_centers, iter)
```

```
                return_values.append([iter, geo_final_str])
    QgsMapLayerRegistry.instance().removeAllMapLayers()
    # remove_tempfiles()
    # ees = QgsVectorLayer("c:\\fire_simulation\\data\\ellipse_wgs1.shp", "sd",
"ogr")
    # processing.runalg("qgis:reprojectlayer", ees, "EPSG:4326",
"c:\\fire_simulation\\data\\wgs.shp")
    #
    # del ees
    return return_values
    def remove_tempfiles():
        for f in os.listdir(os.getcwd()):
            if fnmatch.fnmatch(f, "temp_*"):
                os.remove(f)
                print "To be deleted: " + f
geojson_str, simulation_time, simulation_steps = sys.argv[1], int(sys.argv[2]),
int(sys.argv[3])
if __name__ == "__main__":
    return_values = simulator(geojson_str, simulation_time, simulation_steps)
    print return_values
```

Appendix -C5

Script name: postgres-indexer/__init__.py

```
import psycopg2

def index_postgres(simulation_steps):
    """
    updates index in postgres database after simulation
    Parameters:
    -----------------
    simulation_steps: number of times the simulation as run
    Returns:
    --------------------
    n/a
    """
    conn_string = "host='localhost' dbname='postgis_sample' user='postgres'
password='admin'"
    # get a connection, if a connect cannot be made an exception will be raised
here
    conn = psycopg2.connect(conn_string)
    # print conn
    # conn.cursor will return a cursor object, you can use this cursor to
perform queries
    cursor = conn.cursor()
    deletesql = 'TRUNCATE TABLE "Rasters"'
    cursor.execute(deletesql)
    conn.commit()
    insertsql = ''
    for i in range(1, simulation_steps):
        insertsql += """INSERT INTO \"Rasters\"(the_geom, location, ingestion)
values('0103000020E6100000010000000500000603D3871D44B59C0BC9C3E296D963D40603D3
871D44B59C08C2ACBAA6F4E49402A07C1C685FC53C08C2ACBAA6F4E49402A07C1C685FC53C0BC9C
3E296D963D40603D3871D44B59C0BC9C3E296D963D40', 'Rasters_201601""" + str(
            i).zfill(2) + """.tif', '2016-01-""" + str(i).zfill(2) + """
00:00:00');"""

    cursor.execute(insertsql)
    conn.commit()
    conn.close()

index_postgres(10)
```

Appendix -C6

Script name: shapefile_methods/__init__.py

```
import os
def deleteshapefile(filename):
    """deletes shapefile given file name"""
    name, ext1 = filename.split(".")
    exts = ['shp', 'shx', 'sbx', 'prj', 'dbf', 'qpj']
    for ext in exts:
        if os.path.exists(name + "." + ext):
            os.remove(name + "." + ext)
def createprojection(filename):
    """creates projection for shapefile"""
    prjfile = filename.split(".")[0] + ".prj"
    f = open(prjfile, "w")
    f.write(
'PROJCS["USA_Contiguous_Albers_Equal_Area_Conic_USGS_version",GEOGCS["GCS_North
_American_1983",DATUM["D_North_American_1983",SPHEROID["GRS_1980",6378137.0,298
.257222101]],PRIMEM["Greenwich",0.0],UNIT["Degree",0.0174532925199433]],PROJECT
ION["Albers"],PARAMETER["false easting",0.0],PARAMETER["false northing",0.0],PA
RAMETER["central_meridian",-
96.0],PARAMETER["standard_parallel_1",29.5],PARAMETER["standard_parallel_2",45.
5],PARAMETER["latitude_of_origin",23.0],UNIT["Meter",1.0]]')
    f.close()
```

Appendix - C7

Script name: run.php

```
    <?php
error_reporting(-1);
      $geojson= json_encode($_POST['geojson']);
      $simulation_time= intval($_POST['simulation_time']);
      $simulation_steps= intval($_POST['simulation_steps']);
                //echo $geojson;
      echo exec("python
C:\\Users\\bishrant\\PycharmProjects\\firesimulation\\run_simulation.py
$geojson $simulation_time $simulation_steps");

                          ?>
```

Appendix – C8

Script name: getgeojson.php

```
<?php
error_reporting(-1);

$geojson_id= intval($_GET['geojson_id']);
$data = file_get_contents(".\data\\$geojson_id.geojson");
echo $data

?>
```

Appendix -C9

Script name: index2.html

```
<!DOCTYPE html>
<html>
<head>
    <meta charset="utf-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
```

```
    <meta name="viewport" content="width=device-width, initial-scale=1, user-
scalable=no" />
    <link rel="stylesheet" href="./js/theme/default/style.css" type="text/css"
/>
    <link rel="stylesheet" href="./fontawesome/css/font-awesome.min.css"
type="text/css" />
    <script src="./js/OpenLayers.js"></script>
    <link rel="stylesheet"
href="//maxcdn.bootstrapcdn.com/bootstrap/3.2.0/css/bootstrap.min.css" />
    <link rel="stylesheet" href="//maxcdn.bootstrapcdn.com/font-
awesome/4.2.0/css/font-awesome.min.css" />
    <link rel="stylesheet"
href="//code.jquery.com/ui/1.12.1/themes/base/jquery-ui.css">
    <script src="http://maps.google.com/maps/api/js?v=3&key=AIzaSyAomiv-
9YE9j4S8fJCEQ33OxqXy5Poy5zk"></script>
    <title>WebGIS based Wildfire Simulation </title>
    <style type="text/css">
        body {
            overflow: hidden;
        }
        .navbar-offset {
            margin-top: 50px;
        }
        #map {
            position: absolute;
            top: 50px;
            bottom: 0px;
            left: 0px;
            right: 0px;
        }
            #map .ol-zoom {
                font-size: 1.2em;
            }
        .zoom-top-opened-sidebar {
            margin-top: 5px;
        }
        .zoom-top-collapsed {
            margin-top: 45px;
        }
        .mini-submenu {
            display: none;
            background-color: rgba(255, 255, 255, 0.46);
            border: 1px solid rgba(0, 0, 0, 0.9);
            border-radius: 4px;
            padding: 9px;
            /*position: relative;*/
            width: 42px;
            text-align: center;
        }
        .mini-submenu-left {
            position: absolute;
            top: 60px;
            left: .5em;
            z-index: 40;
        }
        #map {
            z-index: 35;
        }
        .sidebar {
            z-index: 45;
        }
        .main-row {
            position: relative;
            top: 0;
        }
        .mini-submenu:hover {
            cursor: pointer;
        }
```

```
        .slide-submenu {
            background: rgba(0, 0, 0, 0.45);
            display: inline-block;
            padding: 0 8px;
            border-radius: 4px;
            cursor: pointer;
        }
        #slider label {
            position: absolute;
            width: 20px;
            margin-top: 20px;
            margin-left: -10px;
            text-align: center;
        }
        div {
            border-radius: 10px;
        }
    </style>
    <script type="text/javascript" src="//code.jquery.com/jquery-
2.1.1.min.js"></script>
    <script type="text/javascript"
src="//maxcdn.bootstrapcdn.com/bootstrap/3.2.0/js/bootstrap.min.js"></script>
    <script src="https://code.jquery.com/ui/1.12.1/jquery-ui.js"></script>
    <script type="text/javascript">
        var formats
        function updateFormats() {
            in_options = {
                'internalProjection': map.baseLayer.projection,
                'externalProjection': new OpenLayers.Projection("EPSG:4326")
            };
            out_options = {
                'internalProjection': map.baseLayer.projection,
                'externalProjection': new OpenLayers.Projection("EPSG:4326")
            };
            formats = {
                'in': { geojson: new OpenLayers.Format.GeoJSON(in_options) },
                'out': { geojson: new OpenLayers.Format.GeoJSON(out_options) }
            };
        }
      function init(){
          var defaultstylemap = new OpenLayers.StyleMap({
              'strokeWidth': 1,
              fillColor: "#7C7C7F"
          });
          var styleMap = new OpenLayers.StyleMap({
              'strokeWidth': 1
          });
          var lookup = {
              2000.0: { fillColor: "#FF3C00" },
              1800.0: { fillColor: "#800026" },
              1600.0: { fillColor: "#bd0026" },
              1400.0: { fillColor: "#e31a1c" },
              1200.0: { fillColor: "#fc4e2a" },
              1000.0: { fillColor: "#fd8d3c" },
              800.0: { fillColor: "#feb24c" },
              600.0: { fillColor: "#fed976" },
              400.0: { fillColor: "#ffeda0" },
              200.0: { fillColor: "#ffffcc" },
              0: { fillColor: "#7C7C7F" }

          }
          styleMap.addUniqueValueRules("default", "Hour", lookup);
          var styleMap_1 = new OpenLayers.StyleMap({
              'strokeWidth': 1
          });
          var lookup_1 = {
              1000.0: { fillColor: "#FF3C00" },
              900.0: { fillColor: "#800026" },
```

```
              800.0: { fillColor: "#bd0026" },
              700.0: { fillColor: "#e31a1c" },
              600.0: { fillColor: "#fc4e2a" },
              600.0: { fillColor: "#fd8d3c" },
              500.0: { fillColor: "#feb24c" },
              300.0: { fillColor: "#fed976" },
              200.0: { fillColor: "#ffeda0" },
              100.0: { fillColor: "#ffffcc" },
              0: { fillColor: "#7C7C7F" }

          }
          styleMap_1.addUniqueValueRules("default", "Hour", lookup_1);

          map = new OpenLayers.Map('map');

        osm = new OpenLayers.Layer.OSM("Simple OSM Map");
          // the SATELLITE layer has all 22 zoom level, so we add it first to
          // become the internal base layer that determines the zoom levels of
the
          // map.
        gsat = new OpenLayers.Layer.Google(
           "Google Satellite",
           { type: google.maps.MapTypeId.SATELLITE, numZoomLevels: 22,
isBaseLayer: true }
       );
        var gphy = new OpenLayers.Layer.Google(
            "Google Physical",
            { type: google.maps.MapTypeId.TERRAIN, visibility: false }
        );
        var gmap = new OpenLayers.Layer.Google(
            "Google Streets", // the default
            { numZoomLevels: 20, visibility: false }
        );
        ghyb = new OpenLayers.Layer.Google(
           "Google Hybrid",
           { type: google.maps.MapTypeId.HYBRID, numZoomLevels: 22, visibility:
false, isBaseLayer: true }
       );

        vectors = new OpenLayers.Layer.Vector("Simulation Layer", { projection:
map.displayProjection, styleMap: defaultstylemap, rendererOptions: { zIndexing:
true } });
        map.addLayers([gsat, gphy, gmap, ghyb, vectors]);
        map.setCenter(
            new OpenLayers.LonLat(-108, 43)
            .transform(
                new OpenLayers.Projection("EPSG:4326"),
                map.getProjectionObject())
             , 7
        );
        map.addControl(new OpenLayers.Control.MousePosition());
        map.addControl(new OpenLayers.Control.LayerSwitcher({ 'div':
OpenLayers.Util.getElement('layerswitcher') }));
        pol = new OpenLayers.Control.DrawFeature(vectors,
OpenLayers.Handler.Path)
        map.addControl(pol)
        select = new OpenLayers.Control.SelectFeature(vectors, { hover: true,
onSelect: serialize });
        map.addControl(select);
        select.activate();
        updateFormats();
        //$("#ouput").dialog();
          // Attach a submit handler to the form
        $("#inputform").submit(function (event) {
            // Stop form from submitting normally

            select.deactivate();
            var $form = $(this),
```

```
             geojson_str = document.getElementById('geojson').value,
             simulation_time = $('#simulation_time').val(),
             simulation_steps = $('#simulation_steps').val(),
             url = $form.attr("action");
            $.ajax({
                type: 'POST', // define the type of HTTP verb we want to use
(POST for our form)
                url: url, // the url where we want to POST
                data: { geojson: geojson_str, simulation_time:simulation_time,
simulation_steps:simulation_steps  }, // our data object
               // dataType: 'json', // what type of data do we expect back from
the server
                // encode: true
                async:false,
                success: function (e0) {
                    if (simulation_time == 2) { vectors.styleMap = styleMap; }
                    else { vectors.styleMap = styleMap_1; }
                    python_result = e0;
                    //deserialize(e0)
                    fetchgeojson(0,0);
                    $("#toolbar").show();
                    createlegend();
                }
            })
            console.log([0, simulation_time * simulation_steps,
simulation_time, 0])
            createslider(0, simulation_time * simulation_steps,simulation_time,
0)
            // stop the form from submitting the normal way and refreshing the
page
            event.preventDefault();
        });

      };
      function fetchgeojson(current_step) {
          geojson_url = "http://localhost:81/unigis/getgeojson.php";
          geojson_time=parseInt($("#simulation_time").val());
          $.ajax({
              type: 'GET', // define the type of HTTP verb we want to use (POST
for our form)
              url: geojson_url, // the url where we want to POST
              data: { geojson_id:current_step/geojson_time }, // our data
object
              // dataType: 'json', // what type of data do we expect back from
the server
              // encode: true
              async: false,
              success: function (geoj) {
                  current_geoj = geoj;
                  deserialize(geoj, current_step);
                  //fetchgeojson(0);
              }
          })
      }
        slider1=0
      function createslider(min, max, step, current) {
          if (slider1) { $("#slider").slider('destroy'); $('#slider').empty()
};
          slider1= $("#slider").slider({
              value: parseFloat(current),
              min: parseFloat(min),
              max: parseFloat(max),
              step: parseFloat(step),
              slide: function (event, ui) {
                  //fetchgeojson(ui.value);
              },
              change: function (event, ui) {
                  $("#slidervalue").html("Current Time (Hrs):" + ui.value);
```

```
                    fetchgeojson(ui.value);
                    console.log(ui.value);
                },
                slidestart: function (event, ui) {
                    //fetchgeojson(ui.value);
                },
            });
                // Get the number of possible values
                vals = (max - min)/step;
                // Position the labels
                for (var i = 0; i <= vals; i++) {
                    // Create a new element and position it with percentages
                    var el = $('<label>' + (i + min)*step +
'</label>').css('left', (i / vals * 100) + '%');
                    // Add the element inside #slider
                    $("#slider").append(el);
                }
        }
        function serialize(feature) {
            var type = "geojson";
            var str = formats['out'][type].write(feature, false);
            // not a good idea in general, just for this demo
            str = str.replace(/,/g, ', ');
            document.getElementById('geojson').value=str
            console.log(str)
        }
        function createpoly() {
            //vectors.destroyFeatures();
            document.getElementById('output').value = '';
            pol.activate()
        }
        function submitpoly() {
            pol.deactivate()
            console.log(document.getElementById('output').value)
        }
        function deserialize(geojson_str,step) {
            console.log(geojson_str);
            vectors.removeAllFeatures();
            var type = "geojson";
            var features = formats['in'][type].read(geojson_str);
            var bounds;
            if (features) {
                if (features.constructor != Array) {
                    features = [features];
                }
                for (var i = 0; i < features.length; ++i) {
                    if (!bounds) {
                        bounds = features[i].geometry.getBounds();
                    } else {
                        bounds.extend(features[i].geometry.getBounds());
                    }
                }
                var idx = vectors.features.length;
                vectors.addFeatures(features);
                vectors.features[idx].id = step;
                map.zoomToExtent(bounds);
            }
        }
        $(function () {
            $("#playpause").click(function () {
                if ($("#playpause").prop("value") == "Play") {
                    hoursInterval = setInterval(function () {
                        if (parseInt($("#slider").slider("value")) >=
parseInt($("#simulation_time").val()) * parseInt($("#simulation_steps").val()))
{
                            clearInterval(hoursInterval);
                        }
                        else {
```

```
                        var ss = parseInt($("#slider").slider("value")) +
parseInt($("#simulation_time").val());
                        console.log(ss);
                        $("#slider").slider("value", ss);
                    }
                }, 1000);
                $("#playpause").prop("value", "Pause");
            }
            else {
                clearInterval(hoursInterval);
                $("#slider").slider("value",
parseInt($("#slider").slider("value")));
                $("#playpause").prop("value", "Play");
               // $("#slider").slider()
            }
        })

    })
    $(function () {
        $("#animation_start_controls").draggable();
        $("#toolbar").draggable();
        $("#toolbar").hide();
        $("#legend").hide();

    });
    function createlegend() {
        simulation_time = parseInt($('#simulation_time').val());
        simulation_steps = parseInt($('#simulation_steps').val());
        color_chart = ["#7C7C7F", "#ffffcc", "#ffeda0", "#fed976", "#feb24c",
"#fd8d3c", "#fc4e2a", "#e31a1c", "#bd0026", "#800026", "#FF3C00"]
        str = '<b>Legend</b><br/>';
        for (i = 0; i < simulation_steps + 1; i++) {
            str += +simulation_time * i + ' hrs :
                                <svg width="40" height="20">
                                        <rect width="40" height="20"
style="fill:' + color_chart[i] + ';stroke:rgb(0,0,0)">
                                        </svg>';
            str += "
                                </br>";
        }
        document.getElementById('legend').innerHTML = str;
        console.log(str);
        $("#legend").show();
    }
</script>
</head>
<body onload="init()">
    <div class="container">
        <nav class="navbar navbar-fixed-top navbar-default" role="navigation">
            <div class="container-fluid">
                <!-- Brand and toggle get grouped for better mobile display -->
                <div class="navbar-header">
                    <button type="button" class="navbar-toggle" data-
toggle="collapse" data-target="#bs-example-navbar-collapse-1">
                        <span class="sr-only">Toggle navigation</span>
                        <span class="icon-bar"></span>
                        <span class="icon-bar"></span>
                        <span class="icon-bar"></span>
                    </button>
                    <a class="navbar-brand" href="#">WebGIS based Wildfire
Simulation</a>
                </div>
                <!-- Collect the nav links, forms, and other content for
toggling -->
                <div class="collapse navbar-collapse" id="bs-example-navbar-
collapse-1">
                    <!-- <ul class="nav navbar-nav"><li class="active"><a
href="#">Link</a></li><li><a href="#">Link</a></li></ul>-->
```

```
                    <div id="toolbar" style="z-index: 100; position: absolute;
margin-top: 55px; background-color: #F8F8F8; padding: 5px;">
                        <div id="slidervalue"></div>
                        <div id="slider" style="width: 400px;   padding-left:
20px;   margin-left: 20px;    margin-bottom: 30px;"></div>
                        <div syle="margin-left: 100px;    padding-left:
100px;">
                            <button class="btn btn-primary" id="prev">
                                <i class="fa fa-step-backward" aria-
hidden="true"></i>   Prev
                            </button>
                            <button class="btn btn-primary" id="playpause"
value="Play">
                                <i class="fa fa-play" aria-hidden="true"></i>
  Play
                            </button>
                            <button class="btn btn-primary" id="next">
                                <i class="fa fa-step-forward" aria-
hidden="true"></i>   Next
                            </button>
                        </div>
                    </div>
                    <div id="normalcontrols" style="z-index: 100; position:
absolute; margin-top: 5px;
border: 1px solid black; right:10px; background-color: #F8F8F8; padding: 5px;">
                        <button class="btn" id="control_pan" data-
toggle="tooltip" title="Pan">
                            <i class="fa fa-hand-rock-o" aria-
hidden="true"></i>
                        </button>
                        <button class="btn" id="control_fullextent" data-
toggle="tooltip" title="Zoom to full extent">
                            <i class="fa fa-globe" aria-hidden="true"></i>
                        </button>
                        <button class="btn" id="control_prev" data-
toggle="tooltip" title="Zoom to Previous extent">
                            <i class="fa fa-arrow-left" aria-hidden="true"></i>
                        </button>
                        <button class="btn" id="control_next" data-
toggle="tooltip" title="Zoom to Next extent">
                            <i class="fa fa-arrow-right" aria-
hidden="true"></i>
                        </button>
                        <button class="btn" id="control_zoomin" data-
toggle="tooltip" title="Zoom in to your area of choice">
                            <i class="fa fa-search-minus" aria-
hidden="true"></i>
                        </button>
                        <button class="btn" id="control_zoomout" data-
toggle="tooltip" title="Zoom Out from your area of choice">
                            <i class="fa fa-search-plus" aria-
hidden="true"></i>
                        </button>
                        <button class="btn" id="control_identify" data-
toggle="tooltip" title="Identify Features">
                            <i class="fa fa-info-circle" aria-
hidden="true"></i>
                        </button>
                        <button class="btn" id="control_legend" data-
toggle="tooltip" title="Show Legend">
                            <i class="fa fa-th-list" aria-hidden="true"></i>
                        </button>
                        <button class="btn" id="control_help" data-
toggle="tooltip" title="Help and About">
                            <i class="fa fa-question-circle" aria-
hidden="true"></i>
                        </button>
                    </div>
```

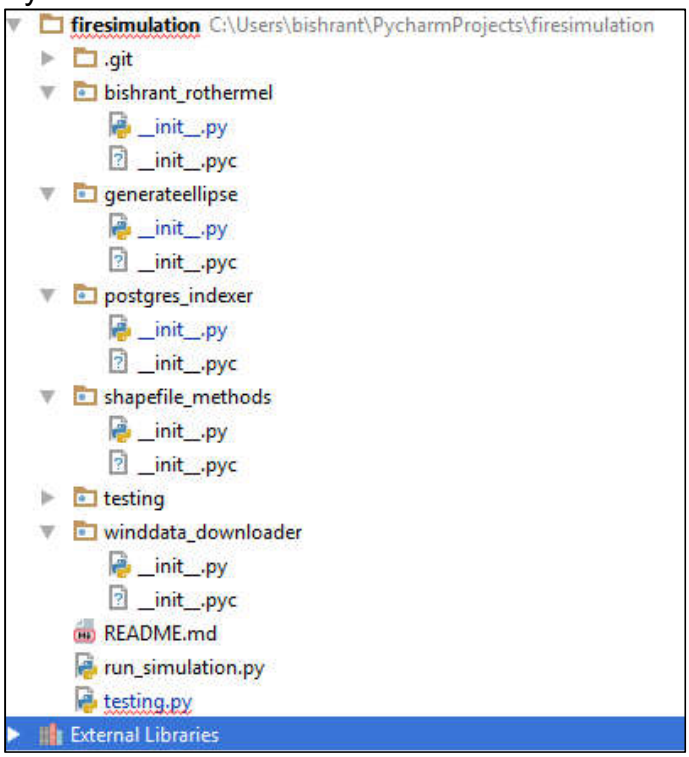
```
                    <div id="animation_start_controls" style="z-index: 100;
position: absolute; margin-top: 55px;
border: 1px solid black; right:40px; background-color: #F8F8F8; padding: 5px;">
                        <form action="http://localhost:81/unigis/run.php"
id="inputform" method="post" class="navbar-form navbar-left">
                            <input type="button" class="form-control"
placeholder="Create Polygon" onclick="createpoly()"
                                   value="   Fire Start
Location"
                                   style="background-image: url(./assets/mono-
edit-polygon.png);
    background-size: 20px;
    background-repeat: no-repeat;
    background-position: 0px;
    " />

                            Time(hrs):
                            <input type="text" style="width:40px"
id="simulation_time" name="simulation_time" placeholder="Time(hrs)" value="" />
                            Steps:
                            <input type="text" style="width:40px"
id="simulation_steps" name="simulation_steps" placeholder="Steps" value="" />
                            <input type="text" id="geojson" hidden="hidden"
name="geojson" value='' />
                            <input type="submit" class="btn btn-primary"
value="      Simulate"
                                   style="background-image:
url(./assets/send.png);
    background-size: 20px;
    background-repeat: no-repeat;
    background-position: 0px;
    " />
                        </form>
                    </div>
                </div>
                <!-- /.navbar-collapse -->
            </div>
            <!-- /.container-fluid -->
        </nav>
    </div>
    </nav>
    <div class="navbar-offset"></div>
    <div id="map"></div>
    <div class="mini-submenu mini-submenu-left pull-left">
        <i class="fa fa-list-alt"></i>
    </div>
    <div id="output" title="Basic dialog" style="left:0"></div>
    <div id="layerswitcher" style="z-index: 100; position: absolute; bottom:
25px; left: 10px;
              border: 1px solid black;  background-color: #F8F8F8; padding:
5px;"><img src="./assets/north.png" width="30" height="40" />
    </div>
    <div id="legend" style="z-index: 100; position: absolute; bottom: 80px;
right: 5px; width: 90px;
              border: 1px solid black;  background-color: #F8F8F8; padding:
5px; font-size:xx-small"></div>
    <div id="disclaimer" style="z-index: 100; position: absolute; bottom: 15px;
right: 5px; width: 300px;
              border: 1px solid black;  background-color: #F8F8F8; padding:
5px; font-size:xx-small">
        <strong>Disclaimer:  </strong>This WebGIS based wildfire
simulation system is created for the partial fulfillment of Msc UNIGIS program
and might not reflect the exact fire conditions and it should be used at user's
own discretion.
        <strong>Developed By: Bishrant Adhikari (GIS_104195) </strong>
    </div>
</body>
</html>
```

Appendix -C10

Hierarchy and arrangement of script files

Python



Apache Server:

Base path: c:\xampp\htdocs\unigis\

```
assets
data
fontawesome
js
getgeojson.php
run.php
index2.html
```

Appendix -C11

Deployment Machine and software specifications:

```
OS Name  Microsoft Windows 10 Enterprise Version    10.0.14393 Build 14393
System Model      Inspiron 5521
System Type       x64-based PC
Processor         Intel(R) Core(TM) i7-3537U CPU @ 2.00GHz, 2501 Mhz, 2 Core(s),
4 Logical Processor(s)
Time Zone         Mountain Standard Time
Installed Physical Memory (RAM)    16.0 GB
Total Physical Memory     15.9 GB
Total Virtual Memory      18.3 GB
Hard Disk              512GB SSD Drive
Python Version  2.7.2
Apache Friends XAMPP Version 5.6.28
PHP Version 5.6.28
Host http://localhost (127.0.0.1)
Port 81
Postgres 9.2
Postgis 2.0.1
```

Appendix D

This appendix shows the static images of time series based animation for sample site 3 chosen in Chapter 5 of this thesis.

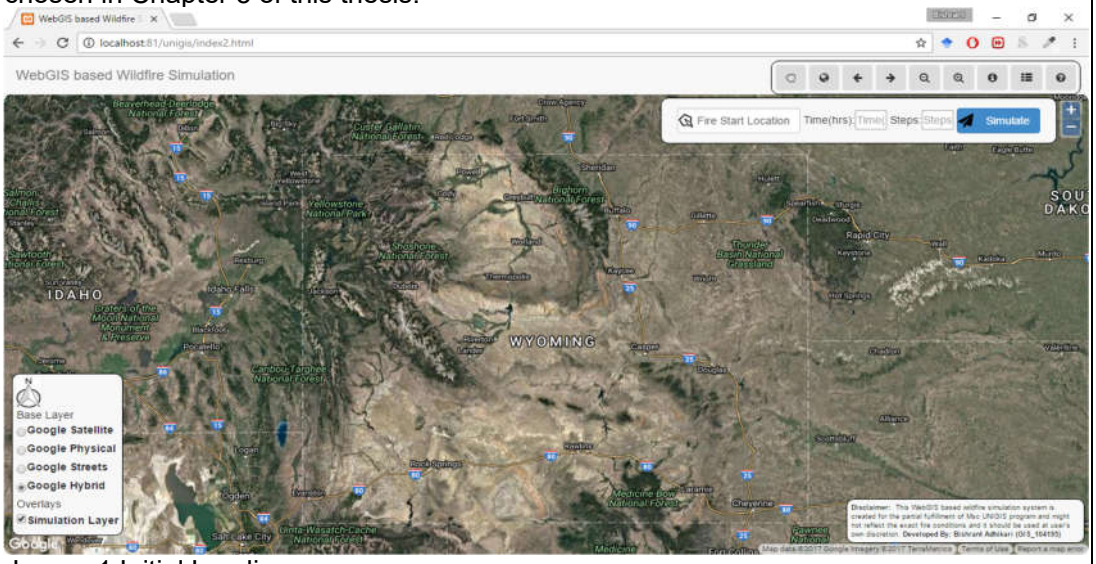
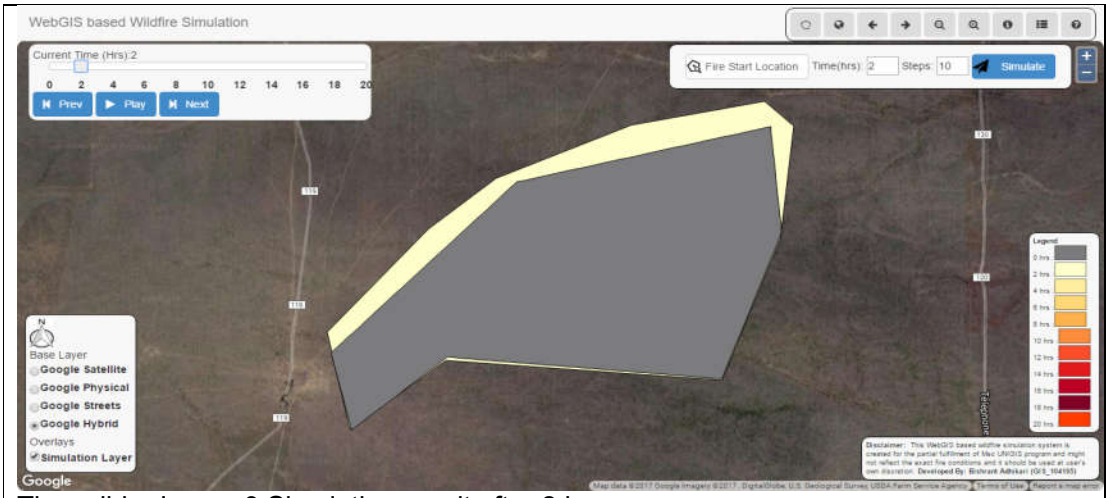


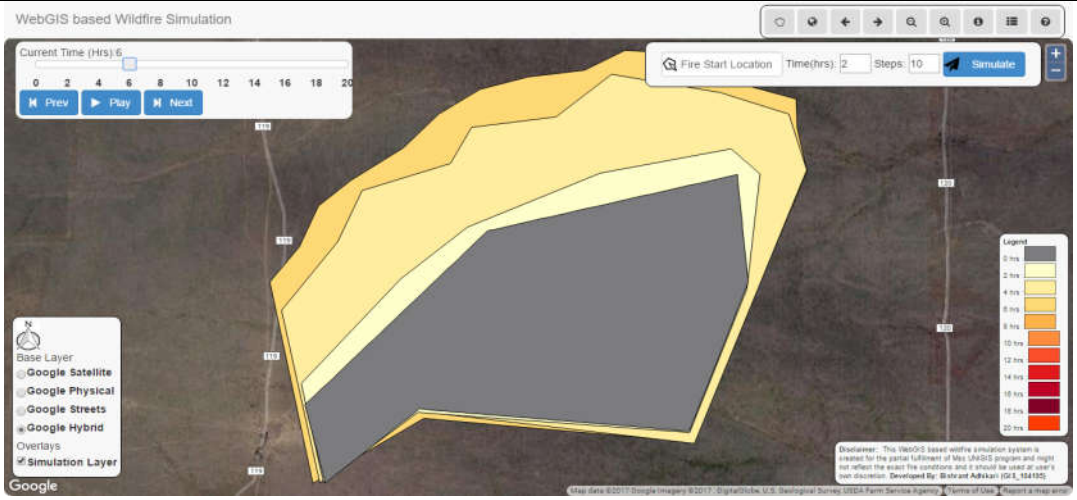
Image 1 Initial Loading

Time slider Image 2 Digitizing input area and specifying simulation time and steps

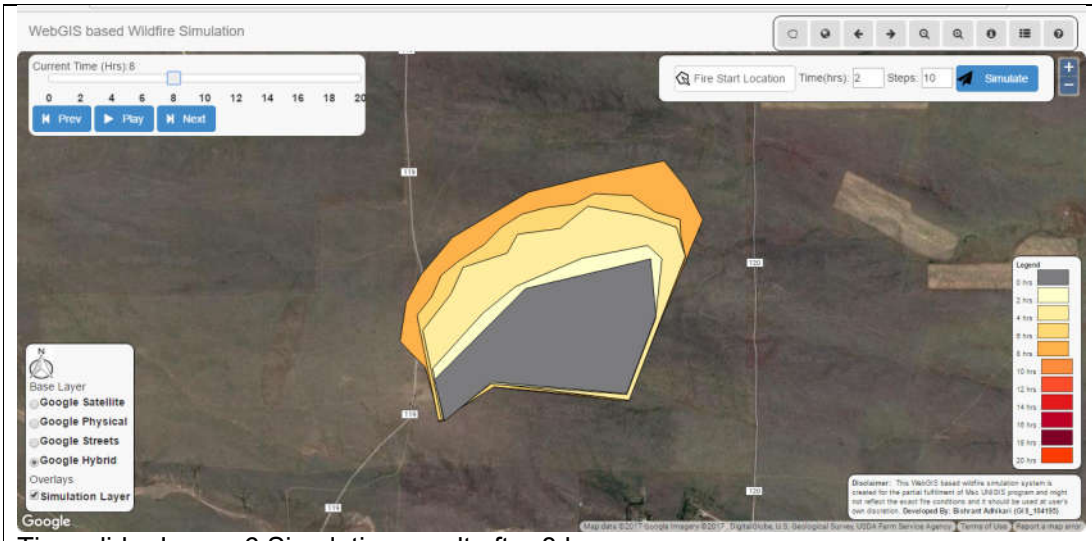


Time slider Image 3 Simulation result after 2 hours

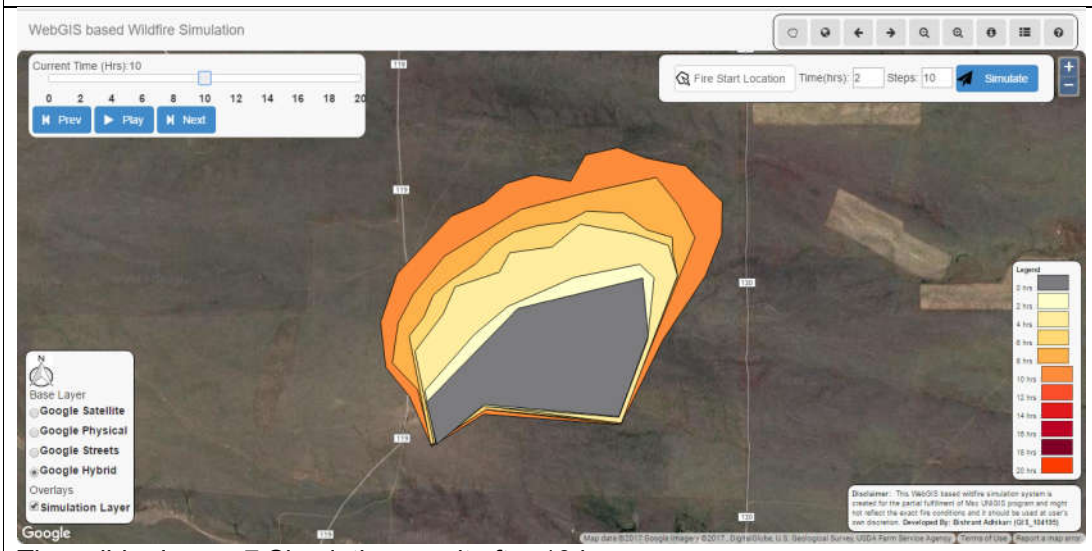
Time slider Image 4 Simulation result after 4 hours



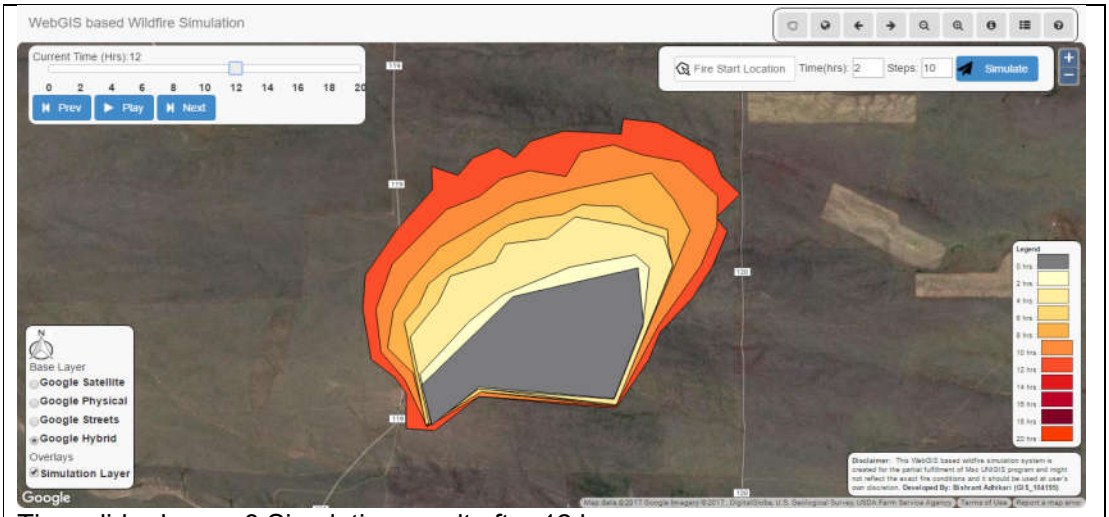
Time slider Image 5 Simulation result after 6 hours



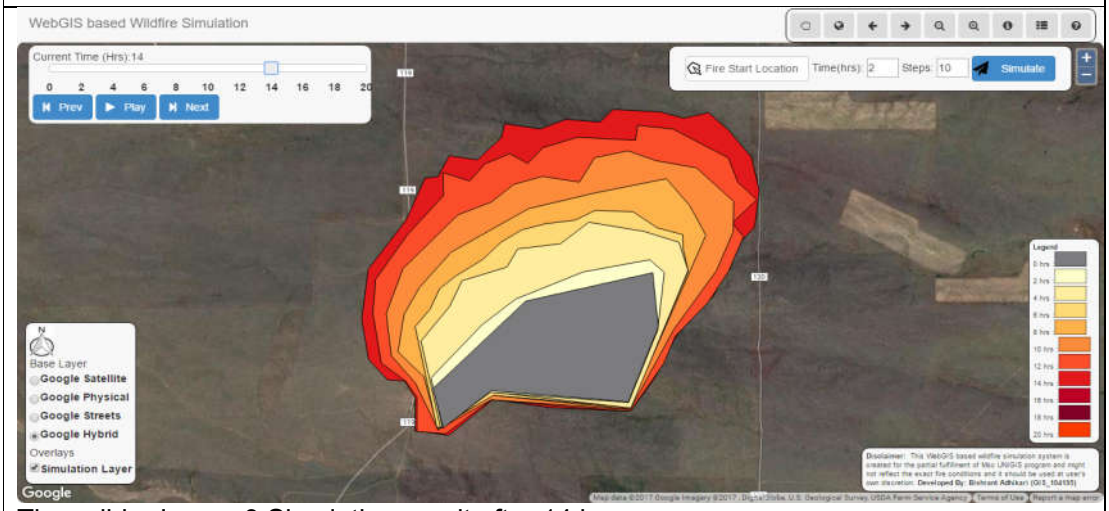
Time slider Image 6 Simulation result after 8 hours



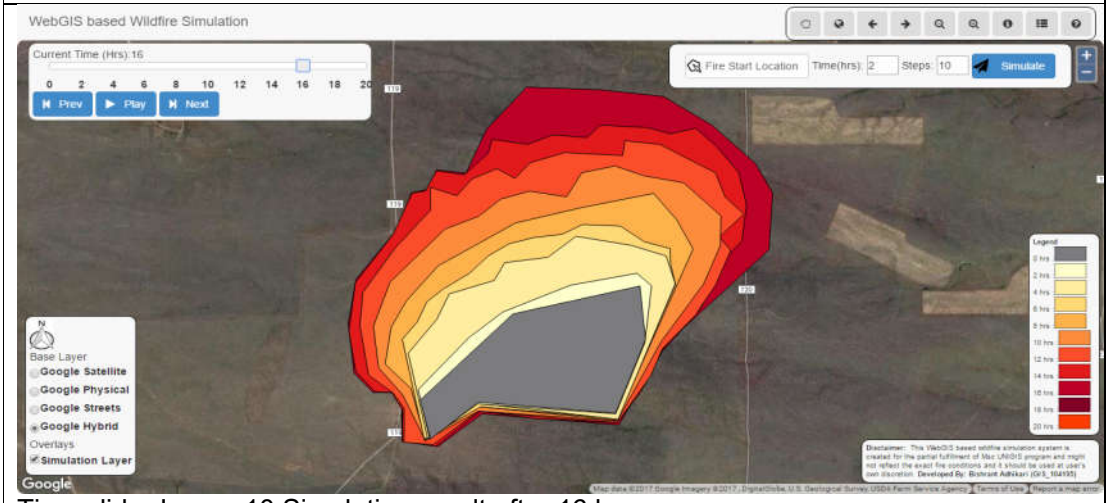
Time slider Image 7 Simulation result after 10 hours



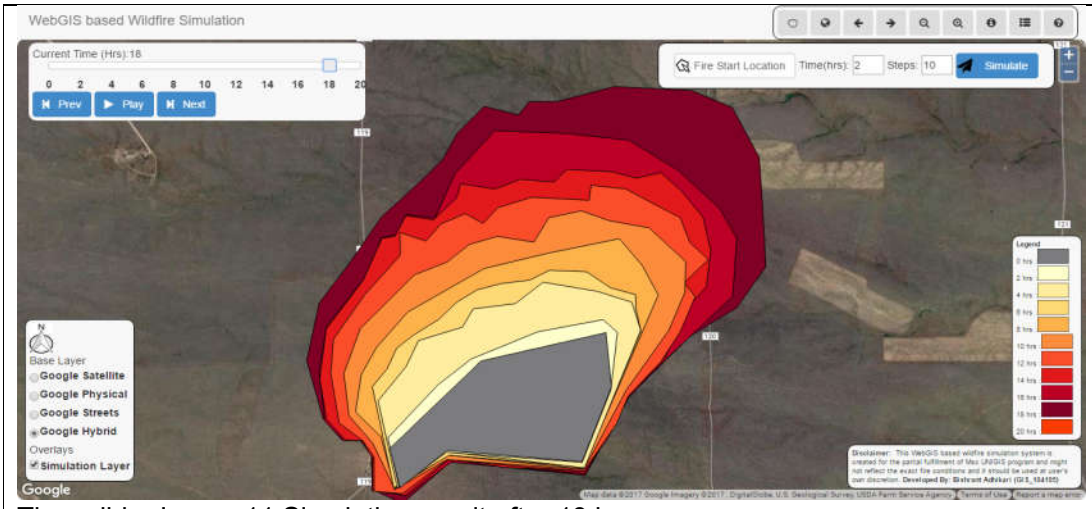
Time slider Image 8 Simulation result after 12 hours



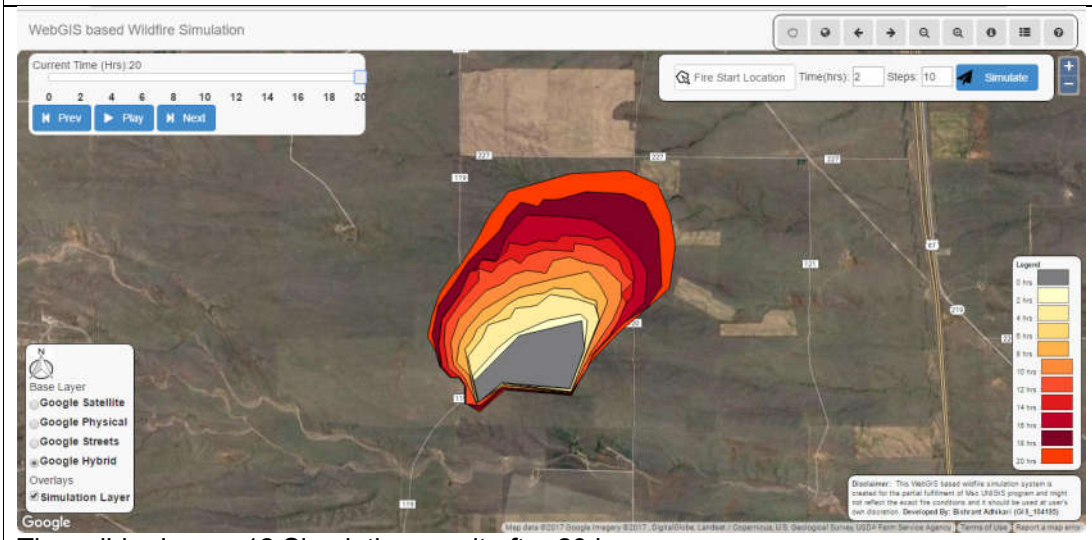
Time slider Image 9 Simulation result after 14 hours



Time slider Image 10 Simulation result after 16 hours



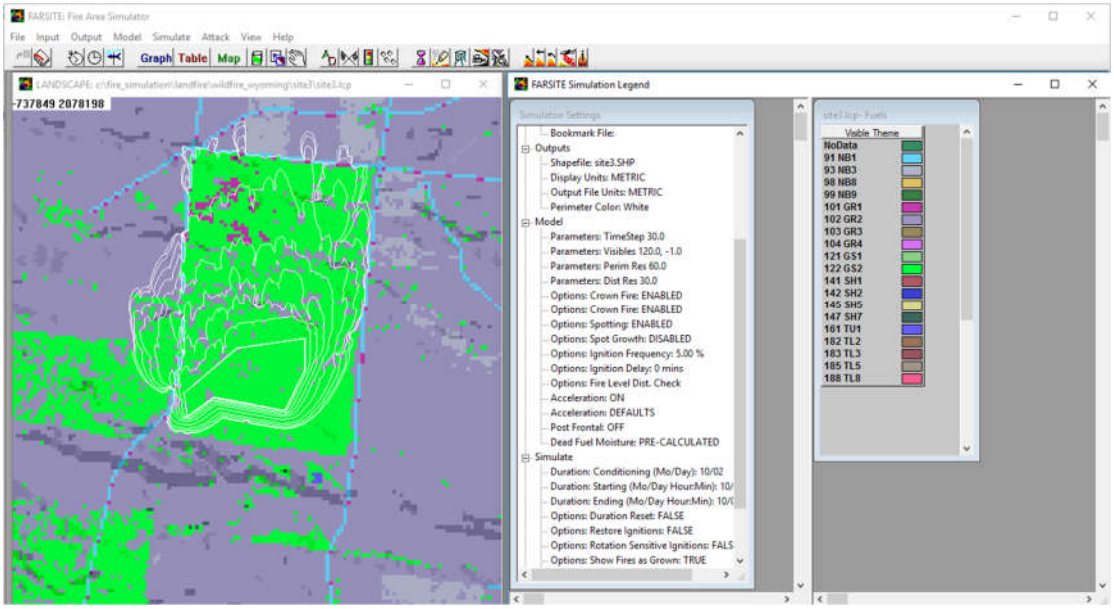
Time slider Image 11 Simulation result after 18 hours



Time slider Image 12 Simulation result after 20 hours

Appendix E

FARSITE Result for site-3:


FARSITE: Fire Area Simulator
File Input Output Model Simulate Attack View Help
Graph Table Map
-737849 2078198
FARSITE Simulation Legend
Bookmark File:
Outputs
Shapefile: site3.SHP
Display Units: METRIC
Output File Units: METRIC
Perimeter Color: White
Model
Parameters: TimeStep 30.0
Parameters: Visibles 120.0, -1.0
Parameters: Perim Res 60.0
Parameters: Dist Res 30.0
Options: Crown Fire: ENABLED
Options: Crown Fire: ENABLED
Options: Spotting: ENABLED
Options: Spot Growth: DISABLED
Options: Ignition Frequency: 5.00 %
Options: Ignition Delay: 0 mins
Options: Fire Level Dist. Check
Acceleration: ON
Acceleration: DEFAULTS
Post Frontal: OFF
Dead Fuel Moisture: PRE-CALCULATED
Simulate
Duration: Conditioning (Mo/Day): 10/02
Options: Duration Reset: FALSE
Options: Restore Ignitions: FALSE
Options: Rotation Sensitive Ignitions: FALS
Options: Show Fires as Grown: TRUE
Visible Theme
NoData
91 NB1
93 NB3
98 NB8
99 NB9
101 GR1
102 GR2
103 GR3
104 GR4
121 GS1
122 GS2
141 SH1
142 SH2
145 SH5
147 SH7
161 TU1
182 TL2
183 TL3
185 TL5
188 TL8