

**Problem Chosen**

**2022**

**Team Control Number**

**D**

**MCM/ICM  
Summary Sheet**

**00000000**

---

### **Summary**

Australia is undergoing huge wildfires in every state. To protect people and safety and property, we establish a model to use two types of drones to help Country Fire Authority (CFA) conduct "Rapid Bushfire Response", and front-line personnel communicate with Emergency Operations Center (EOC).

We use Victoria Fire Report Data, using fire report places in certain time period to represent locations where fires happened on average. This allows us to take into account of factors such as fire size and frequency, economic cost and safety, weighted region area covered by drones.

We set Fast Response Model for deployment of drones for fast response, we quantify the coverage with a weighted quantity. We build a model to investigate its' relationship with the economic cost. For a certain coverage by drone, we devise a strategy to distribute SSAs by using k-means with special distance function and we use minimum spanning tree to distribute repeaters to connect them. We show the mathematical properties to such distance to ensure the correctness of our algorithm.

We set Fire Prediction Model for second part of problem. we divide Victoria into several zones, and use statistics of zones in different time to form a time series. We predict the time series using convolutional Long Short-Term Memory (convLSTM).

We set Pearl Model and Spur Model for Deployment of drones for front-line personnel in different circumstances, we use separate deployment strategies for small and big sized fire considering the effect of terrain.

The sensitivity analysis shows robustness in our model. Meanwhile, we combine all the models to finish the annotated Budget Request to help CFA with acceptable cost.

**Keywords:** Clustering, Minimum Spanning Tree, ConvLSTM, Terrian;

# Contents

<b>1</b>	<b>Introduction</b>	<b>2</b>
1.1	Background . . . . .	2
1.2	Problem Restatement . . . . .	2
1.2.1	Limits . . . . .	2
1.2.2	Targets . . . . .	2
1.3	Our Work . . . . .	3
<b>2</b>	<b>List Of Symbols</b>	<b>3</b>
<b>3</b>	<b>General Assumptions</b>	<b>4</b>
<b>4</b>	<b>The Models</b>	<b>4</b>
<b>5</b>	<b>Conclusions</b>	<b>5</b>
<b>6</b>	<b>Model Evaluation And Improvement</b>	<b>5</b>
6.1	Strength . . . . .	5
6.2	Weakness . . . . .	5
6.3	Improvement . . . . .	5
	<b>Appendices</b>	<b>7</b>
	<b>Appendix A Code for map plotting</b>	<b>7</b>

# 1 Introduction

## 1.1 Background

Wildfire spreads rapidly in Australia. In fire season, it's devastating for people's safety and properties. Victoria's Country Fire Authority (CFA) uses different means to protect its people. Drones carrying high definition & thermal imaging cameras and telemetry sensors were sent for surveillance and situational awareness (SSA). Drone repeaters, transceivers that automatically rebroadcast signals at higher powers can help connect Emergency Operations Center (EOC) with SSA and front-line employees with VHF/UHF bands.

## 1.2 Problem Restatement

### 1.2.1 Limits

- Drone-related
  - Cost \$1000 per drone
  - Flight Range 30 km
  - Transmission Range 20 km
  - Flight Speed 20 m/s
- Fire-related
  - Size
  - Frequency

### 1.2.2 Targets

- Deployment of drones for fast response
  - Reduce cost
  - Increase weighted coverage
- Fire Prediction

- Deployment of drones for front-line personnel in different circumstances
  - Build models for different fire size
  - Build models considering different terrains

### 1.3 Our Work

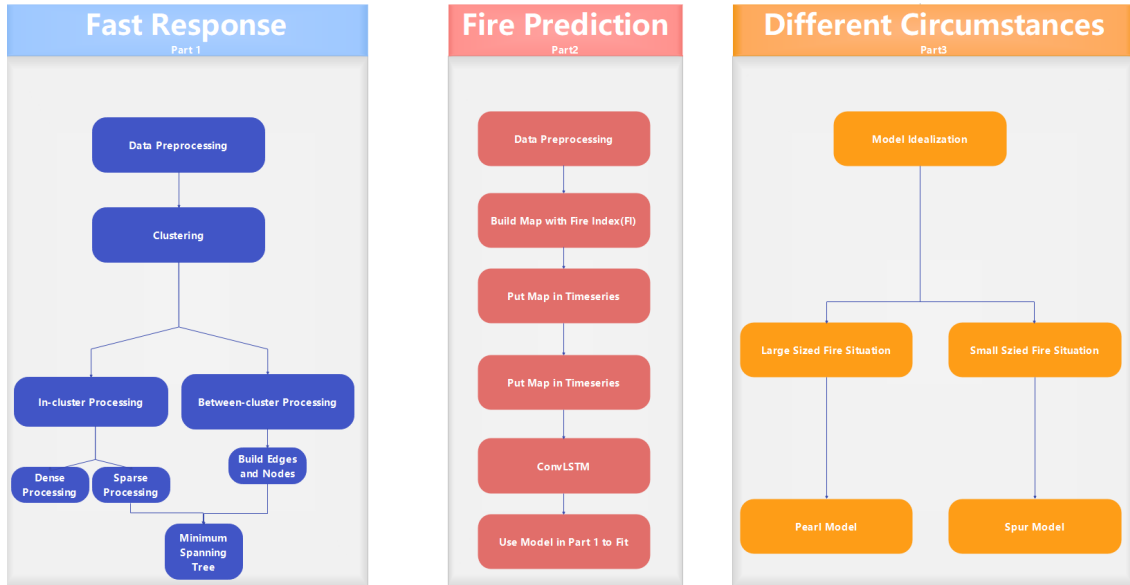


Figure 1: Our Work

## 2 List Of Symbols

Definition	Description
$dist$	Distance between two points in Euclidean coordinate system
$R_e$	Radius of earth
$\Delta\varphi_{lat}$	Thange of latitude
$\Delta\lambda_{lon}$	Thange of longitude
$eps$	Tf the distance between two points is lower or equal to ( $eps$ ), these points are consid
$minPoints$	The minimum number of points to form a dense region.
$x_0$	Distance to drone when the fire is recorded in our data

---

Definition	Description
$x_1$	Distance to drone when drone detect fire
$dis_{j,t}$	The shortest distance of fire location indexed $j$ to the nearest drone at distance $x$ from the drone
$r_c$	The radius of drone in idealized condition.
$vs_{j,x}$	The spread rate of fire at the rim of fire area indexed $j$ at distance $x$ from the drone
$p_{j,x}$	The probability for drones to detect rim of fire location $j$ at distance $x$ from the drone
$v$	Stable spread rate which is assumed to simplify the model
$WCL$	Weighted coverage loss : describing the loss the weighted area for a deployment strategy
$r_o$	Outer-radius of drone, meaning the furthest distance the drone can detect, that is, 500m
$tWCL$	Threshold for $WCL$ to determine how much SSA should be deployed.

---

### 3 General Assumptions

- We use one year data in Victoria with data provided by Earth Data to represent the general cases in Australia. However, our model to this case adapt to arbitrary cases, so it's without losing generality.
- We assume once the fire is within the detective range of drones, it will be found out without delay.
- We assume the spread rate of fire is stable and at a certain value, which is not the case in real world, but one can use the original formulae given in the model to simply modify the model.
- Only 20 years of data is used for machine learning, the error produced is within the acceptable range.
- The terrain situation can be more complicated in real world, we idealize mountain and other barriers as parabolic-like object.

### 4 The Models

;

## **5 Conclusions**

Conclusions

## **6 Model Evaluation And Improvement**

### **6.1 Strength**

### **6.2 Weakness**

### **6.3 Improvement**

## References

# Appendices

## Appendix A Code for map plotting

```
from mpl_toolkits.basemap import Basemap
import matplotlib.pyplot as plt

m = Basemap(projection='mill',llcrnrlat=-60,urcnrlat=90,\
            llcrnrlon=-180,urcnrlon=180,resolution='l')
m.drawcoastlines()
m.drawcountries()
m.drawstates()
m.fillcontinents(color='#04BAE3', lake_color='#FFFFFF')
m.drawmapboundary(fill_color='#FFFFFF')

lat = 30,31,34,33,32
lon = -103,-110,-107,-111,-115

lat2 = 40,33,44,31,30
lon2 = -113,-100,-102,-111,-112

x,y = m(lon,lat)
m.plot(x,y,'ro',markersize=2,alpha=.5)

x,y = m(lon2,lat2)
m.plot(x,y,'go',markersize=2,alpha=.5)

plt.title('Geo Plotting')
plt.show()
# plt.savefig("map1.png")
```