Brown Mathematical Contest for Modeling Fall 2021

Saving the Trees Through Visual Modeling

A Monte Carlo Approach

Hyoungjin Kim Benjamin Shih Alexander Zheng

Department of Applied Mathematics, Brown University

Contents

1 Problem Explanation

The advent of a technologically developed, modern society has brought many great advancements in our way of living, but along with such benefits, it has also brought along the now ever-pertinent issue of climate change. Although many may not be fully aware of the immediate impacts of climate change, Earth is already showing the symptoms of this illness, from dying reefs and coastal ecosystems to rises in sea level encroaching onto coastal lands to raging wildfires that have become more pervasive and devastating in recent years.

We want to explore this last issue, wildfires, in greater detail with regard to the growing crisis of bushfires in Australia. Although Australia has been a hotbed for wildfires for millions of year due to its geographical and climatic characteristics, the frequency of these fires has been on the rise alongside increases to global temperatures [1], and now poses a greater threat to wildlife as well as human population centers. Combined with other consequences of global warming that result in more frequent droughts, it has become more difficult to fight these bushfires after they develop into larger conflagrations. This issue is exemplified by the Australian bushfires in the summer of 2019-2020, which caused record levels of environmental, ecological, and financial damage and only slowed down due to increased rain fall in the winter months [2].

These issues are not unrealized, as the Australian government has been taking greater strides towards efforts aimed at ameliorating the wildfire problem through preventative methods, which work to reduce the ability of fire to spread without additional after-the-fact intervention methods. Namely, forest management techniques such as hazard reduction burning (HRB) and logging have been employed to mitigate the dangers of wildfires. These methods are based on the idea that removing combustible fuel sources in the form of foliage causes wildfires to spread more slowly and also burn out more quickly, allowing fires to be contained and responded with a reduced likelihood of them getting out of control in magnitude.

1.1 Existing Works and Research

Existing research into both HRB and logging as preventative measures working to lessen the severity of forest fires has led to contentious claims being made, especially more so regarding HRB.

The general agreement amongst researchers of the field is that logging does not help to prevent forest fires, but can actually lead to exacerbating the problem [3]. This is backed by evidence from sampling research done across the globe and the reason often attributed to this is that logging leaves behind a large amount of organic debris on the forest floors. Such debris is composed mostly of smaller, combustible materials such as branches, stripped barks, and thatch shavings. These materials may not contain as much biomass as the original forest, but they are liable to catching on fire more quickly as well as being good initializers for spotting events. Spotting is the process in which smaller plant materials are lit, then carried far away from the current region on fire due to updrafts that form in forest fires. These materials can then start their own fires kilometers away, creating new fronts and rapidly increasing the rate at which forest fires can expand to. Especially considering the Eucalyptus-heavy forests of Australia, spotting becomes a major factor to watch out for in forest management because Eucalyptus bark is known for being a major source of spotting spread to far away areas of the forest [4].

Despite this general agreement from the scientific community, the United States Department of Agriculture still undertakes logging operations as a way to combat forest fires, albeit with a greater focus on clearing underbrush and not leaving behind foliage detritus [5].

The practice of HRB however, is more hotly contested in both academia and amongst policymakers. Research conducted in 2015 on 34 years worth of fire data in Australia [6] used the idea of leverage to measure the effectiveness of HRB efforts conducted by the government. Essentially, the evaluation was based on whether the total forest area burned by HRB and forest fires in those treated regions was significantly less compared to similar regions that had not been treated with forest fires, given that conditions like topography and weather were similar between the regions. It found that based on the measure of leverage, HRB efforts did not produce a significant reduction to the total forest acreage burned and concluded that in the majority of cases, HRB, or prescribed burning as it is called in the paper, "is likely to have very little effect" on the severity of subsequent fires in the treated region.

On the other hand, research into the impact of fuel-reduction burning efforts in the states of New South Wales and Victoria found that satellite imaging showed wildfires in forest swaths pre-treated by HRB efforts to be less severe by statistically significant margins [7]. However, the research team did acknowledge that more recent treatments via HRB, such as that within the year of the fire, played a much more impactful role in reducing the severity of wildfires. Thus, they concluded that in general, under non-extreme weather conditions, recent HRB efforts can play a role in effectively reducing wildfire severity.

2 Outline

To address the question at hand, our research objective is to develop and implement a model for fire spread and forest recovery for South Eastern Australia. Given this model, we propose the most successful combinations of HRB and logging in order to reduce bushfire severity. To do so, we proceed as follows:

(1) Enumerate and discuss independent environmental factors that affect bushfire spread, as well as the assumptions and constraint that we impose on our model.

(2) Formulate a model based on parameterization of the aforementioned factors.

(3) Implement a simulation method that generates results for fire spread and allows for experimentation with the proposed model.

(4) Analyze and discuss the results of simulations and modeling.

We follow these steps to try to answer two key questions. First, we wish to confirm whether theoretical models of fire spread coincide with and affirm ecological theory and empirical evidence that logging causes more harm than does good with regards to mitigating the severity of forest fires. Secondly, we aim to explore the effectiveness of HRB as a forest management tool and to derive a rough model estimate of the magnitude of fire reduction as a result of such efforts in an attempt to help guide policy discourse with regards to the value of HRB efforts.

3 Model Breakdown

To achieve these goals, we ultimately created a model can be broken up into 3 integral parts: the landscape, local spreading, and spotting.

3.1 Landscape

We chose to use a discretized forest representation showing a forest divided into an n by n grid, with each cell in the grid representing a 1 km by 1 km region. Each cell within the grid has one of the following ecological terrain types: normal, HRB, logged.

We represent the entirety of the forest as a square rather than in a more natural shape because our model represents an arbitrary, small segment of the forest. Our model seeks to explore what happens in the continuous swaths of trees that makes up the vast majority of forests rather than look at the scenarios in which the fire is near the edges of the forests, given that such fires are much more common as well as posing a bigger threat of continuous spreading.

3.1.1 Description of Plants

1. **Tree-like Plants.** Tree-like plants describe the larger plants within the Australian Eucalyptus forest. These plants usually are taller, have crowns/canopies, and have more biomass than shrub-like plants.

2. **Shrub-like Plants.** Shrub-like plants describe the smaller plants within the forest. These plants are usually smaller, shorter, closer to the ground, and have a higher tendency to drift with the wind and cause spotting events.

3.1.2 Descriptions of Land Types in our Model

In the rest of this article, we will be using land and cells interchangeably. Land represents a plot of land in the real world, but in our model n by n representation, we refer to the same region as a cell.

1. **Normal**: Normal cells represent land that is not affected by HRB and Logging. Normal cells represent natural forest without human modifications. In this model we assume that the plant life here consists of tree like plants and shrub like plants. This land type has the greatest biomass since it contains trees and shrubs. Because it has the greatest biomass, it is the most flammable of the 3 types of land.

2. **HRB**: HRB is a cell type for a region characterized by a biomass composition of almost entirely trees. This is because hazard reduction burning removes most of the flammable shrub-like vegetation as well as the more flammable outer layers of tree bark. Of the three land types in this model, HRB regions have the second highest biomass because this region represents a normal forest with just the smaller trees and shrubbery removed. Because this land type has the second highest biomass, it is the second most flammable of the 3 types of land.

3. **Logged**: Logged is a cell type or region where the composition of biomass is almost entirely shrubs. This is because in the years following the logging of trees, the regrowth of plants consist of smaller more shrub-like plants. Out of the three land types in this model, logged regions have the lowest biomass because this region only has shrubs and shrubs tend to have lower biomass than trees.

   An important distinguishing feature about logged lands is that the shrub-like material which characterizes the vast majority of the region's biomass is light and therefore has a higher chance of causing spotting events compared to tree-dense regions. This is an crucial characteristic of this land type that will be the cornerstone of the reasoning behind mathematical methods used later in our paper.

3.1.3 Landscape Assumptions

To turn a real world representation of forests into a simplified model on which we can project and simulate, we made the following assumptions regarding the landscape of a forest:

1. We assume a relatively flat landscape for the forests, allowing us to ignore the effect terrain angle has on fire spread. We found this to be a reasonable assumption to make, given the non-mountainous characteristics of Australia's terrain.

2. We do not implement an explicit representation of other environmental factors that may affect the spread of wildfires such as wind speed and direction, temperature, and topographical features such as rivers, roads, and lakes.

3. We simplify the terrain into a three different types of land, when in reality, the biomass and burning characteristics of the forest section is more continuous and unpredictable. We do this because the policy we hope to be informing revolves centrally around logging and HRB operations. To distinguish the effects of these particular forest management methods, we treat each treatment as a distinct cell type and group the rest of the untreated forest into another singular cell type.

3.2 Fire Spread

In this section we wish to capture the following ideas using mathematical models.

1. Burn Rate
2. Local Spreading
3. Spotting

3.2.1 Burn Rate

Burn rate describes the rate at which biomass is consumed given that a cell is burning. We can model the burn rate as a differential equation:

$$\beta = \frac{db_t}{dt} = -nb_t - m, \text{ for } b > 0$$

where:

$$\begin{aligned} n &: \text{ scaling constant representing flammability of biomass in each land type} \\ b_t &: \text{ biomass in cell at time t} \\ m = .25 &: \text{ constant representing flat burn rate, which is independent of cell type} \end{aligned}$$

The constants n are defined in the following table:

	Normal	HRB	Logged
n	0.15	0.1	0.2

Reasoning. Logged regions have a higher constant N because logged regions are mostly consisting of shrubs and shrubs burn quicker. HRB regions are the opposite, with trees and no shrubs, meaning that burning is slower. Normal regions have both shrubs and trees, so it's constant N is in between that of logged and HRB. Additionally, our usage of 0.25 as our constant burn rate is derived from literature on how long a forest typically burns for. By integrating our differential equations with respect to time and solving for when biomass in the system reaches 0, we were able to arrive at a value of 0.25 for the constant spread.

3.2.2 Local Spread

Consider $L(b_t)$ to be the function of local spread from a cell to an adjacent cell.

$$L(b_t) = \frac{b_t}{B_{max}} \times j + k$$

where:

$$\begin{aligned}
t &: \quad \text{time step of model} \\
b_t &: \quad \text{current remaining biomass of the cell} \\
B_{max} = max(B_{normal}|B_{hrb}|B_{logged}) &: \quad \text{maximum initial biomass of each cell type} \\
j = 0.5 &: \quad \text{scaling factor for increased fire spread due to cell type} \\
k = 0.3 &: \quad \text{flat probability of fire spread} \\
\text{with condition} & \quad\;\; j + k < 1
\end{aligned}$$

Reasoning. Previous research on the correlation between the amount of biomass available to be consumed as fuel and rate of spread indicates that the two variables have a linear relationship [9]. Given this, we began our formulation of a function for the probability of local spread based on the proportion of biomass remaining relative to the maximum initialization of biomass for a given cell type. From there, we scaled it with factor j and added a flat probability k with the condition that the sum $j + k$ is less than 1 to ensure that the probability of spread does not exceed 1. We settled on values of 0.5 and 0.3 for j an k, respectively, giving us a probability distribution of local spread ranging from 0.3 to 0.8, with the exact value within the range being determined by how much burning has already occurred. We found these settings to be reasonable, because it gave a greater weighting to cell type in the determination for local spread probability, while the constant k ensured that our model fire would mimic that of a real world fire in continuing to spread to nearby regions of the forest even when low on available fuel.

3.2.3 Spotting Spread

Spotting is the cause of new wildfire due to ignited materials being carried downwind. For the rest of the paper, we will define a 'spotting event' as a single piece of ignited material from an existing fire, drifting downwind to a destination causing a new potential fire. In Figure 1, you can see examples of spotting. The warm colors in the model represent fire. The warm colors that are disconnected from the central fire are caused by spotting events. By inspection, you can see that there are more spot fires closer to the center and less spot fires farther away. The methods to create this model are detailed in the remainder of this section.

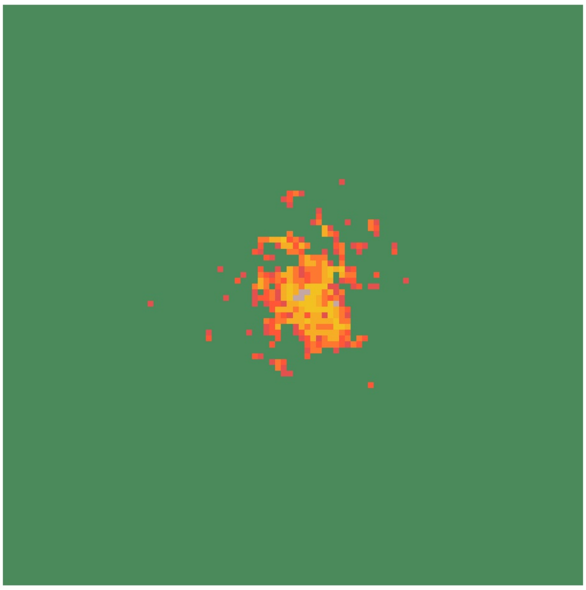


Figure 1: Spotting Example

Fire Intensity

- High
- Medium
- Low
- No Fire

Given that a cell or piece of land is on fire, we want to answer the following questions about spotting:

1. How many cells within spotting radius will be catch on fire due to spotting?
2. What (is/are) the index of the cell(s) within spotting radius that will catch on fire?
3. Once the index of destination cells are decided, what is the probability they catch on fire from ignited material from a spotting event?

To answer Question 1, consider the function $\mathbf{N}(\gamma)$:

$$\Gamma = \{\text{'normal'}, \text{'HRB'}, \text{'logged'}\}.$$
$$\mathbf{N}(\gamma) \to s \in S, \gamma \in \Gamma \quad : \text{number of spotting events.}$$
$$\text{The output space for } \mathbf{N}(\gamma) \text{ is } \{0,1,2\}$$

Let $N(\gamma)$ be a function that maps the cell type to the number of spotting causing artifacts (ignited material) that will arise from that cell. This function maps the cell type t to an integer representing the number of other cells that will catch fire due to spotting. Given that the cell is burning, the cell can produce between $0-2$ spotting events. The cell produce 0,1, or 2 spotting events with probabilities listed in the chart below.

N(γ)	Normal	HRB	Logged
0	0.85	0.9	0.5
1	0.13	0.09	0.35
2	0.02	0.01	0.15

To answer Question 2, consider a probability distribution P:

The sample space for $\mathbf{P}$ is the set of integers $\{1, 2, ..., 30, ..., 45\}$, such that $p \in \mathbf{P}$ where p is the Manhattan distance between the spotting event and spotting destination. This distribution is represented in Figure 2. In summary, the distribution resembles a normal distribution with a strong rightward skew. The reason for this is because the a report from 1983 for an Australian wildfire reported the average spotting radius for Australian Eucalyptus forests is 5-12 kilometers with extreme distance up to 25km. In order to capture this, we wanted a probability model that regularly outputs spotting distances between 5-12 kilometers and on rare occasions produce a spotting distance that is around beyond 25 kilometers.

The distribution P captures these intentions because the bulk of the probability density lies above $\pm$(5 to 12) kilometers and there is a long tail for longer distances such as 25-30 kilometers. The exact probabilities are less relevant here since what we care about is the relative frequencies. This distribution can be substituted in with other probability distributions that represent different spotting regions. For example, if the fire starts on top of a tall mountain, the spotting region would be farther and we would want to replace the distribution with one that has less of a peak and fatter tails.

From this distribution $\mathbf{P}$ we can now select a cell/land that is p Manhattan distance away from the source of the spotting event. The details of this algorithm is detailed in the section on Algorithms.

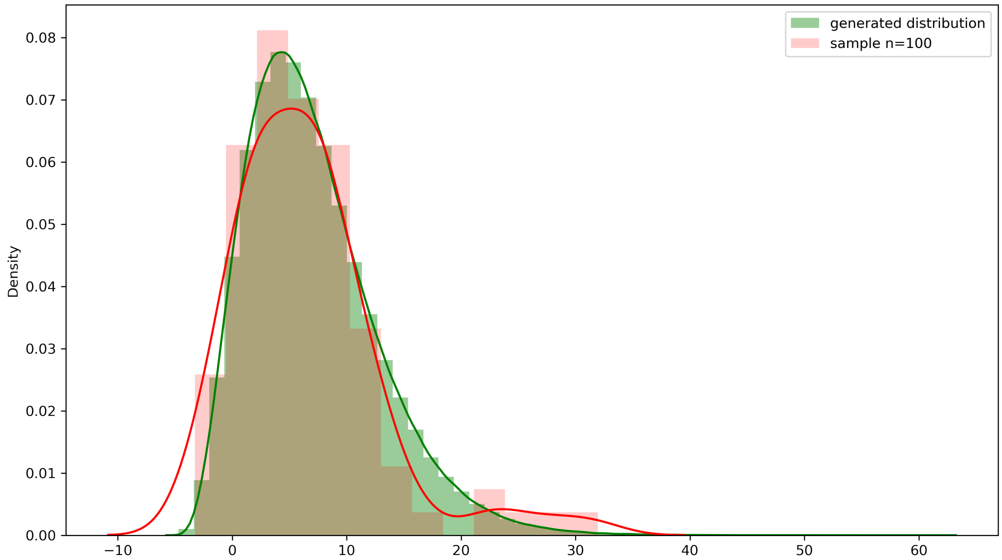


Figure 2: **P** Distribution

To answer Question 3, consider a function $\mathbf{C}(\gamma)$:

$\mathbf{C}(\gamma)$: the probability that the destination cell catches on fire with ignited material.

The function $\mathbf{C}(\gamma)$ maps the cell type to a combustion probability, which is the probability that the destination square will catch on fire because of the ignited material being carried by wind. The combustion probabilities per cell type are listed below.

	Normal	HRB	Logged
$\mathbf{C}(\gamma)$	0.9	0.7	0.5

The reasoning behind these probabilities is based on the relative biomass of each cell type. The normal cell type has highest chance of catching on fire because there exists shrubs and trees. The HRB regions have a slightly lower probability compared to the normal region because HRB regions have less shrubs. The logged regions have the smallest probability because there there is less biomass since only shrubs exist and no trees.

Conclusion:

1. With the answer to question 1, we can look at all the existing cells in the model that are on fire and determine how many spotting events will occur from those cells.

2. With the answer to question 2, we can decide from the burning cells, how far away should we send the spotting event.

3. With the answer to question 3, we can model the probability that the spotting event creates a fire at the destination cell.

3.3 Spreading(Local/Spotting) Assumptions

1) **Frequency of Spotting Events**

   In our model, we assume that a burning tile will produce 0 to 2 spotting events that can cause fire elsewhere within a reasonable range. In real life, the number of spotting events is likely more variable than the assumed range. Because the frequency of spotting depends on so many factors such as wind speed, wind angle, temperature, and aerodynamics, calculating the true number of spotting events is near impossible. This assumption is still reasonable because it still captures the relative frequencies at which the 3 different land types will produce spotting events.

2) **Spotting Radius**
In this model, we represent the spotting radius as a probability distribution based on Manhattan distances. This is not true in reality since forests are not discretized and spotting occurs in a continuous distribution, which cannot be captured by Manhattan distances. Additionally, the distance over which real-world spotting events can travel depends on other factors such as wind, temperature, and characteristics of ignited material(species, mass, dryness, etc), etc...

For our model we used previous scientific literature to determine that the most likely distance for spotting is 5-12 kilometers with low probabilities of spotting occurring up to 30 kilometers away [8]. We used this as the basis for our distance probability distribution, so that despite the imperfections, the model approximated real-world spotting events with reasonable accuracy.

3) **Burn rate is only dependent on biomass** In this model, the biomass of each cell type is recorded and updated at each time step. We make the assumption that variations in burn rate is only dependent on the biomass of the cell type. This is not true entirely true in the real world because burn rate is also dependent on factors such as the dryness and other characteristics of the biomass present at the time of burn. This is a reasonable assumption because we believe overall biomass plays a greater role in fire intensity than smaller factors such as plant dryness, plant type, and plant density. Because fire intensity is directly related to burn rate, we conclude that biomass is likely a primary contribute to burn rate. Therefore, in a setting barring extreme weather conditions, we conclude that that this approximation allows for a simplified yet accurate representation of biomass burn rate.

4) **Direction of burn is probabilistic uniform** The direction of burn does not depend on factors such as wind or land elevation. We make this assumption because the simulation of environmental conditions such as wind and terrain elevation is highly unpredictable and difficult within the scope of this paper.

5) **Discretized forest in which fires only spread in x and y direction** Because of the n by n grid simplification of the forest, fires can only spread in the horizontal and vertical directions. This was a reasonable simplification and assumption to make because the macroscopic effects of fire through the forest can still be observed when the n by n representation has a significantly large n such as $n = 1000$, where the grid discretization of the forest becomes less noticeable, and more closely approximates a continuous forest terrain.

6) **Combustion probability only depends on biomass** We assume that the landing of an ignited material on unburnt land has a combustion probability that is only dependent on biomass. This is reasonable because the biomass within a cell is correlated to the cell type, and thus indirectly maintains the association to the relative flammabilities of the 3 cell types.

3.4 Time Model

The model uses a time step variable t which is initialized to 0, when the fire begins to spread. The model iterates to a time step of $t = 1000$ and terminates. At every time step, a series of events happen to simulate the spread of a wildfire, described below:

1. Check every cell in the model that is burning.
2. For every cell in the model that is burning, calculate probability of local spread using $L(b_t)$ and set fire to the adjacent square.
3. For every cell in the model that is burning, calculate how many spotting events each burning cell produces, using function $\mathbf{N}(\gamma)$.
4. For every spotting event that needs to happen, calculate the destination square based on a probability distribution $\mathbf{P}$.
5. For every destination that the spotting event lands, calculate the probability of combustion using $\mathbf{C}(\gamma)$
6. For every cell in the model that is burning, use the burn rate function β to change the biomass of burning cells.

4 Results and Analysis

4.1 Wildfire Analysis

In this section we describe how to use our model to initialize the simulation by setting terrain and initial fire location.

4.1.1 Case: Uniform Normal Forest

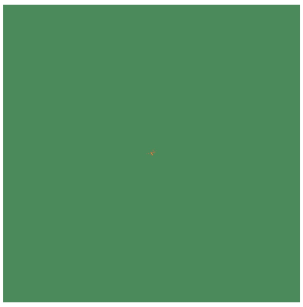
We first examine the forest model where every cell in the model forest is of type normal. This means the entire forest is untouched and has not been modified by HRB or logging. This will serve as a control for the analysis of other scenarios within this model. With the terrain generated, we set fire to the middle of the forest and let the model run for 100 time steps. Throughout the simulation, we see evidence of spotting and local spread. The burn of this model forest is shown in Figure 3.

4.1.2 Case: Uniform HRB Forest

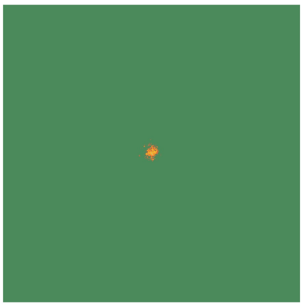
Here we examine the forest model where every cell has been been modified by HRB, meaning that the entire forest biomass consist mostly of trees with very few shrub like vegetation that cause spotting events. With the terrain generated, we set fire to the middle of the forest and let the model run for 100 time steps. The burn of this model forest is shown in Figure 4. Based on inspection, we see that the HRB burn region is similar in size to the normal model, but HRB has a slightly smaller burn area.

4.1.3 Case: Uniform Logged Forest

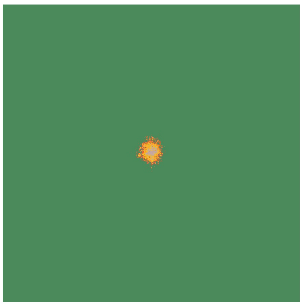
Here, we examine the forest model where every cell has been modified by logging. This means that this model forest has little to no trees and the majority of the biomass in this scenario is shrub-like plants that have a higher tendency to cause spotting events. With the terrain generated, we set fire to the middle of the forest and let the model run for 100 time steps. The burn of this model forest is shown in Figure 5. We observe that the logged burn region is considerably larger than the burn region of the normal forest.



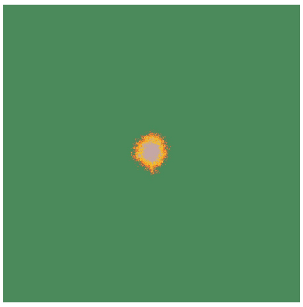
(a) t = 10



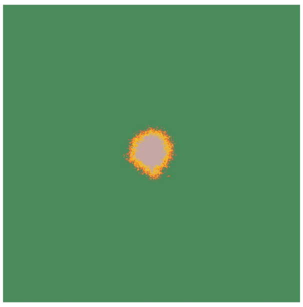
(b) t = 20



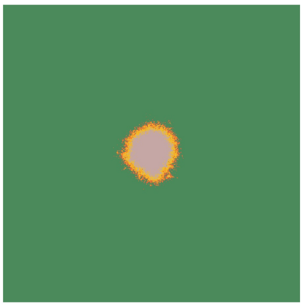
(c) t = 30



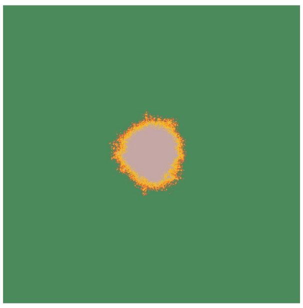
(d) t = 40



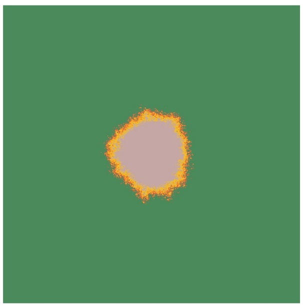
(e) t = 50



(f) t = 60

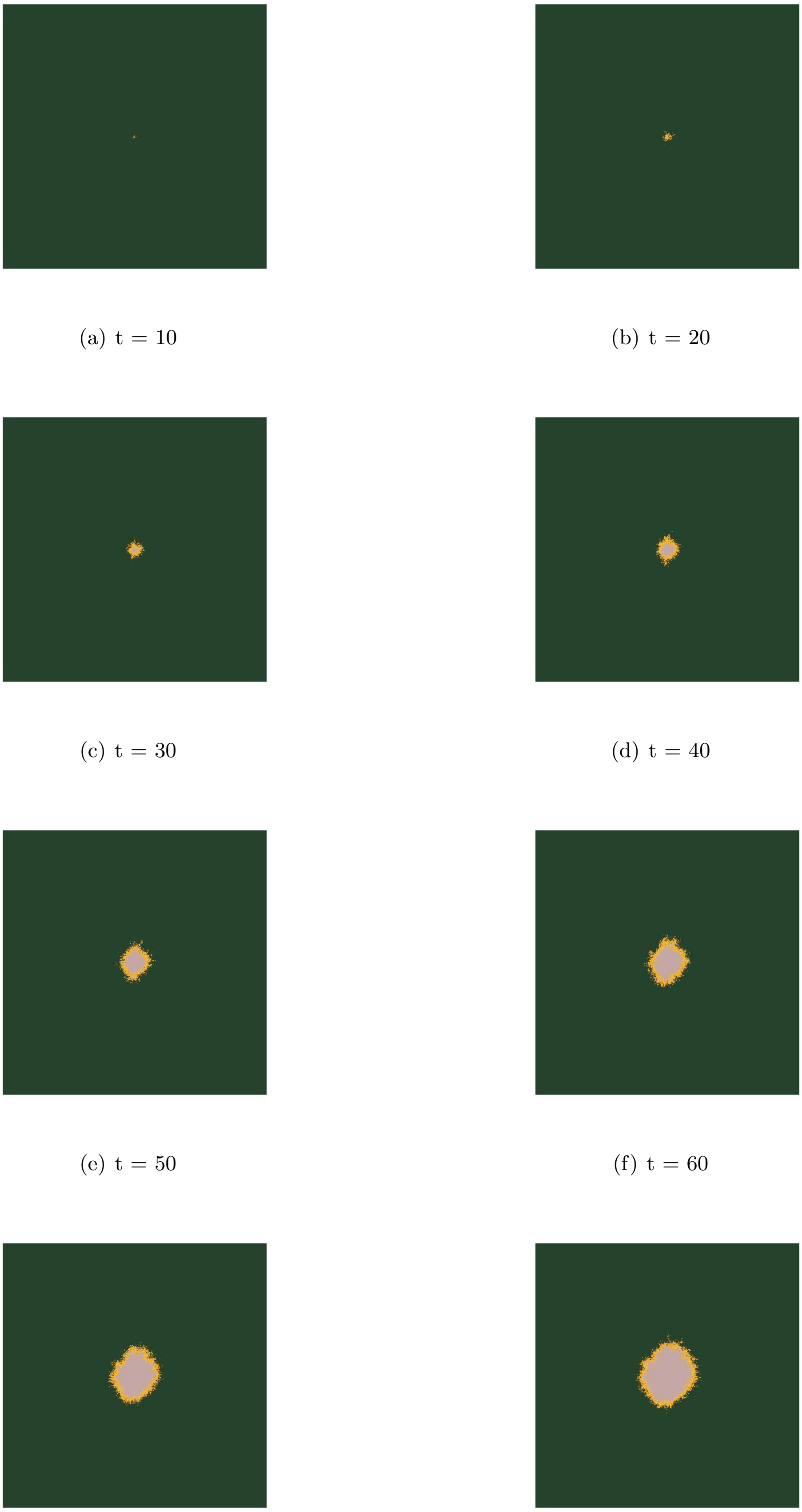


(g) t = 70



(h) t = 80

Figure 3: Simulation on normal-only terrain



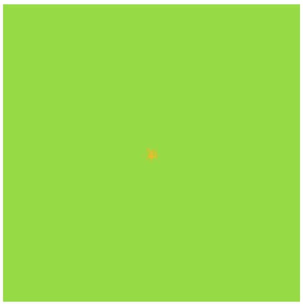
(a) t = 10 (b) t = 20

(c) t = 30 (d) t = 40

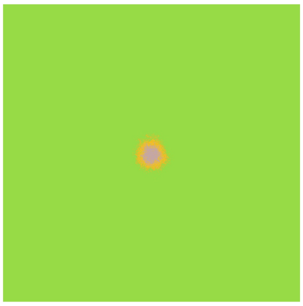
(e) t = 50 (f) t = 60

(g) t = 70 (h) t = 80

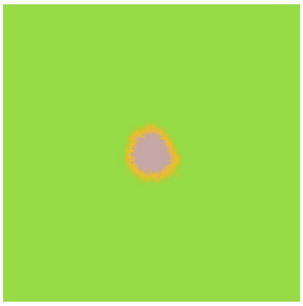
Figure 4: Simulation on HRB-only terrain



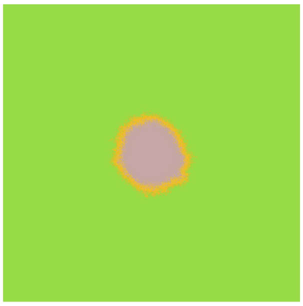
(a) t = 10



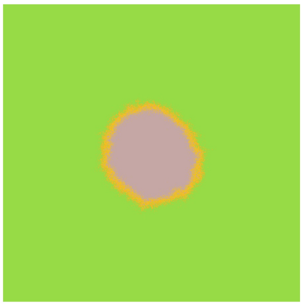
(b) t = 20



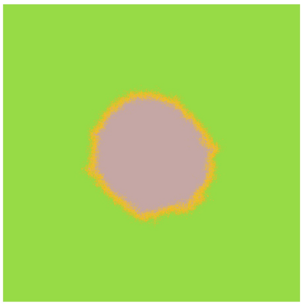
(c) t = 30



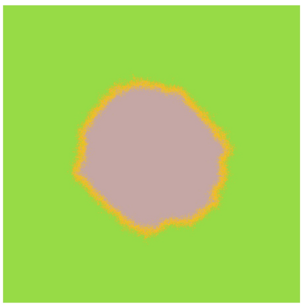
(d) t = 40



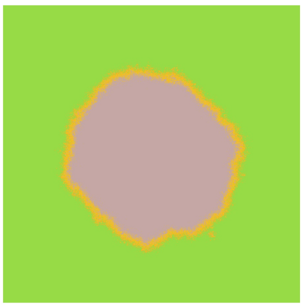
(e) t = 50



(f) t = 60

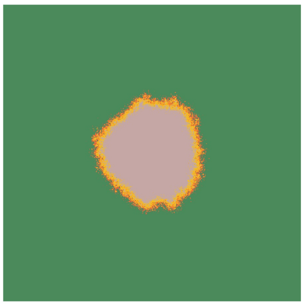


(g) t = 70

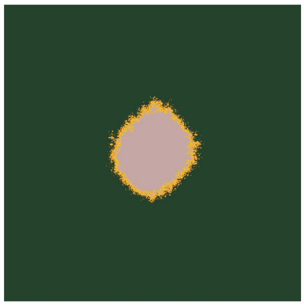


(h) t = 80

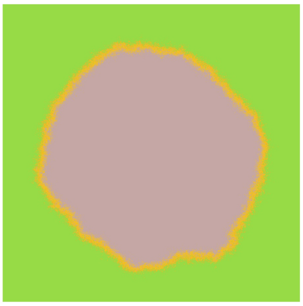
Figure 5: Simulation on logging-only terrain



(a) Normal Forest at t=100

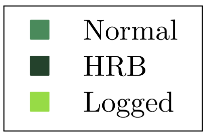


(b) HRB Forest at t=100



(c) Logged Forest at t=100

Figure 6: Normal, HRB, and Logged Forest Fires at t=100



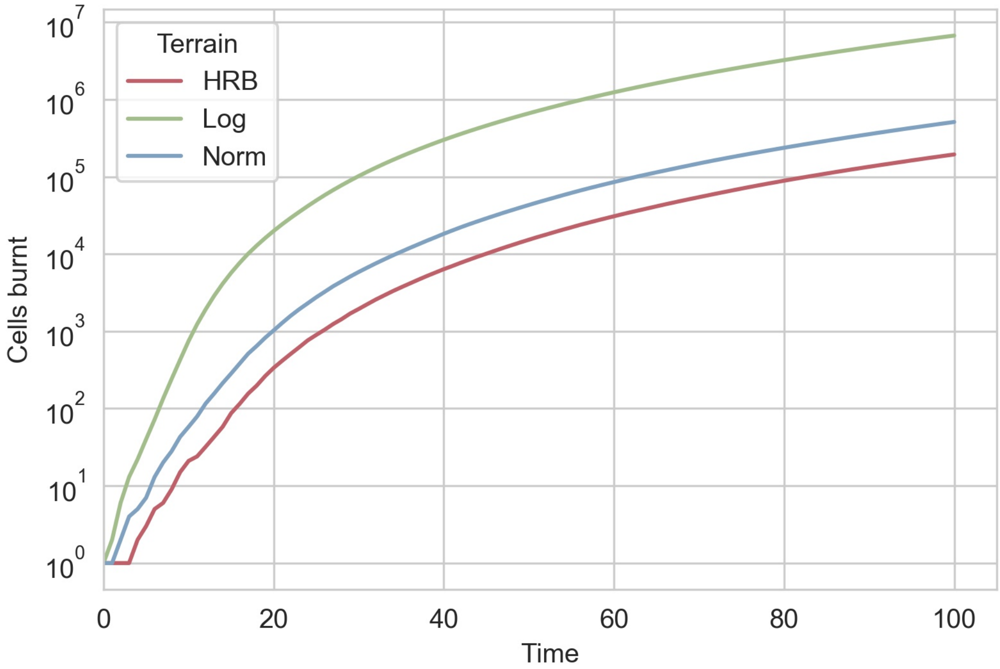


Figure 7: Average Cells Burned as a Function of t

4.1.4 Insights From Uniform Forest Models

Figure 7 is a graph of average cells burnt as a function of time t, where t ranges from 0 to 100. We obtained the averages for the different terrains at each time step through Monte Carlo simulation, in which we conducted a 100 iterations for each cell type. Due to the stochastic nature of our model, despite maintaining the same initial environment, each run of the simulation resulted in varying patterns of wave fronts, spotting, and directional spread, allowing us to capture a variety of scenarios that could occur in a given setting.

In the graph above, we measure the number of cells burnt across the timesteps, which we can use as a proxy for the severity of the fire at a given time point. We can easily see that the logged regions had the worst burnout, followed by normal, then by the HRB forest models. From our model, Figure 7 suggests that logging a forest and leaving behind a higher concentration of shrub-like plants results in more intense and faster-burning wildfires.

4.1.5 Normal Forest vs Fully HRB Forest

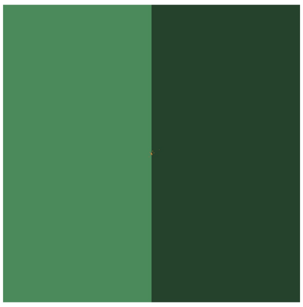
In this simulation, the forest is divided into two halves - the left half as uniform normal forest and the right side as uniform HRB forest. We initialize a fire at the middle of the forest where the two types of terrain meet and observe the fire spread across throughout both forests.

Looking at Figure 8 we note that fire spread is fairly similar on both sides of the forest; however, the burn reaches slightly further on the normal forest side. Given the assumptions made in our model, the simulation suggests that HRB management of the forest reduces burn intensity and rate of fire spread. Considering the nature of HRB regions, there should be less spotting since there is less shrub material, and the model visually captures that theoretical distinction by way of our model weights.

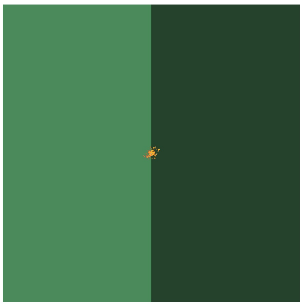
4.1.6 Normal Forest vs Fully Logged Forest

In this simulation, the forest is divided into two halves - the left half as uniform normal forest and the right side as uniform logged forest. We initialize fire at the middle of the forest where the two types of terrain meet and see the fire spread across throughout both forests.

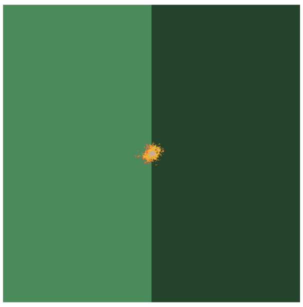
Looking at 9, we observe that fire spread for the logged regions is much more pervasive than that in the normal forest. Given the assumptions made in our model, the simulation suggests that logging forests does increase the intensity of wildfires. Considering the nature logged lands which have more shrubs to cause spotting events, our model captures the impact of these spotting events in pushing along the front of the fire at a much more rapid pace when compared to fire spread in an untreated forest.



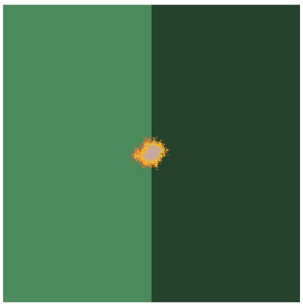
(a) t = 10



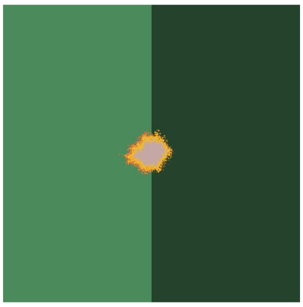
(b) t = 20



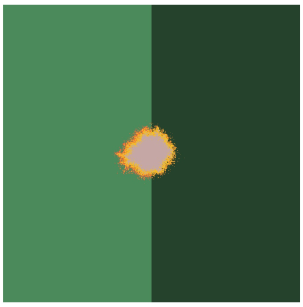
(c) t = 30



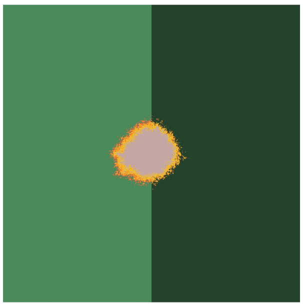
(d) t = 40



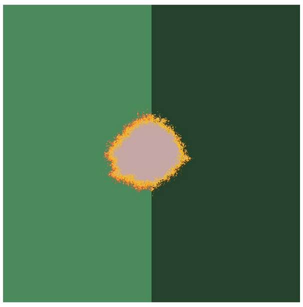
(e) t = 50



(f) t = 60

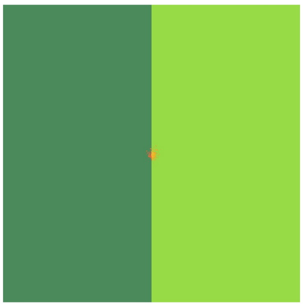


(g) t = 70

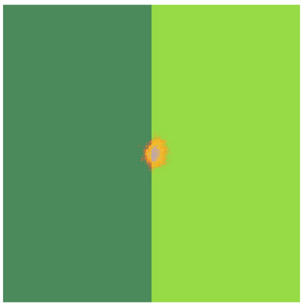


(h) t = 80

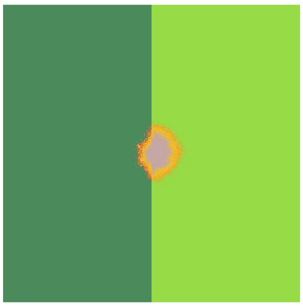
Figure 8: Simulation with normal forest on the left and HRB on the right



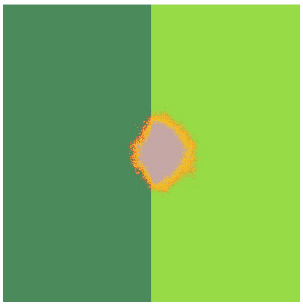
(a) t = 10



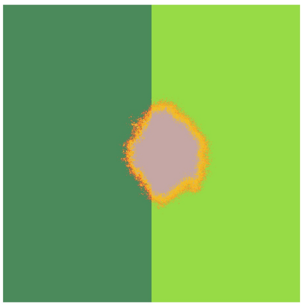
(b) t = 20



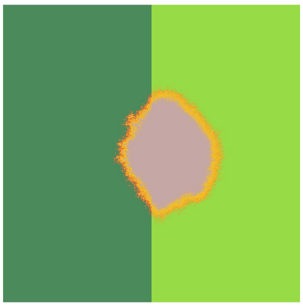
(c) t = 30



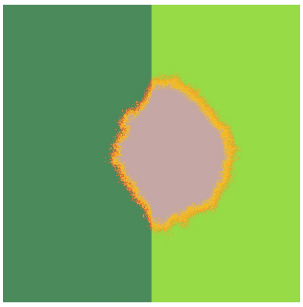
(d) t = 40



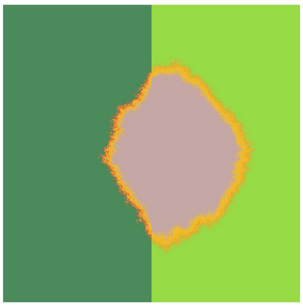
(e) t = 50



(f) t = 60



(g) t = 70



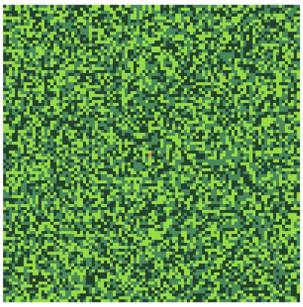
(h) t = 80

Figure 9: Simulation with normal forest on the left and logged on the right

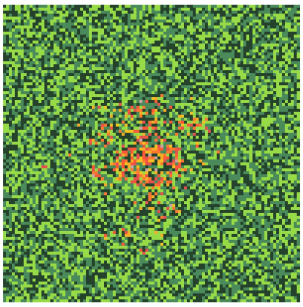
4.1.7 Additional Simulations

We provide a more granular view of our model simulations in the following figures; due to the smaller dimension of the environments, spotting and its effect on state is more apparent compared to environments with larger dimensions. We demonstrate our simulator and model's encapsulation of multiple terrain types and modularity by providing figures depicting fire spread in a "four corners" configuration, where each quadrant of the grid is of a different land type. With the versatility of our model to essentially represent any type of terrain and modification given the necessary map and input parameterizations, we hope to further explore different aspects of wildfire containment with additional environmental variables, such as climate, altitude, etc.

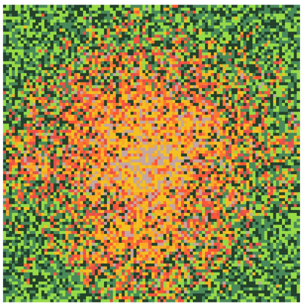
100x100 Randomly Generated Terrain



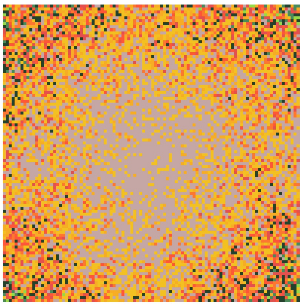
(a) t=1



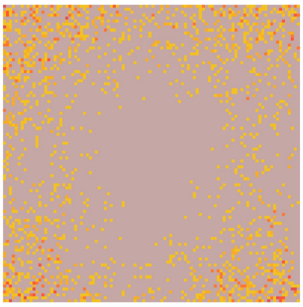
(b) t=10



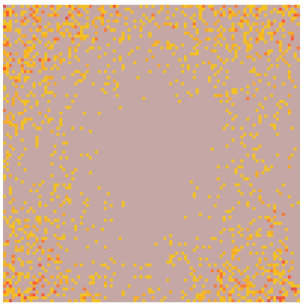
(c) t=20



(d) t=30



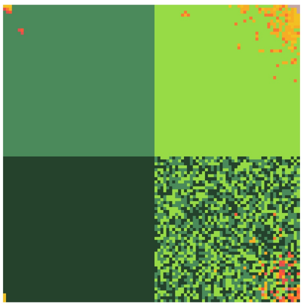
(e) t=40



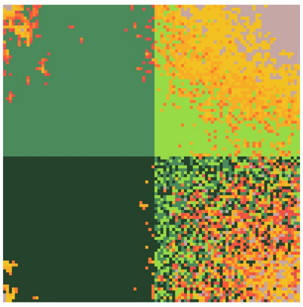
(f) t=50

Figure 10: 100x100 Randomly Generated Terrain

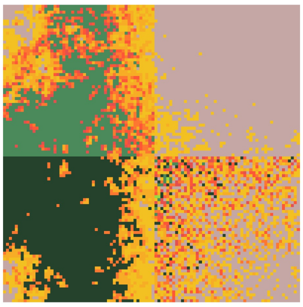
TopL,BottomL,TopR,BottomR is normal,HRB,logged,mixture respectively.



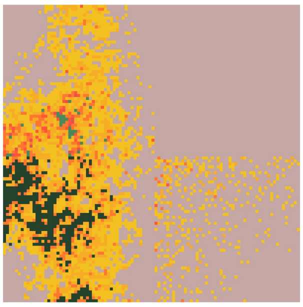
(a) t=10



(b) t=20



(c) t=30

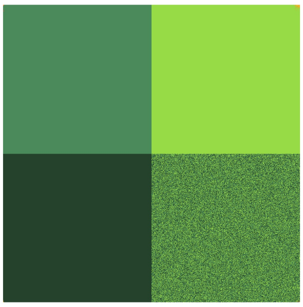


(d) t=40

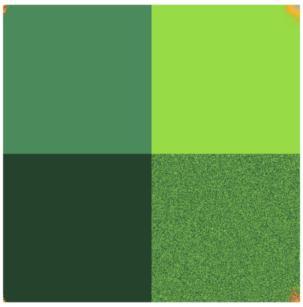
(e) t=50

Figure 11: 4 quadrants of different terrain types with fires starting in each of the 4 corners

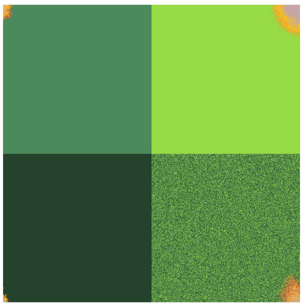
TopL,BottomL,TopR,BottomR is normal,HRB,logged,mixture respectively.



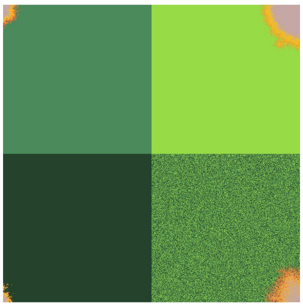
(a) t=10



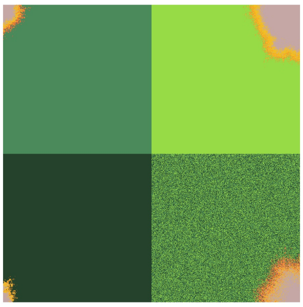
(b) t=20



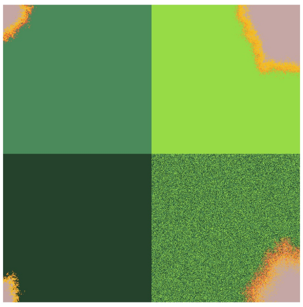
(c) t=30



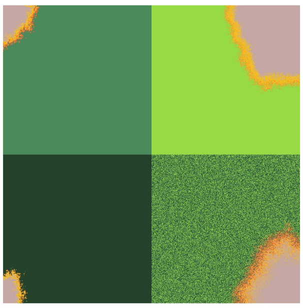
(d) t=40



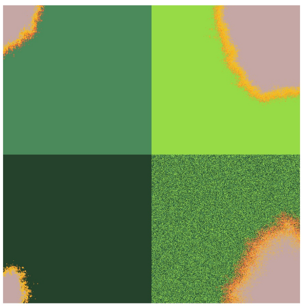
(e) t=50



(f) t=60



(g) t=70



(h) t=80

Figure 12: 1000x1000 4 Different Corners

Figure 11 shows a race between the wildfire across different terrains. Logged regions show the highest spread, and HRB regions show the lowest spread. Figure 12(d) shows an excellent example of spotting.

5 Strengths and Weaknesses

5.1 Strengths

1) Terrain Generation

The benefit of our model is that it is able to model fire spread across different terrains. In our model, we simulated local and spotting spread across 3 different land types(normal, HRB, and logged). Taking the current model and code, we are easily able to generate new cell types based on different terrain conditions or forest management techniques with different initial conditions. Therefore it is possible to generate more complex terrains with patterns such as logging in strips, logging in clusters, and other patterns of forest management.

Additionally, the model can even incorporate terrain such as mountain ranges, rivers, and other common environment features, which each have its own effect on fire spread. We can easily adapt our terrain generation to model specific instances of real-world forests, and include terrain that disrupts the spread of fire.

Overall, the ability to customize the terrain makes this model not limited to the Australia's bushfires. The model is able to adapt, recreate, and simulate forests of different conditions throughout the world.

2) Visual Simulation of Wildfire

Because our model is a simplification of a forest with careful considerations to real world dimensions, our model simulation is able to visually represent the state of the forest at discrete time steps throughout the wildfire. From our n by n grid model, we are able to see a top-down view of which sections of the forest are on fire and which sections have been burned, in addition to seeing the step-by-step progression of the fire advancing out from the origin. This creates an intuitive representation of the wildfire, instead of graphs that represent raw statistics regarding the spread of fire. Additionally, our model represents the positional features of the fire front and spotting in a visual manner that allows us to now apply our model to analyze new terrain types without prior knowledge of the terrain's fire spreading characteristics.

5.2 Weakness

1) Unnatural Generation of Terrain

Although we have great control over how the terrain is laid out at the beginning of the models simulation, the terrain is over-simplified since there are some underlying patterns in the forest growth. Similar trees and similar plants tend to grow in localized patches. Forest management techniques are not applied completely across the entire forest or in a random fashion. In our terrains, which are uniformly one land type, we are ignoring some of the natural patterns of forest growth. With more time, we would have continued exploring different terrain patterns to see the effects of HRB and logging compared to normal forest land.

2) The Fire Spread Dependency

Our fire spread model looks at the squares that are on fire and through probabilistic models, create more nearby fires that align with the terrain's conditions. This is a weakness of the model since the fire-front's cohesive properties are not expressed. Since every cell that is burning is considered by itself, the model does not capture the relationship between different cells which are on fire, and perhaps the size and shape of the entire fire-front.

6 Potential Improvements and Future Models

1) **Wind**

One major factor that contributes to the spread of fire is the characteristics of the wind during the fire. Fire has tendency to spread along the direction of the wind, and the rate of the spread is also influenced by the magnitude of wind speed.

In a improved model, we can keep track of a wind vector $v = (\theta, V)$ which has a direction and magnitude. Its direction θ will be incorporated into the $L(b_t)$, the probabilistic determination of which adjacent cell local spreading will set fire to. The velocity V can be used to model the extent the wind affects the probabilistic spread in direction θ.

By tracking wind, we can also use that to improve the **spotting model**. The main reason for spotting is wind, and the region of forest susceptible to spotting fire is dependent on wind direction and strength. Again with a vector $v = (\theta, V)$, we can apply the mentioned reasoning to spotting spread. Wind will directly influence the distribution $\mathbf{P}$ in our spotting mode. With greater wind speeds, $\mathbf{P}$ will have longer tails meaning spotting events will have higher probability of travelling greater distances.

2) **Using Machine Learning to Find Optimal Terrain Layout**

An interesting future analysis using our model would be to find optimal terrain layouts using feedback systems, i.e. search for the best possible placements of HRB, Logged, and unaltered terrain cells. Roughly speaking, we could take a computer vision or reinforcement learning approach to the problem and have an agent generate map settings and provide negative reward to the agent for each burnt area accrued while simulating. The idea is that over time, the agent will converge to some optimum solution for a particular placement of each cell type in each cell on a map to minimize wildfire spread.

This has additional applications in aiding the development of HRB plans, as the goal of HRB is to maximize the reduction in the severity of future forest fires while minimizing the area burned in a controlled setting. By using our model and creating an evaluator function based on the HRB region and wildfire burn regions, we could use Machine Learning algorithms to find optimal burn sizes and formations, which could then be used to guide actual forest management. Additionally, due to the flexibility of our terrain design, we could combine satellite imaging data of forests we wish to manage and easily create custom models for a specific forest on which we could perform HRB optimization.

7 Conclusion

7.1 Final Thoughts

Wildfires are an incredibly unpredictable phenomenon that depend on many factors that are hard to measure. Despite the large number of generalizations and assumptions that were made, we believe that the visualizations our model produced provided reasonable and accurate representations of fire spread, in line with past scientific literature from the field. The model's visualizations conveyed the different rates at which fires spread across the different terrains, and the probabilistic nature of spotting. Referring back to Figure 1, we see that there are more local clusters of fires around the center and fewer spotting fires farther away. This natural representation of spotting gives us confidence in the probability model we used to represent the process of spotting. Thus we conclude that spotting is a significant factor in the intensity of Australian wildfires, and forest management techniques that increase the potential for spotting should be reconsidered in the overall forest fire management plan.

7.2 Proposal to NSW Parliamentary Research Service

Considering the results of our simulations, we return to address the question of what forest management technique should be used to optimally reduce the impact of wildfires. Our model decisively agrees with past research in the field in concluding that logging not only does not help in reducing the spread of fires, but further exacerbates the problem. Our model visualization makes the reason for this particularly clear, with the frequency of the spotting events allowing for the generation of new nucleation sites off of which new fire fronts can spread from. This point is particularly important to the forests in New South Wales, with Australia's forests consisting heavily of Eucalyptus trees which are notorious for having bark that easily spots to new locations.

With regards to HRB forest management however, our simulation results show that there is a distinct improvement over non-treated forests in HRB's ability to mitigate the impacts of forest fires. This comes with the caveat of needing a rather frequent process of controlled burning, ranging between 3 to 5 years. However, if conducted properly, HRB has the ability to burn off a large portion of excess biomass in forests that play a major factor in sustaining forest fires. Additionally, HRB forest management causes less pervasive and more temporary ecological damage, allowing for the preservation of forest ecosystems for most years while still working to effectively reduce the dangers of a wildfire spreading out of control.

In summary, HRB forest management is certainly a tool to be considered as part of forest fire prevention policy, but must be conducted following rigorous analysis of the local landscape and weather conditions to prevent losing control of the intentional fire. In contrast, although logging may have a niche role in forest management, we do not recommend its use as a reliable way to reduce the impact of forest fires, or as a way to slow down the spread of future forest fires.

8 Algorithms

8.1 P Distribution and Distance Algorithm Methodology

One of the more intricate algorithms and likely the most interesting in our implementation is the one that we use to simulate non local wildfire spread, i.e. spotting. We briefly discussed this procedure beforehand in Section 4.2.3 Spotting Spread; however, we further motivate the formulation we implemented here.

For any cell that is on fire, we first decide whether that cell will spot to 0, 1, or 2 other cells. In order to determine this, we define an explicit discrete probability distribution for each cell type with probabilities of spreading to 0, 1, or 2 other cells. This probability distribution can be found in the Section 4.2.3 in the $\mathbf{N(\gamma)}$ table.

Now that the number of cells to be spotted is known, it is then necessary to determine which particular cells the fire spots to. In order to determine this, we first determine an effective radius based on research. We found that spotting occurs most likely around distance of $7km$; however, particles have been observed to fly up to ± 5 to $12km$. Since our model is made of discretized cells, we imposed a distance metric using Manhattan distance and first generate the distance spotting should occur at.

As mentioned before, it is most likely that spotting occurs at distances of 7 kilometers but extreme distances are nonetheless observed. As such, we decided to use a probability distribution with a peak of 7 distance from the source cell and with a long right skew such that far spotting distances are also taken into account in our model. Forum board user B. Poe's implementation of an F distribution-based probability density curve [10] perfectly served our modelling needs and allowed for flexibility as a bonus (different spotting distance probabilities are easily modelled).

We first determine what Manhattan distance the spotting should occur at relative to any cell that is spreading fire non-locally. To do so, we generate a number with the distribution discussed above, then perform necessary algebraic manipulation in order to use it with our model, i.e. flooring and taking absolute value. Since there are multiple such cells that lie at the generated distance d away from the source cell, we then determine which one of these cells the fire spreads to.

In order to determine the cells, we simply generate a random integer from a uniform distribution, i.e. $r \in [0, d]$ and take its complement relative to d to come out with a coordinate difference (x, y). For example, for a generated $d = 17$, a resultant (x, y) pair could be any of $\{(0, 17), (1, 16), ...(16, 1), (17, 0)\}$. To conclude this process, we randomly assign the signs of these coordinates and add them to the coordinates of the spotting event to find the spotting destination.

9 Appendix

9.1 Code

Attached below is the code that was used to generate our model environment, establish our stochastic methods, and run Monte Carlo simulations across the varying terrains.

In [ ]:
```
import numpy as np
import random as rand
import random
import math
import seaborn as sns
import pandas as pd
import matplotlib.pyplot as plt
from matplotlib import rc
from matplotlib import colors
from scipy import stats
from dataclasses import dataclass
plt.rcParams['figure.dpi'] = 300
plt.rcParams['savefig.dpi'] = 300
```

Constants

$\texttt{N}$ $\rightarrow$ constant for biomass decay for different terrain types \ $\texttt{B}$ $\rightarrow$ constant for lower and upper bounds of initial biomass \ $\texttt{M_CONST}$ $\rightarrow$ biomass decay constant \ $\texttt{ENV_SIZE}$ $\rightarrow$ dimension of environment \ $\texttt{TIME}$ $\rightarrow$ Number of time steps to simulate \ $\texttt{J, K}$ $\rightarrow$ $L(B)$ constants

In [ ]:
```
N_n = 0.15
N_h = 0.1
N_l = 0.2
B_n_lower, B_n_upper = 10., 11.
B_h_lower, B_h_upper = 5., 6.
B_l_lower, B_l_upper = 3., 4.
M_CONST = 0.25
ENV_SIZE = 1000
TIME = 100
J = 0.5
K = 0.3
```

Cell Information

Biomass $\textit{biomass: float} \rightarrow$ The amount of biomass at a particular cell \ Burn Status $\textit{on_fire: bool} \rightarrow$ Whether or not a particular cell is burning

In [ ]:
```
@dataclass
class Cell:
    """Class for keeping track of an information in a cell."""
    curr_biomass: float
    max_biomass: float
    on_fire: bool
    is_burnt: bool
    N: float
    B: float

    def __init__(self, curr_biomass=0.0, on_fire=False, is_burnt=False):
        self.curr_biomass = curr_biomass
        self.max_biomass = 11.0 if isinstance(self, Normal) else 6.0 if isinstance(self, HRB) else 4.(
        self.N = N_n if isinstance(self, Normal) else N_h if isinstance(self, HRB) else N_l
        self.B = random.uniform(B_n_lower, B_n_upper) if isinstance(self, Normal)  \
                    else random.uniform(B_h_lower, B_h_upper) if isinstance(self, HRB) \
                    else random.uniform(B_l_lower, B_l_upper)
        self.on_fire = on_fire
        self.is_burnt = is_burnt

    def can_burn(self):
        return self.curr_biomass == 0

    def decay_biomass(self):
        nxt = self.curr_biomass - (self.N * self.curr_biomass + M_CONST)
        self.curr_biomass = nxt if nxt > 0 else 0.0
        if self.curr_biomass == 0:
            self.on_fire = False
            self.is_burnt = True

    def set_fire(self):
        self.on_fire = True

    def n_t(self):
        pass
```

```
    def l_b(self):
        return (self.curr_biomass / self.max_biomass) * J + K

class Normal(Cell):
    """Normal cell class"""
    def n_t(self):
        return random.choices(population=[0, 1, 2], weights=[0.85, 0.13, 0.02])[0]

class HRB(Cell):
    """HRB cell class"""
    def n_t(self):
        return random.choices(population=[0, 1, 2], weights=[0.9, 0.09, 0.01])[0]

class Log(Cell):
    """Logged cell class"""
    def n_t(self):
        return random.choices(population=[0, 1, 2], weights=[0.5, 0.35, 0.15])[0]
```

Test Cell Types

In [ ]:

```
a = Log(curr_biomass = 5, on_fire=False)
a.n_t()
```

Skewed F Distribution

Taken from user B. Poe, generates a skewed distribution with parameterized mean, standard deviation, skew

In [ ]:

```
def createSkewDist(mean, sd, skew, size):

    # calculate the degrees of freedom 1 required to obtain the specific skewness statistic, derived
    loglog_slope=-2.211897875506251
    loglog_intercept=1.002555437670879
    df2=500
    df1 = 10**(loglog_slope*np.log10(abs(skew)) + loglog_intercept)

    # sample from F distribution
    fsample = np.sort(stats.f(df1, df2).rvs(size=size))

    # adjust the variance by scaling the distance from each point to the distribution mean by a const
    k1_slope = 0.5670830069364579
    k1_intercept = -0.09239985798819927
    k2_slope = 0.5823114978219056
    k2_intercept = -0.11748300123471256

    scaling_slope = abs(skew)*k1_slope + k1_intercept
    scaling_intercept = abs(skew)*k2_slope + k2_intercept

    scale_factor = (sd - scaling_intercept)/scaling_slope
    new_dist = (fsample - np.mean(fsample))*scale_factor + fsample

    # flip the distribution if specified skew is negative
    if skew < 0:
        new_dist = np.mean(new_dist) - new_dist

    # adjust the distribution mean to the specified value
    final_dist = new_dist + (mean - np.mean(new_dist))

    return final_dist

desired_mean = 7
desired_skew = 1.05
desired_sd = 6

final_dist = createSkewDist(mean=desired_mean, sd=desired_sd, skew=desired_skew, size=1000000)
fig, ax = plt.subplots(figsize=(12,7))
sns.distplot(final_dist, hist=True, ax=ax, color='green', label='generated distribution')
sns.distplot(np.random.choice(final_dist, size=100), hist=True, ax=ax, color='red', hist_kws={'alpha':
ax.legend()
fig = ax.get_figure()
fig.savefig('f_distribution.png')
```

In [ ]:

```
TYPES = [HRB, Normal, Log]

class Env:
```

```
"""Environment class"""
size: int
max_time: int

def __init__(self, size: int, max_time=100, map_type="rand_env", plotting=True):
    self.size = size
    self.map = np.array([[Normal(curr_biomass=random.uniform(B_n_lower, B_n_upper), on_fire=False
                  if (i < self.size//2 + 1 and j < self.size//2 + 1) else  \
                      Log(curr_biomass=random.uniform(B_h_lower, B_h_upper), on_fire=False) \
                if (i > self.size//2 and j < self.size//2 + 1) else \
                      HRB(curr_biomass=random.uniform(B_h_lower, B_h_upper), on_fire=False) \
                if (i < self.size//2 + 1 and j > self.size//2) else \
                      random.choice(TYPES)(curr_biomass=np.random.uniform(0, 50), on_fire=Fal
                      for i in range(self.size)] \
                     for j in range(self.size)], dtype=object) if map_type == "corner" \
    else np.array([[Normal(curr_biomass=random.uniform(B_n_lower, B_n_upper), on_fire=False) \
                  if i < self.size//2 else Log(curr_biomass=random.uniform(B_h_lower, B_h_upp
                                              on_fire=False) for i in range(self.size)] \
                       for _ in range(self.size)], dtype=object) if map_type == "nlcomp" \
    else np.array([[Normal(curr_biomass=random.uniform(B_n_lower, B_n_upper), on_fire=False) \
                      for _ in range(self.size)] \
                      for _ in range(self.size)], dtype=object) if map_type == "nflat" \
    else np.array([[HRB(curr_biomass=random.uniform(B_h_lower, B_h_upper), on_fire=False) \
                      for _ in range(self.size)] \
                      for _ in range(self.size)], dtype=object) if map_type == "hflat" \
    else np.array([[Log(curr_biomass=random.uniform(B_l_lower, B_l_upper), on_fire=False) \
                      for _ in range(self.size)] \
                      for _ in range(self.size)], dtype=object) if map_type == "lflat" \
    else np.array([[random.choice(TYPES)(curr_biomass=np.random.uniform(0, 50), on_fire=False) \
                      for _ in range(self.size)] \
                      for _ in range(self.size)], dtype=object)
    if map_type == "corner":
        self.map[0, 0].set_fire()
        self.map[0, self.size - 1].set_fire()
        self.map[self.size - 1, 0].set_fire()
        self.map[self.size - 1, self.size - 1].set_fire()
    else:
        self.map[self.size//2, self.size//2].set_fire()
    #locations of currently burning cells
    self.curr_burn = set((i, j) for i in range(self.size) \
                      for j in range(self.size) if self.map[i, j].on_fire is True)
    self.max_time = max_time
    self.map_type = map_type
    self.burn_data = [len(self.curr_burn)]
    self.plotting = plotting

def simulate(self):
    t = 1
    while t <= self.max_time:
        curr_add_burn = set() # keep current burned during a single tick in a set so no instantan
        #look at what's burning already since only those cells can change state
        for (i, j) in self.curr_burn:
            #check the adjacent cells to see if they're on fire and set fire
            #LOCAL SPREAD
            if (i - 1 >= 0) and not self.map[i - 1, j].on_fire  \
            and (i - 1, j) not in self.curr_burn: #left
                set_fire_prob = self.map[i, j].l_b() / 4.0
                if random.random() < set_fire_prob:
                    curr_add_burn.add((i - 1, j))
            if (j - 1 >= 0) and not self.map[i, j - 1].on_fire  \
            and (i, j - 1) not in self.curr_burn: #top
                set_fire_prob = self.map[i, j].l_b() / 4.0
                if random.random() < set_fire_prob:
                    curr_add_burn.add((i, j - 1))
            if (i + 1 < self.size) and not self.map[i + 1, j].on_fire  \
            and (i + 1, j) not in self.curr_burn: #right
                set_fire_prob = self.map[i, j].l_b() / 4.0
                if random.random() < set_fire_prob:
                    curr_add_burn.add((i + 1, j))
            if (j + 1 < self.size) and not self.map[i, j + 1].on_fire  \
            and (i, j + 1) not in self.curr_burn: #bottom
                set_fire_prob = self.map[i, j].l_b() / 4.0
                if random.random() < set_fire_prob:
                    curr_add_burn.add((i, j + 1))

            #NONLOCAL SPREAD
            num_nonlocal = self.map[i, j].n_t()
```

```
                for k in range(num_nonlocal):
                    man_dist = math.floor(abs(np.random.choice(final_dist)))
                    x = random.randint(0, man_dist)
                    y = man_dist - x
                    sign = random.randint(0, 3)
                    if sign == 0 and (i + x < self.size) and (j + y < self.size):
                        curr_add_burn.add((i + x, j + y))
                    elif sign == 1 and (i + x < self.size) and (j + y >= 0):
                        curr_add_burn.add((i + x, j - y))
                    elif sign == 2 and (i - x >= 0) and (j + y < self.size):
                        curr_add_burn.add((i - x, j + y))
                    else:
                        if (i - x >= 0) and (j - y >= 0):
                            curr_add_burn.add((i - x, j - y))

            #update current burning
            self.burn_data.append(self.burn_data[-1] + len(curr_add_burn))
            for (i, j) in curr_add_burn:
                self.map[i, j].set_fire()
                self.curr_burn.add((i, j))

            for (i, j) in self.curr_burn:
                self.map[i, j].decay_biomass()

            if self.plotting is True:

                if t % 10 == 0 or t == 1:
                    self.plot_simulations(t=t, plot_type="biomass")
                    print("Saving map for: " + self.map_type + " @ t = " + str(t))

                if t == self.max_time:
                    self.plot_burnt()
                    print("Saving burn graph for: " + self.map_type)

            t += 1

    def get_burn_data(self):
        return self.burn_data

    def populate_state(self, plot_type="terrain"):
        ret = np.array([[0 for _ in range(self.size)] for _ in range(self.size)])
        if plot_type == "burn":
            for i in range(self.size):
                for j in range(self.size):
                    if self.map[i, j].on_fire:
                        ret[i, j] = 0
                    else:
                        ret[i, j] = 1
        elif plot_type =="biomass":
            for i in range(self.size):
                for j in range(self.size):
                    if isinstance(self.map[i, j], Normal):
                        ret[i, j] = 0
                    elif isinstance(self.map[i, j], HRB):
                        ret[i, j] = 1
                    elif isinstance(self.map[i, j], Log):
                        ret[i, j] = 2
                    if self.map[i, j].is_burnt:
                        ret[i, j] = 3
                    bm = self.map[i, j].curr_biomass
                    if self.map[i, j].on_fire:
                        if 0 < bm and bm <= 2:
                            ret[i, j] = 4
                        elif 2 < bm and bm <= 4:
                            ret[i, j] = 5
                        elif 4 < bm and bm <= 6:
                            ret[i, j] = 6
                        elif 6 < bm and bm <= 8:
                            ret[i, j] = 7
                        elif 8 < bm and bm <= 10:
                            ret[i, j] = 8
                        elif 10 < bm and bm <= 11:
                            ret[i, j] = 9

        else:
            for i in range(self.size):
```

```
                for j in range(self.size):
                    if isinstance(self.map[i, j], Normal):
                        ret[i, j] = 0
                    elif isinstance(self.map[i, j], HRB):
                        ret[i, j] = 1
                    elif isinstance(self.map[i, j], Log):
                        ret[i, j] = 2
            return ret

        def plot_simulations(self, t=0, plot_type="terrain"):
            curr_state = self.populate_state()
            terrain_colors = colors.ListedColormap(['#24422c', '#4a8a5b', '#aab560'])
            if plot_type == "burn":
                curr_state = self.populate_state("burn")
                terrain_colors = colors.ListedColormap(['#000000', '#4a8a5b'])
            elif plot_type == "biomass":
                curr_state = self.populate_state("biomass")
            from matplotlib.colors import LinearSegmentedColormap
            cmap_reds = plt.get_cmap('Reds')
            num_colors = 9
            col = ['#4a8a5b', '#24422c', '#97DB46', '#C5A7A5', '#f3c222', '#f7ae26', \
                   '#ff7831', '#ff5737', '#e4514e']
            cmap = LinearSegmentedColormap.from_list('', col, num_colors)
            ax = sns.heatmap(curr_state, cmap=cmap, vmin=0, vmax=num_colors, square=True, cbar=False, \
                             xticklabels=False, yticklabels=False)
            plt.savefig("./images/" + self.map_type + "/t_1000" + str(t) + ".jpeg")

        def plot_burnt(self):
            x = list(range(self.max_time + 1))
            sns.set_style(style='whitegrid')
            ax = sns.lineplot(x=x, y=self.burn_data, color='#e4514e')
            ax.set(xlabel='Time', ylabel='Cells burnt')
            plt.xlim(0)
            plt.ylim(0)
            plt.savefig("./images/" + self.map_type + "/burn_plot" + ".jpeg")
```

In [ ]:

```
test = Env(size=1000, map_type="corner")
test.simulate()
```

In [ ]:

```
test = Env(ENV_SIZE, map_type="hflat", max_time=100, plotting=False)
test.simulate()
hrb_burn_data = test.get_burn_data()
test = Env(ENV_SIZE, map_type="lflat", max_time=100, plotting=False)
test.simulate()
log_burn_data = test.get_burn_data()
test = Env(ENV_SIZE, map_type="nflat", max_time=100, plotting=False)
test.simulate()
norm_burn_data = test.get_burn_data()

t = list(range(100 + 1))
burn_data = pd.DataFrame()
burn_data['Time'] = t
burn_data['HRB'] = hrb_burn_data
burn_data['Log'] = log_burn_data
burn_data['Norm'] = norm_burn_data

sns.set_style(style='whitegrid')
ax = sns.lineplot(x='Time', data=pd.melt(burn_data, ['Time'], var_name='Terrain', value_name='Cells Bu
                  y='Cells Burnt', hue='Terrain', palette=['#BF616A', '#A3BE8C','#81A1C1'])
ax.set(xlabel='Time', ylabel='Cells burnt')
ax.set_yscale('log')
plt.xlim(0)
plt.ylim(0)
plt.savefig("./images/burn_comparison.jpeg")
```

In [ ]:

```
test = Env(ENV_SIZE, map_type="hflat", max_time=10)
test.simulate()
```

Simulations on Preset Environment Generations

To show effectiveness and compare different anti-wildfire techniques

In [ ]:

```
#normal terrain only
```

```
nflat = Env(ENV_SIZE, map_type="nflat")
nflat.simulate()
```

In [ ]:
```
#HRB terrain only
hflat = Env(ENV_SIZE, map_type="hflat")
hflat.simulate()
```

In [ ]:
```
#logged terrain only
lflat = Env(ENV_SIZE, map_type="lflat")
lflat.simulate()
```

In [ ]:
```
#Close ups of spotting
close_up = Env(100, map_type="nflat", max_time=20)
close_up.simulate()
```

10 References

References

[1] Jones, M. W., et. al. (n.d.). *Keep up with Science.* ScienceBrief. Retrieved November 14, 2021, from https://sciencebrief.org/briefs/wildfires.

[2] Deb, P., Moradkhani, H., Abbaszadeh, P., Kiem, A. S., Engström, J., Keellings, D., amp; Sharma, A. (2020, November 11). *Causes of the widespread 2019–2020 Australian bushfire season.* AGU Journals. Retrieved November 14, 2021, from https://agupubs.onlinelibrary.wiley.com/doi/10.1029/2020EF001671.

[3] Stone, C., et. al. (n.d.). Forest Harvest can Increase Subsequent Forest Fire Severity. *Proceedings of the Second International Symposium on Fire Economics, Planning, and Policy: A Global View*, 525–534. Retrieved November 14, 2021, from fs.fed.us/psw/publications/documents/psw_gtr208en/psw_gtr208en_525-534_stone.pdf.

[4] Hall, J. (2015). Long-distance spotting potential of bark strips of a ribbon gum (Eucalyptus viminalis). International Journal of Wildland Fire. Retrieved from https://doi.org/10.1071/WF15031.

[5] Rands, B. (2019). *Making Forests Stronger through Active Management.* Retrieved from https://www.fs.usda.gov/features/making-forests-stronger-through-active-management.

[6] Price, O. (n.d.). Biogeographical variation in the potential effectiveness of prescribed fire in south-eastern Australia. *Faculty of Science, Medicine and Health - Papers: Part A* https://doi.org/http://dx.doi.org/10.1111/jbi.12579

[7] Hislop, S., Stone, C., Haywood, A., amp; Skidmore, A. (2020). The effectiveness of fuel reduction burning for wildfire mitigation in sclerophyll forests. *Australian Forestry, 83*(4), 255–264. https://doi.org/10.1080/00049158.2020.1835032

[8] Martin, J., amp; Hillen, T. (2016). The Spotting Distribution of Wildfires. Applied Sciences. https://doi.org/10.3390/app6060177

[9] McCormick, B. (n.d.). Is Fuel Reduction Burning the Answer? *Current Issues Brief.* Retrieved November 14, 2021.

[10] Poe, B. (2019, September 26). *How can I use skewnorm to produce a distribution with the specified skew?* Stack Overflow. Retrieved November 14, 2021, from https://stackoverflow.com/questions/49801071/how-can-i-use-skewnorm-to-produce-a-distribution-with-the-specified-skew/5811185958111859.