

# AV2 PROGRAMAÇÃO

Líder: Maria Eduarda

Grupo: Alex, Beatriz, Bianca Moreno, Daniel, Davi Boechat, Fernanda, Gabriely, Laura, Lucía, Maria Fernanda

# Implementações

## Atributos na Classe TreeCell:

- Atributo resistência da árvore ao fogo
- Função Vento

## Novas classes:

- Person
- Cidades
- Clima
- Grama
- Bombeiro Aéreo
- Cidadão

arquivo agent.py

```
class TreeCell(mesa.Agent):  
    """  
    A tree in the forest.  
    A árvore da floresta, com a probabilidade de sobrevivência e influência do vento.  
    """
```

```
class CityCell(mesa.Agent):  
    """  
    A cidade na floresta, com alerta de evacuação baseado no fogo.  
    """  
  
    def __init__(self, pos, model, condition="City"):  
        super().__init__(pos, model)  
        self.pos = pos  
        self.condition = "City" # Condições possíveis: "City", "Evacuated"  
        self.alert = False # Flag para indicar o alerta de evacuação
```

```
class Person(mesa.Agent):  
    #Classe para as pessoas envolvidas na simulação  
    def __init__(self, pos, model, resistencia_fogo = 0, resistencia_fumaca = 0.5):  
        """  
        Cria pessoas  
        Args:  
            pos: coordenada atual da pessoa no grid.  
            model: standard model reference for agent.  
        """
```

# TreeCell - Probabilidade de Sobrevivência

- O que é?

A probabilidade de sobrevivência de uma árvore define a chance de uma árvore não ser consumida pelo fogo durante a propagação do incêndio florestal no modelo.

- Como funciona?
  - A árvore possui uma chance de sobrevivência definida por um valor entre  $[0, 1]$ .
- Qual é a fórmula utilizada?
  - Se `random.random() > probabilidade_de_sobrevivencia`: a árvore é consumida pelo fogo. Caso contrário, a árvore sobrevive.

## Agent (TreeCell):

- Atributo: **prob\_de\_sobrevivencia** -> Armazena a probabilidade de uma árvore sobreviver ao fogo.
- Método step: Quando uma árvore pega fogo, o método verifica a probabilidade de sobrevivência em cada ciclo para decidir se a árvore será consumida ou não.
  - Interação:
    - Quando o fogo atinge a árvore, ela verifica sua chance de sobrevivência.
    - Se a árvore sobrevive, mantém a condição "Fine". Caso contrário, muda a condição para "Burned Out", indicando que a árvore foi consumida pelo fogo

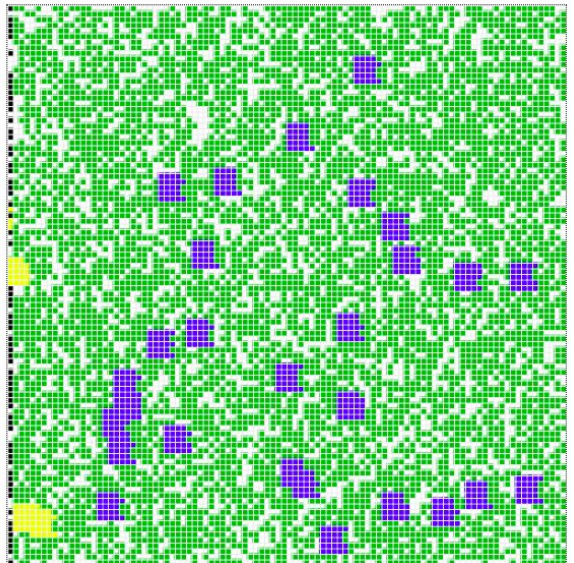
**Model (ForestFire):** **prob\_de\_sobrevivencia** é inicializada como 0.5

**Server (Visualização):** Parâmetro **prob\_de\_sobrevivencia** para escolha em slider

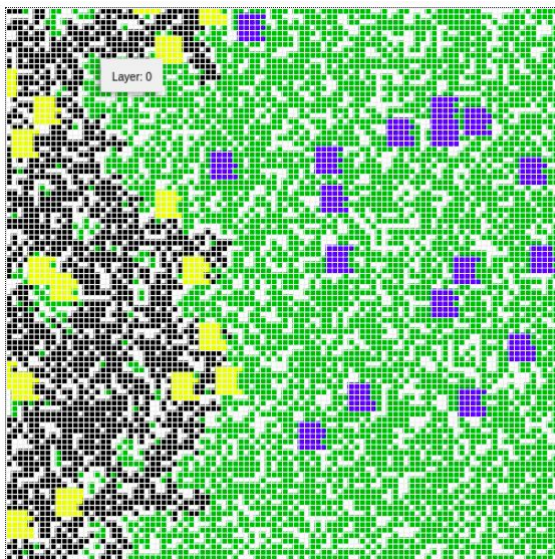
> Segue abaixo exemplos de simulações mudando apenas o parâmetro “survival probability”.

\* Cada simulação durou 20 segundos.

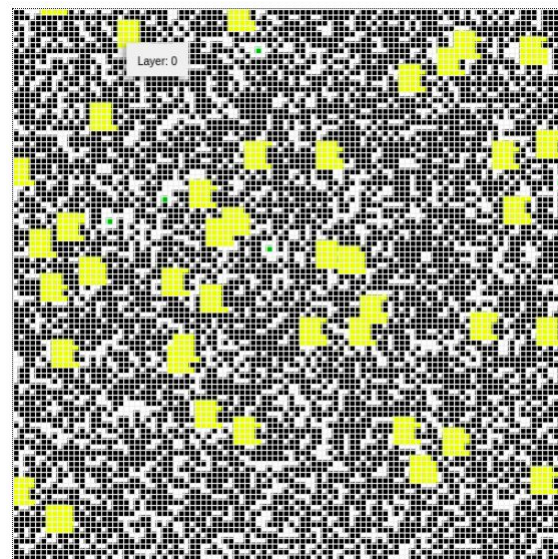
survival probability = 1



survival probability = 0,5



survival probability = 0



# TreeCell (Função Vento)

**Objetivo:** Simular o impacto do vento na propagação do fogo em um cenário de incêndio florestal.

- Se o **vento** está vindo do Norte, por exemplo, árvores que estão ao sul da árvore em questão têm sua probabilidade de pegar fogo ajustada.
- O código também lida com diferentes direções de **vento** (Norte, Sul, Leste, Oeste) e ajusta a probabilidade com base na posição relativa das árvores em relação à direção do vento.

# Implementação no código ForestFire

## **Agent (TreeCell):**

- Função `ajusta_probabilidade_por_vento`: Ajusta a probabilidade de uma árvore pegar fogo com base na direção do vento.

## **Model (ForestFire):**

- Direção do vento é passada como parâmetro na inicialização(Norte, Sul, Leste, Oeste, Sem direção).

## **Server (Visualização):**

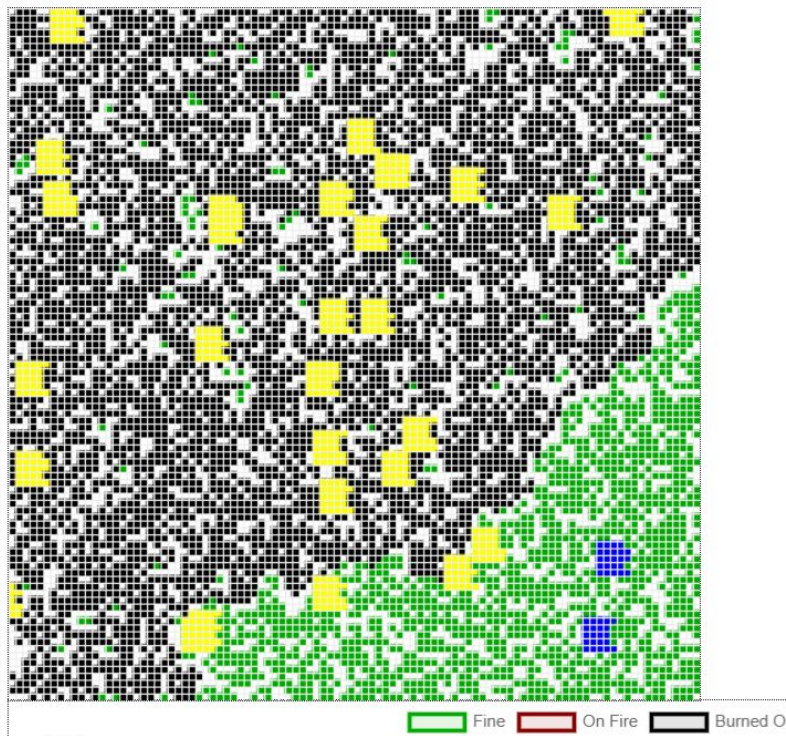
- Parâmetro “vento” adicionado como uma opção de escolha no servidor (Choice).

Usuário pode selecionar a direção do vento, que será passada no modelo.

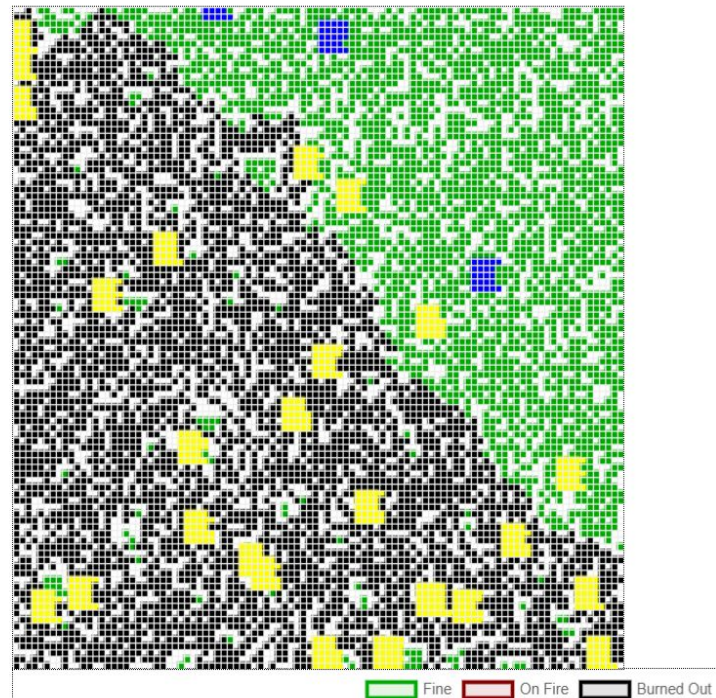


# Vento Norte e Sul:

Current Step: 59



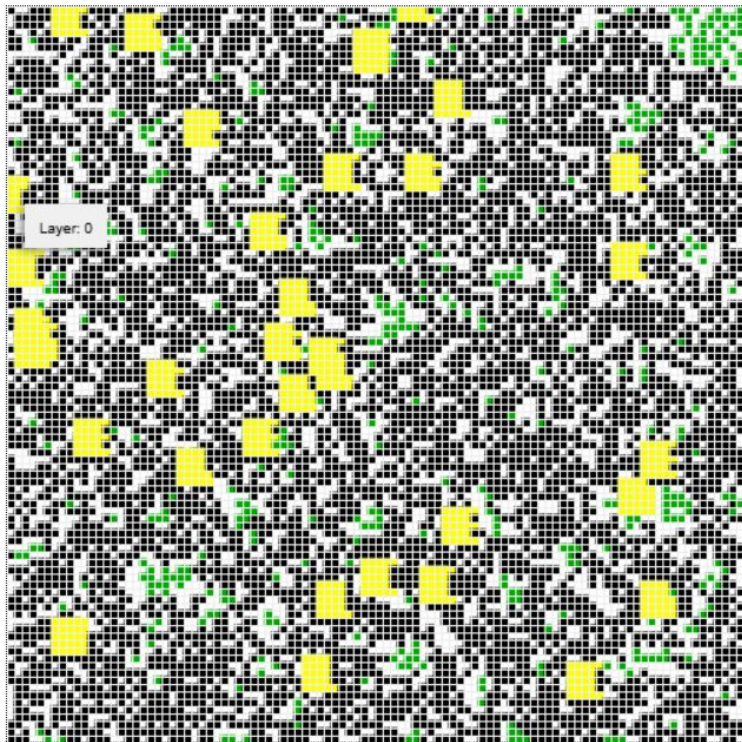
Current Step: 57



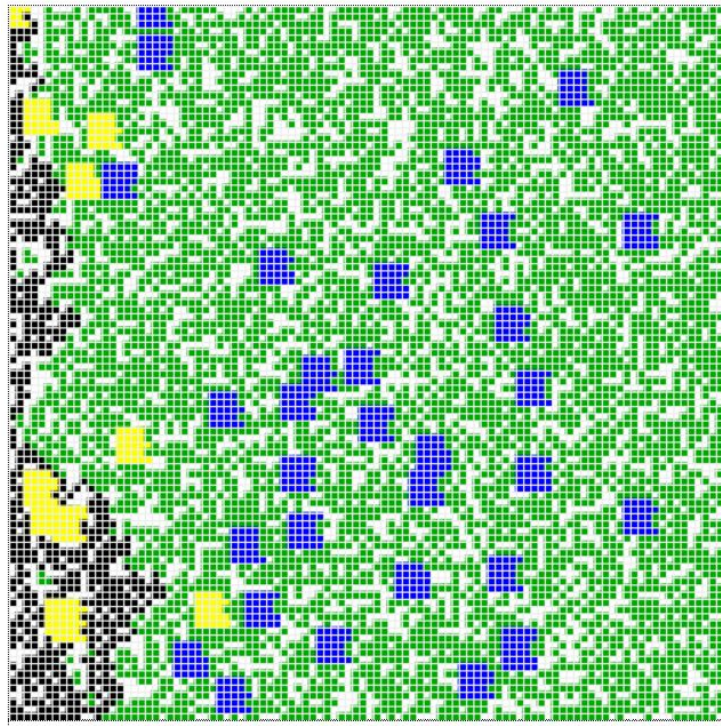


# Vento Leste e Oeste

Current Step: 40



Current Step: 23

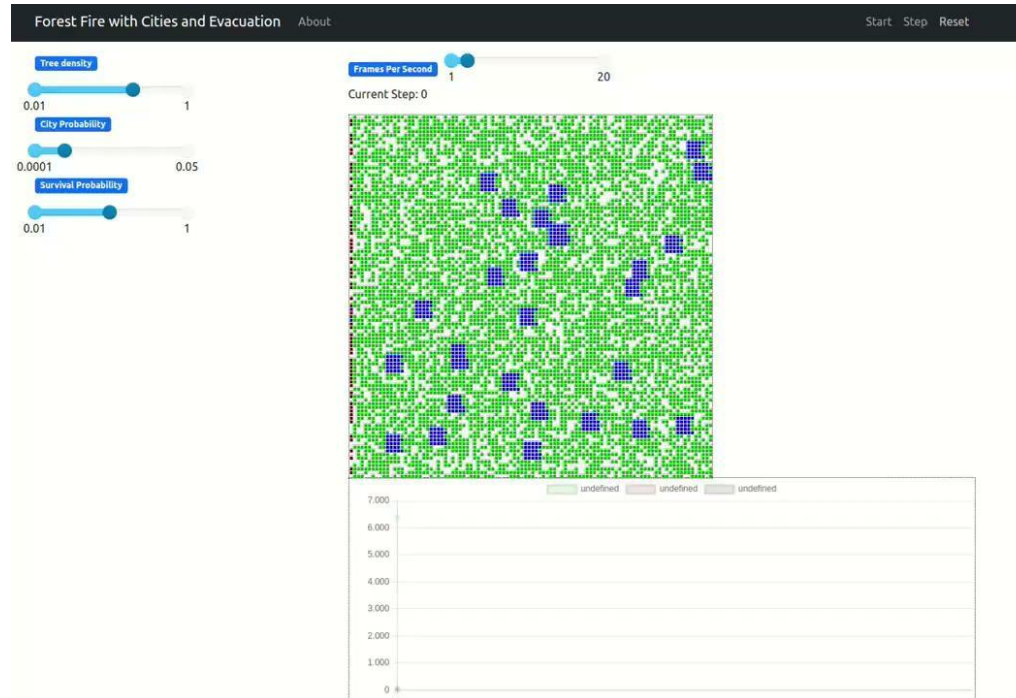


# Class CityCell

**Objetivo:** Simular cenários emergentes de evacuação

- As cidades são adicionadas no Grid com uma probabilidade/incidência entre  $[0, 1]$
- As cidades tem um tamanho de 5 células
- Se o **fogo** está se aproximando da cidade, ela é evacuada
- Definimos o alerta de aproximação do fogo para quando a distância do fogo for menor que 10 células

# Class CityCell



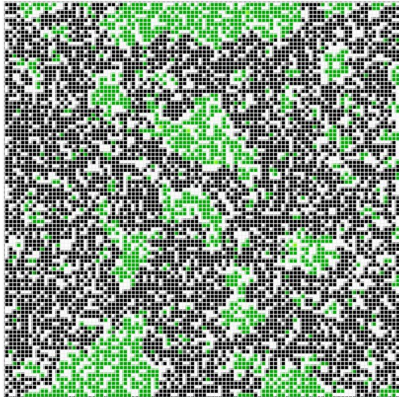
# Class GrassCell

**Objetivo:** Adicionar uma nova forma de propagação de fogo

- a grama terá menor probabilidade de queimar seus vizinhos (15%)

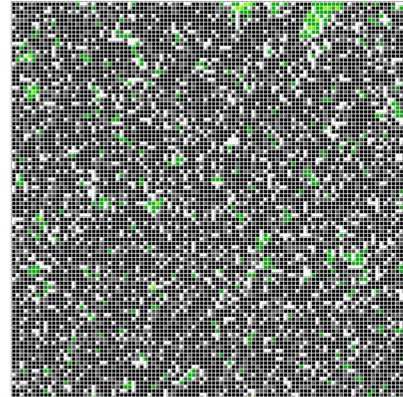
após incêndio sem grama:

Current Step: 92



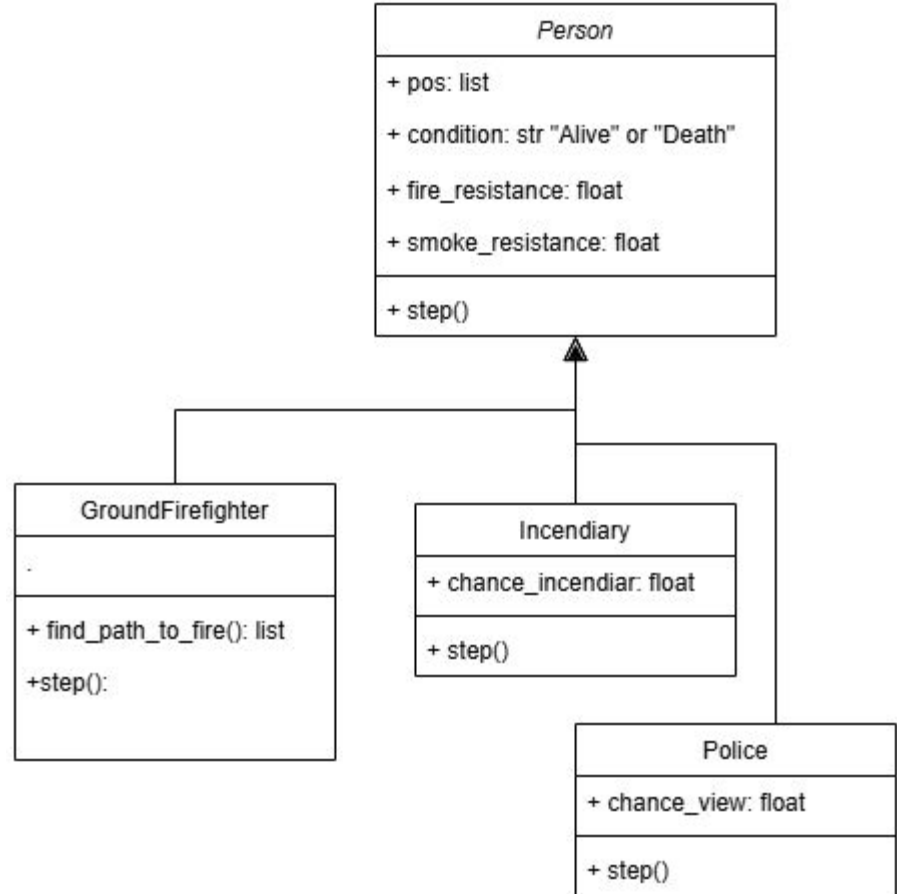
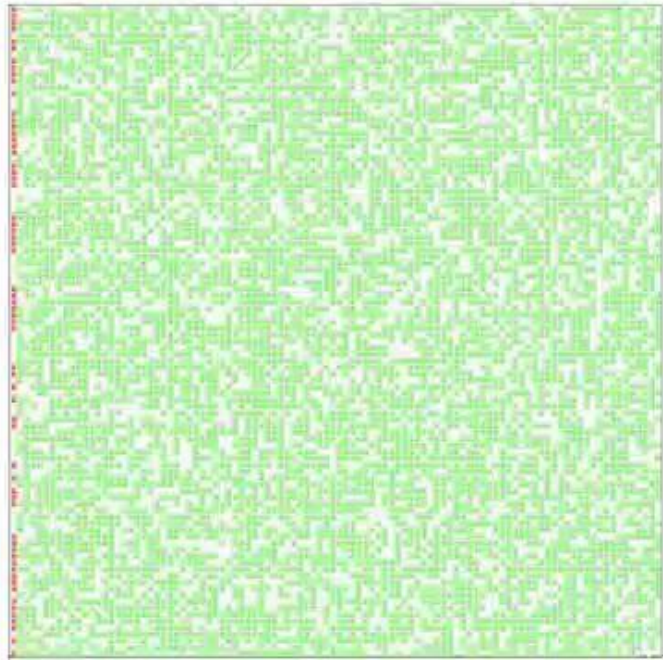
após incêndio com grama:

Current Step: 62





# Class Person





# Herança Bombeiro



**Objetivo:** Implementar agentes que apaguem o fogo, minimizando a propagação em um incêndio florestal.

- **Herança:** O agente bombeiro herda da classe Pessoa, que define atributos como posição, resistência ao fogo(1.0) e fumaça.(1.0)
- **Interação:** O bombeiro altera o estado de árvores em chamas (TreeCell), localizado por uma BFS, para "Fire Off"

# Bombeiro Terrestre × Bombeiro Aéreo

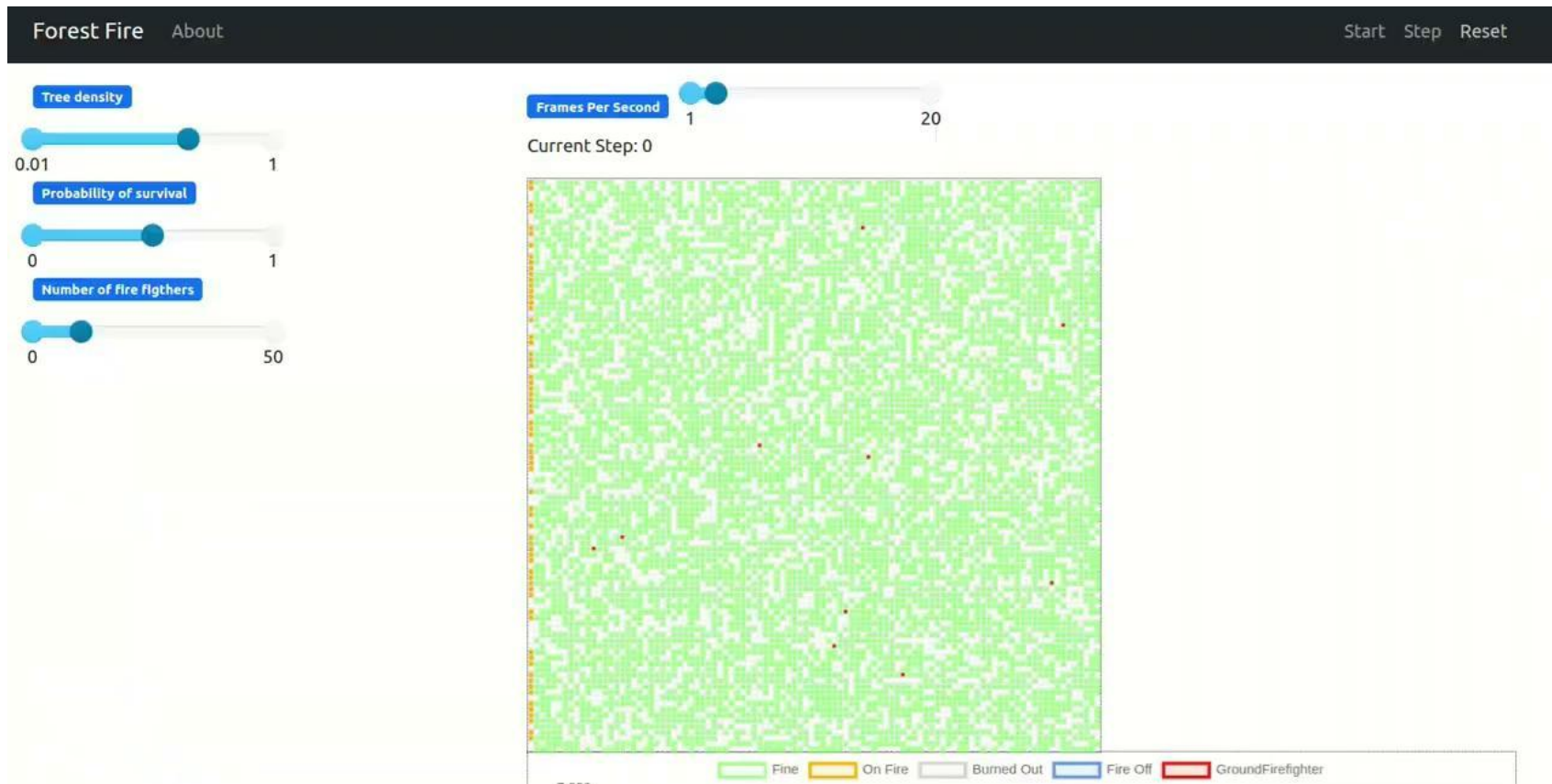
Um agente no solo que apaga incêndios localizados.

- bombeiro procura a árvore em chamas mais próxima a cada step
- Move-se para a árvore mais próxima e apaga o fogo.
- Explora áreas não visitadas.

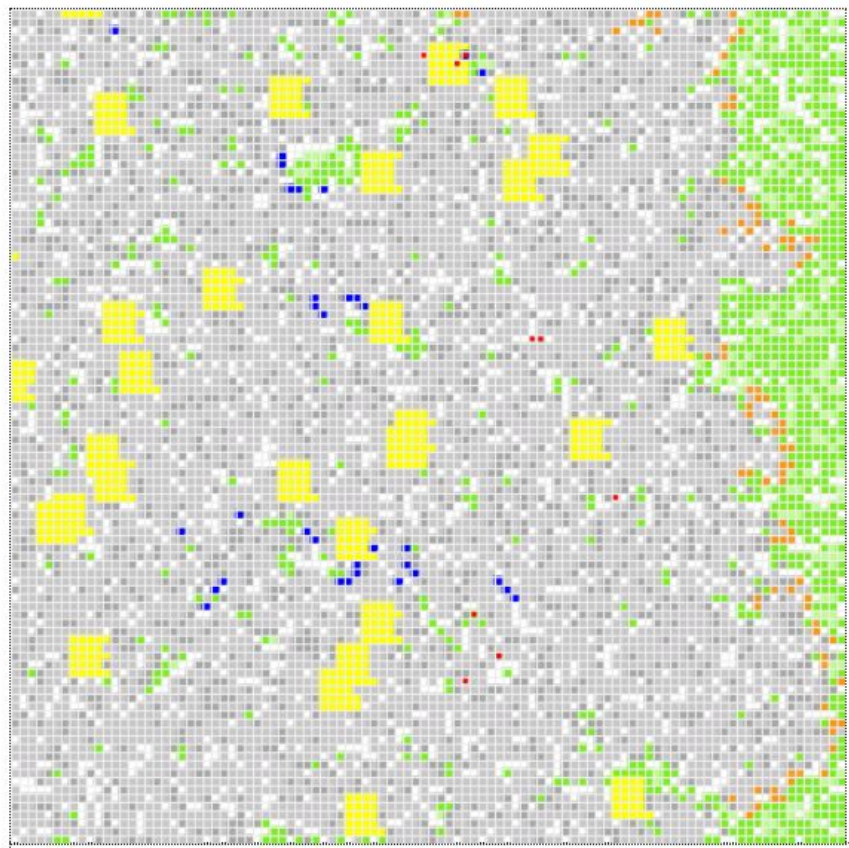
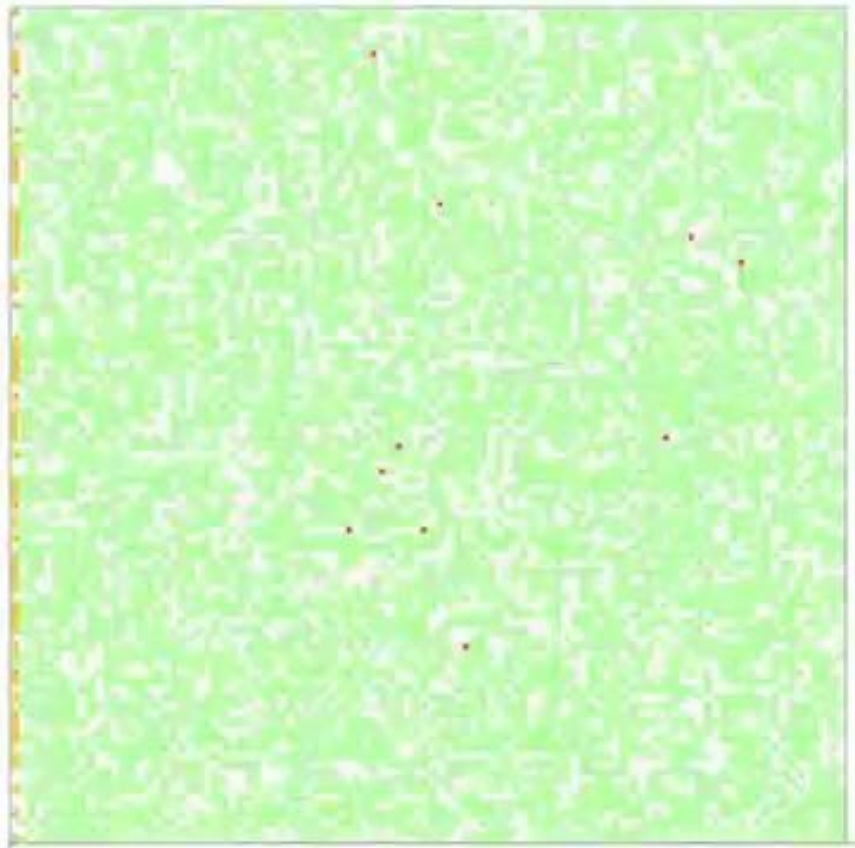
Agente que cobre áreas maiores para conter rapidamente o fogo.

- Sobrevoa células vizinhas, apagando árvores em chamas.
- Move-se para as árvores mais próximas ao detectar incêndios ativos.

# BOMBEIRO IDEAL



# BOMBEIRO IDEAL x BOMBEIRO REAL



# Modelo e Visualização

## Agent (TreeCell e GroundFirefighter):

- O TreeCell recebe uma nova condição: “Fire Off”
- Implementação da classe GroundFirefighter

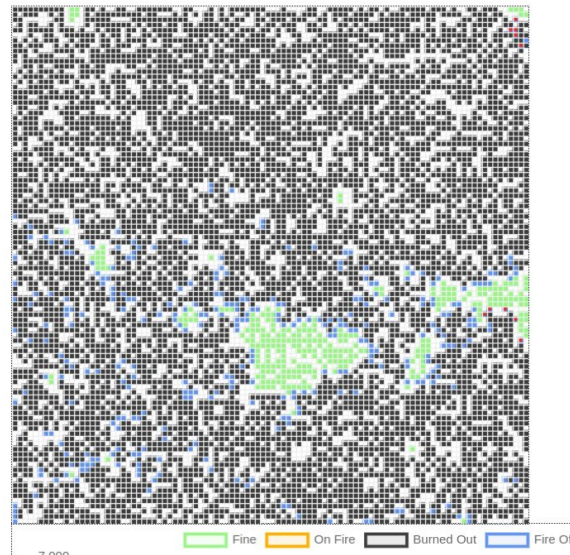
## Model (ForestFire):

- O bombeiro é colocado em uma posição x, y aleatória.
- Inicializada com 10 bombeiros

## Server (Visualização):

- Slider para a quantidade de bombeiros

Melhor Resultado:





# Bombardeiro, Madeireiro e Policial

**Policial:** pessoas que caminham pela floresta em busca dos criminosos(Madeireiro e Bombardeiro). Seu objetivo, prendê-los.

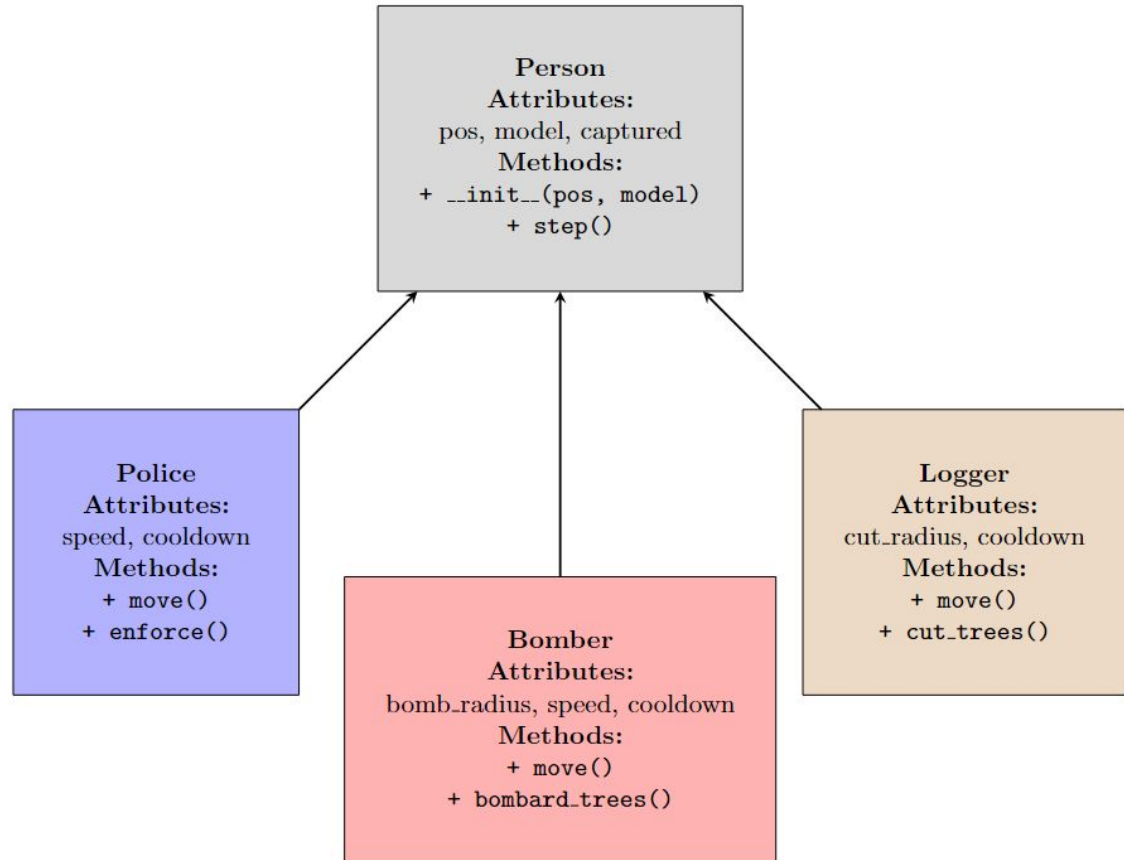
**Madeireiro:** Criminosos que caminha pela floresta desmatando-a.

**Bombardeiro:** Criminoso que caminha pela floresta soltando bombas que queimam imediatamente as árvores ao redor .

# Bombardeiro, Madeireiro e Policial

## Herança: Os 3

herdam da classe Person, que define atributos como por exemplo posição e condição (livre ou capturada, no caso dos criminosos).



# Modelo Policial

## **Agent (Policial):**

O Policial se move pela área utilizando um algoritmo de BFS para encontrar o criminoso mais próximo e persegui-lo e quando alcançar mudar o parâmetro capture para True e seguir patrulha.

## **Model (ForestFire):**

O número de Policiais é passado como parâmetro na inicialização do modelo. A posição dos Policiais é gerenciada dentro do modelo.

## **Server (Visualização):**

Adiciona o parâmetro "Número de Policiais" como um controle deslizante no servidor. O usuário pode ajustar a quantidade de Policiais.

# Modelo Bombardeiro

## **Agent (Bomber):**

O Bombardeiro se move aleatoriamente e pode bombardear árvores dentro de um raio alterável.

## **Model (ForestFire):**

O número de Bombardeiros é passado como parâmetro na inicialização do modelo. A ação de bombardeio é gerenciada dentro do modelo.

## **Server (Visualização):**

Adiciona o parâmetro "Número de Bombardeiros" como um controle deslizante no servidor. O usuário pode ajustar a quantidade de Bombardeiros.

# Modelo Madeireiro

## **Agent (Logger):**

Função `cut_trees` e função `step`: O Madeireiro se move aleatoriamente e corta árvores saudáveis dentro de um raio alterável.

## **Model (ForestFire):**

Número de Madeireiros é passado como parâmetro na inicialização. A ação de corte de árvores é gerenciada no modelo.

## **Server (Visualização):**

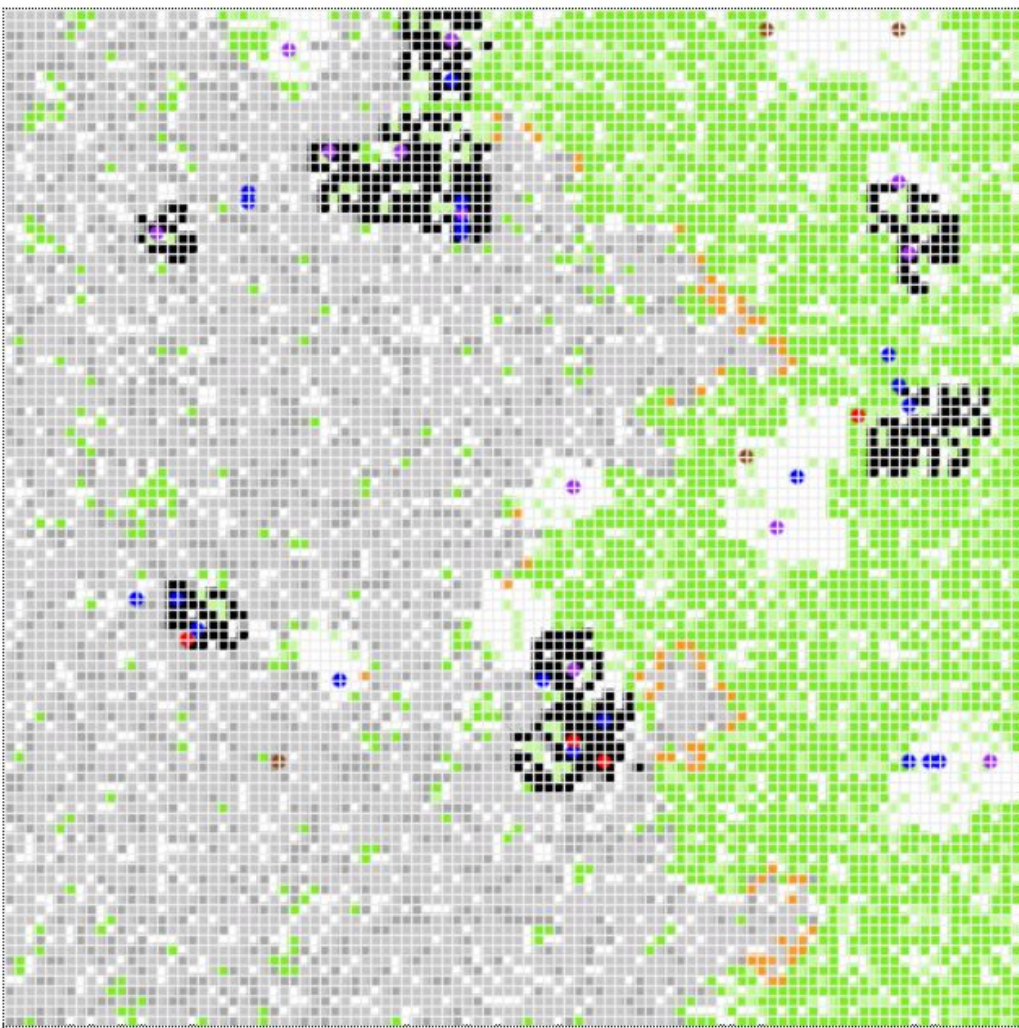
Parâmetro "Número de Madeireiros" adicionado como um controle deslizante (Slider) no servidor. Usuário pode ajustar a quantidade de Loggers.



# Visualização

Legenda:

- Policial
- Bombardeiro
- Madeireiro
- Criminoso capturado



# Exemplo Policiais x Criminosos

Forest Fire com Cidades e Pessoas About

Start Step Reset

Número de Policiais



Número de Bombardeiros



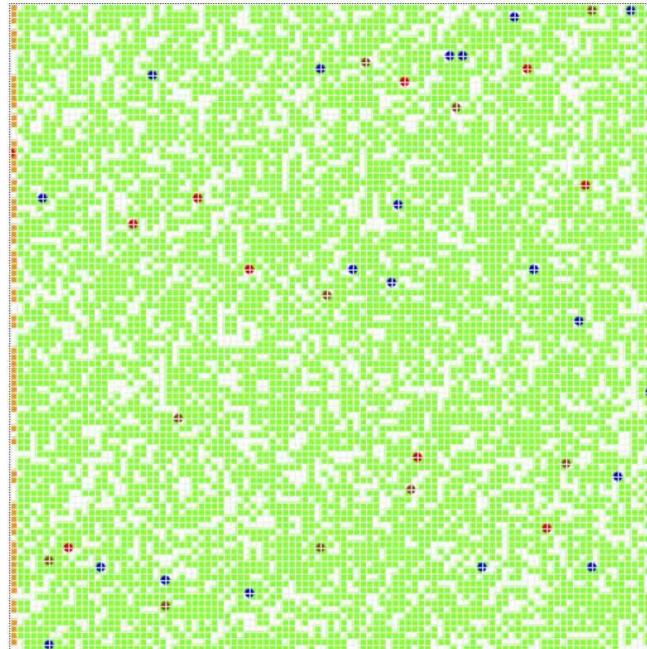
Número de Madeiros



Frames Per Second



Current Step: 0



7.000

Fine On Fire Burned Out Fire Off Bombed Cut

# Herança Cidadão

**Objetivo:** Simular a fuga das pessoas para cidades próximas em caso de incêndio.

- **Herança:** herdam as cores da classe Pessoa, mas não tem resistência ao fogo (0.0) e suportam moderadamente fumaça (0.5)
- **Método específico:** Busca pela cidade segura mais próxima

## Método Principal: `step()`

- Verifica se o cidadão está em uma cidade evacuada.
- Utiliza **Busca em Largura (BFS)** para encontrar a cidade segura mais próxima.
- Move-se passo a passo em direção ao alvo.
- Caso pego por fogo ou fumaça:
  - Atualiza o estado para `"Dead"`.
  - Remove o cidadão da dinâmica do modelo.

# Class Clima

- **Objetivo:** A classe clima tem como objetivo monitorar precipitação, pressão, umidade e temperatura da floresta.
- Esses fatores dependem da porcentagem da floresta que está em chamas. Essa porcentagem é calculada pela função `verificar_fogo()`

# Class Chuva

- **Objetivo:** A classe tem como objetivo implementar chuva que possivelmente apaga o incêndio de algumas árvores. Ela acontece em locais aleatórios e depende da precipitação.
- A chuva tem chance de apagar o fogo dependendo das suas redondezas e intensidade da chuva.

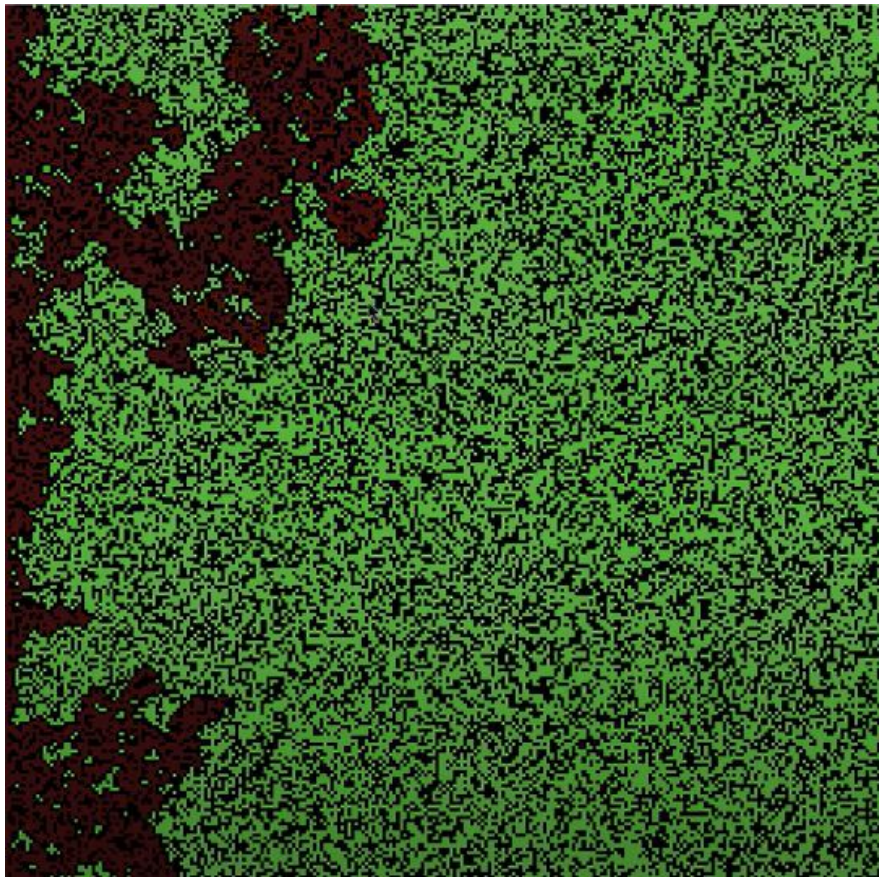
# Estatísticas

Considerando apenas a presença de árvores idênticas no grid (251x251), o modelo apresenta uma transição de fase entre 57% e 62% de densidade arbórea:

- Com 57%, a chance do fogo atravessar o grid é quase nula, com não mais de 15% de queima
- Com 62%, a chance do fogo atravessar o grid é quase certa, com não menos de 80% de queima

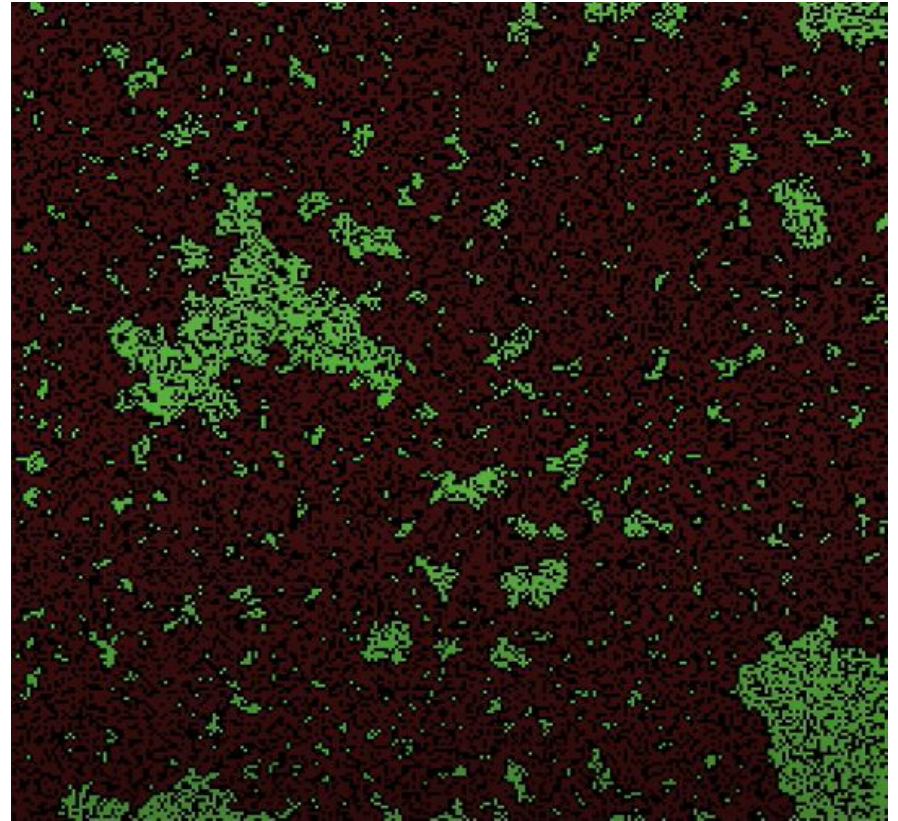
```
"BurnedFraction": lambda m: m.count_type(m, "Burned Out") / (m.grid  
    .width * m.grid.height)
```





density == 0.57

density == 0.62

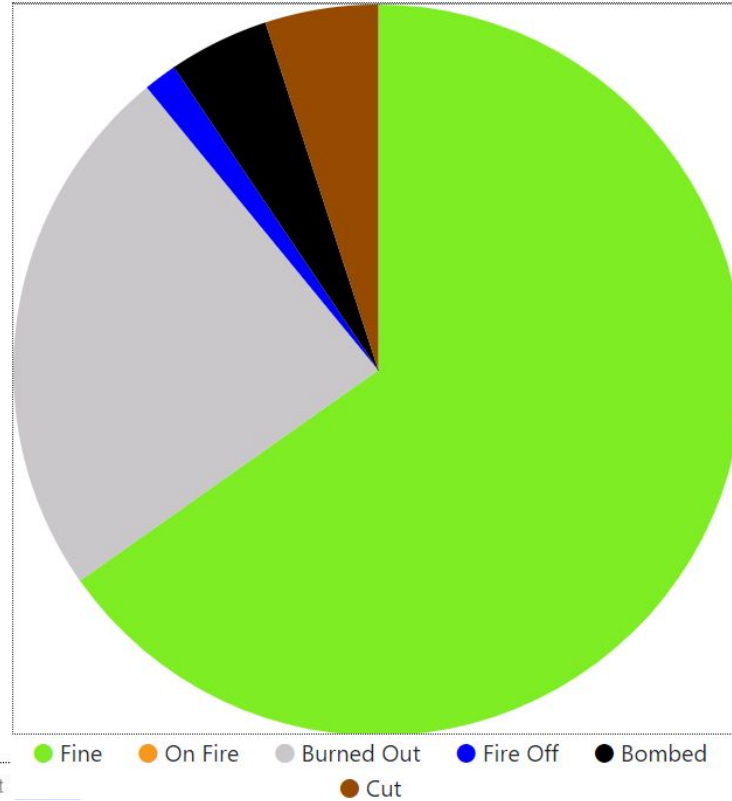
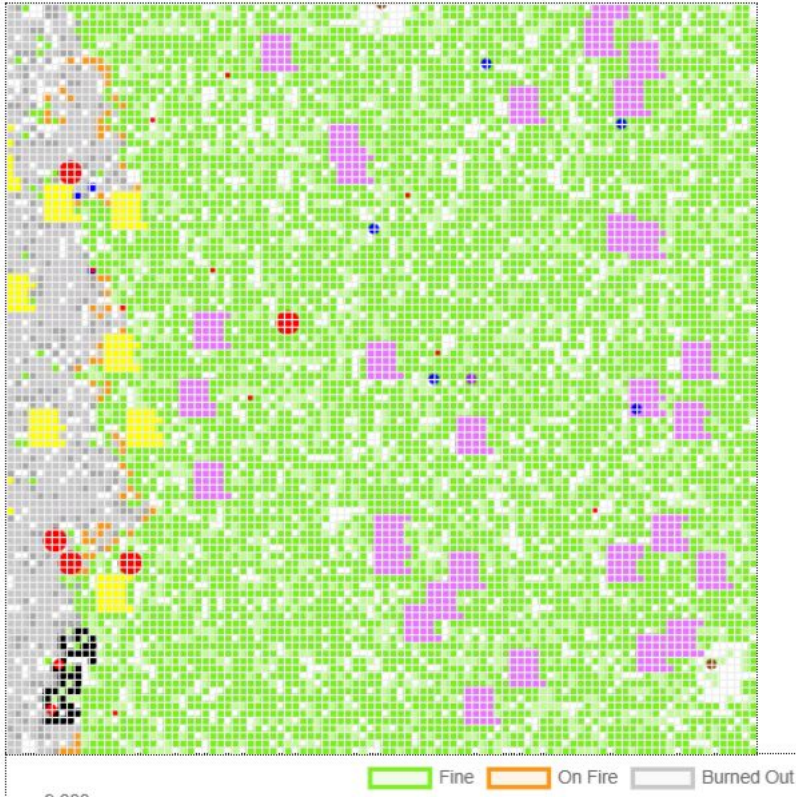


# Desafios

- Multigrid
- RandomActivation



# Resultados





Densidade de Árvore



Densidade de Cidades



Sobrevivência da grama



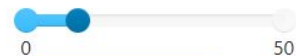
Resistência da árvore ao fogo



Direção do vento

Sem direção ▾

Número de bombeiros terrestres



Número de helicópteros



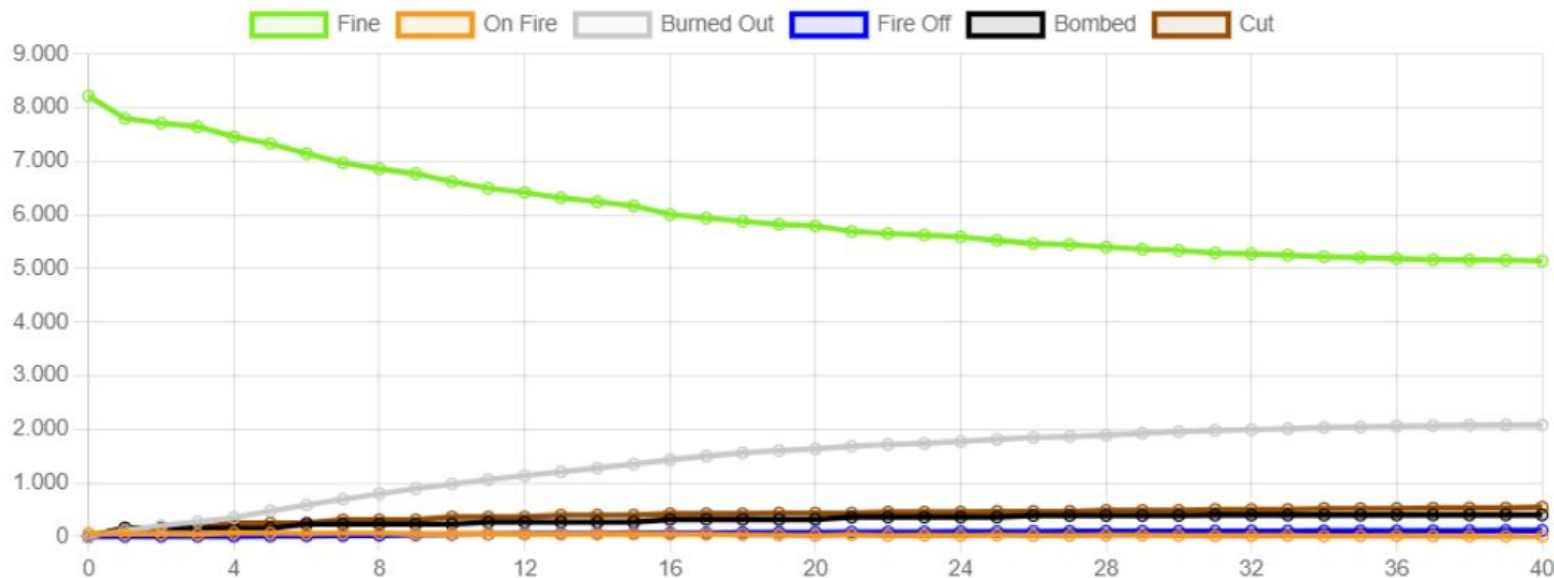
Número de Policiais



Número de Bombardeiros



Número de Madeireiros



# Agradecimentos

Ao professor Emílio e aos nossos monitores.

Ao Vinicius Maciel e Gustavo pela ajuda no código

ESTAMOS ABERTOS A PERGUNTAS!