

# Simuler les feux de forêt

Victor Sarrazin

10 mai 2025

## Introduction

Avec le changement climatique, les feux de forêt sont de plus en plus fréquents et dévastateurs, à l'image de ceux en Californie en janvier 2025.

Dans ce cadre, les modélisations informatiques des feux de forêt permettent de simuler leur évolution et ainsi de prévoir les zones à risque. Elles offrent également la possibilité de tester l'impact de certaines transformations sur ces incendies, afin d'identifier des moyens de réduire les conséquences des catastrophes sans dénaturer les forêts.

## 1 Automates cellulaires

### 1.1 Généralités

Pour réaliser nos modélisations, nous allons utiliser des automates cellulaires. Ces outils permettent de représenter des systèmes avec des interactions locales entre les éléments qui les constituent. Les automates cellulaires ont notamment été popularisés avec *Le Jeu de la Vie* de *Conway*.

**Définition 1.** Un automate cellulaire est la donnée d'un triplet  $(Q, M, f)$  avec :

- $Q$  un ensemble d'états
- $M$  une matrice de taille  $n \times m$ , où chaque  $m_{i,j}$  représente une cellule de la grille
- $f : \mathcal{M}_{n,m}(Q) \longrightarrow \mathcal{M}_{n,m}(Q)$  une *fonction de transition* qui à une grille renvoie la grille suivante

La définition de la fonction de transition dépend donc du type de voisinage considéré. Il existe deux principaux types de voisinages : celui de *von Neumann* et celui de *Moore*. Nous avons choisi d'utiliser le voisinage de *Moore* pour nos modélisations pour prendre en compte le plus d'interactions possibles, celui de *von Neumann* étant plus limitatif, notamment en ce qui concerne les interactions diagonales.

**Définition 2.** Le voisinage de *Moore* est composé de la cellule centrale et de ses 8 voisins adjacents (horizontalement, verticalement et diagonalement).

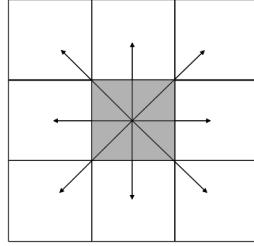


FIGURE 1 – Voisinage de Moore

Il serait possible dans une optique d’avoir des modèles encore plus précis d’utiliser des voisinages de *Moore* étendus, avec plusieurs rayons de voisins. Cependant par souci de simplicité nous nous sommes restreints à un rayon.

## 1.2 Représentation d’un forêt

Afin de représenter une forêt avec un automate cellulaire, nous avons défini une liste d’états que les cellules pouvaient prendre.

**Définition 3.** Les états possibles sont :

- Arbres
- Arbres denses
- Champs
- Feu
- Case brûlée\*
- Eau\*

A noter que les cases (\*) ne peuvent brûler.

Nous avons choisi de travailler avec une matrice de taille  $256 \times 256$  pour des raisons techniques : au-delà le temps de calcul devenait trop long, et l’ajout de lignes/colonnes ne représentait pas un intérêt suffisant au regard du temps de calcul que cela impliquait.

## 2 Modèle d’Alexandridis

Dans un premier temps, il est possible de réaliser une modélisation simple des feux de forêt en associant à chaque cellule une probabilité fixe de s’enflammer si l’un de ses voisins est en feu. Une telle tentative a donné des résultats mitigés. C’est pourquoi l’objectif de cette deuxième

partie est de présenter un modèle plus avancé de modélisation des feux de forêts, appelé modèle d'*Alexandridis*.

## 2.1 Présentation du modèle

Le modèle d'*Alexandridis* prend en compte divers phénomènes pour simuler la propagation des incendies. Par la suite, nous avons choisi de nous concentrer sur la densité de végétation, le type de végétation ainsi que le vent dans la zone. Le modèle original prend également en considération l'élévation du terrain.

**Proposition 1.** On utilise les règles de transition suivantes, pour  $(i, j) \in \mathbb{N}^2$ ,  $t \in \mathbb{N}$  :

- Si  $m_{i,j}(t) = \text{feu}$  alors  $m_{i,j}(t+1) = \text{brulé}$
- Si  $m_{i,j}(t) = \text{feu}$  alors  $m_{i\pm 1, j\pm 1}(t+1) = \text{feu}$  avec une probabilité  $p_b$
- Si  $m_{i,j}(t) = \text{brulé}$  alors  $m_{i,j}(t+1) = \text{brulé}$

**Proposition 2.** La probabilité  $p_b$  qu'une cellule brûle est définie par  $p_b = p_h(1+p_{veg})(1+p_{den})p_{vent}$  avec  $p_h = 0.27$  une constante, et  $p_{veg}$  et  $p_{den}$  des coefficients relatifs au type de cellule.

	$p_{veg}$	$p_{den}$
Arbres	0.3	0.3
Arbres denses	0.3	0
Champs	-0.1	0

FIGURE 2 – Probabilités  $p_{veg}$  et  $p_{den}$  selon le type de végétation

**Proposition 3.** La probabilité  $p_{vent}$  liée au vent est définie par

$p_{vent} = \exp(0.045v) \times \exp(0.131v \times (\cos(\theta) - 1))$  avec  $\theta$  l'angle entre la propagation du feu et la direction du vent, et  $v$  la vitesse du vent (en  $m/s$ )

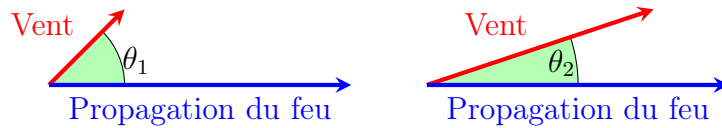


FIGURE 3 – Configurations de vent

On a  $\theta_1 > \theta_2$  et une vitesse plus élevée dans la seconde situation, ainsi la probabilité  $p_{vent}$  sera plus élevée dans la seconde situation.

## 2.2 Résultats

Comparons d'abord les résultats de la modélisation naïve avec ceux du modèle d'*Alexandridis* :

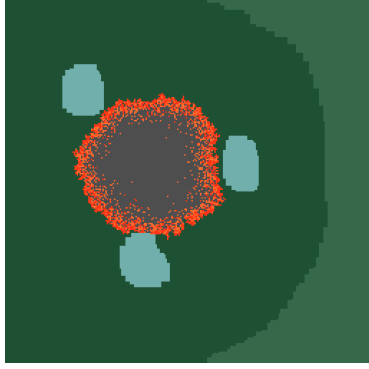


FIGURE 4 – Solution naïve

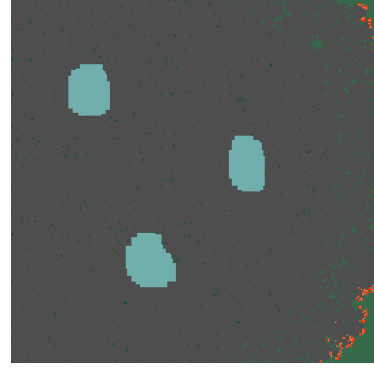


FIGURE 5 – *Alexandridis*

On observe que la solution naïve présente une propagation plus lente et un front de feu plus étendu que le modèle d'*Alexandridis*.

Comparons maintenant deux modélisations du modèle d'*Alexandridis* avec des densités de végétation différentes, sous un vent de  $15m/s$  orienté vers l'est :

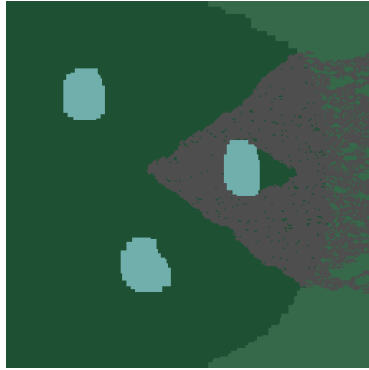


FIGURE 6 – Végétation normale

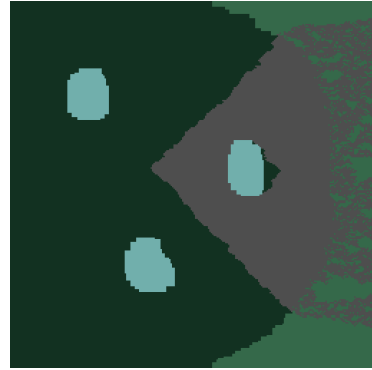


FIGURE 7 – Végétation dense

On remarque principalement deux effets importants sur les deux modélisations.

Premièrement, la progression du feu dans une végétation normale est plus lente (ce qui est cohérent avec l'observation empirique), et le front de feu est donc moins étendu. La surface brûlée dans le cas de la densité normale présente également des zones non brûlées, contrairement à celle de la densité élevée qui est plus uniforme. Ce phénomène s'explique par une probabilité  $p_b$  plus élevée pour une densité plus forte.

Deuxièmement, dans les deux modélisations, l'impact du vent est clairement visible : le front de feu s'est propagé selon un cône poussé par le vent.

## 2.3 Considérations algorithmiques

Cette sous-partie a pour objectif de présenter l'implémentation de la modélisation réalisée et de justifier le choix d'une grille de taille  $256 \times 256$ .

Voici les complexités temporelles des différentes fonctions créées :

- Calcul de  $p_b$  :  $O(1)$
- Calcul du passage de l'état à l'instant  $t$  à l'état à l'instant  $t + 1$  :  $O(n^2)$ , où  $n$  est la dimension de la grille. Cette estimation repose sur environ 100 opérations élémentaires (lectures, comparaisons, affectations, calculs arithmétiques, etc.) pour chaque cellule.
- Affichage de l'état à l'instant  $t$  :  $O(m^2 \times n^2)$ , où  $n$  est la dimension de la grille et  $m$  est la taille en pixels d'une cellule représentée à l'écran.

Ainsi, pour  $N$  itérations, la complexité totale est de  $O(m^2 \times n^2 \times N)$ . Numériquement, avec  $n = 256$ ,  $m = 2$  et  $N = 200$  on trouve  $\approx 10^5$ , et donc on en déduit grossièrement  $\approx 10^7$  opérations élémentaires.

En considérant un temps d'exécution moyen par opération élémentaire, cela correspond grossièrement à un temps de calcul d'environ 5 secondes.

On comprend donc pourquoi augmenter la valeur de  $n$  (toutes choses égales par ailleurs) ou de  $N$  conduirait à des simulations plus longues.

## 3 Transformations

Puisque nous disposons maintenant d'un modèle pour réaliser des simulations, nous allons nous intéresser aux transformations réalisables, notamment ici, dans le cadre de cette étude, aux tranchées et chemins.

### 3.1 Explications

Nous allons donc considérer l'ajout de tranchées ou de chemins forestiers, des transformations légères qui permettent de limiter l'impact des feux sans dénaturer les forêts.

**Proposition 4.** On ajoute un nouvel état Chemins/Tranchées avec  $p_{veg} = -0.55$  et  $p_{den} = 0$  à notre modèle.

## 3.2 Résultats

L'objectif est maintenant d'observer l'impact des tranchées avec différentes configurations de vent et de largeur de tranchées.

### 3.2.1 Tranchées/Sans tranchées

La première comparaison intéressante est d'étudier la différence de propagation du feu avec et sans tranchées.

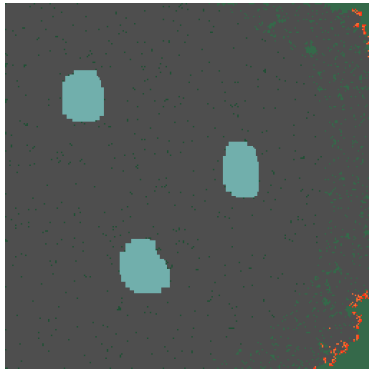


FIGURE 8 – Sans tranchées

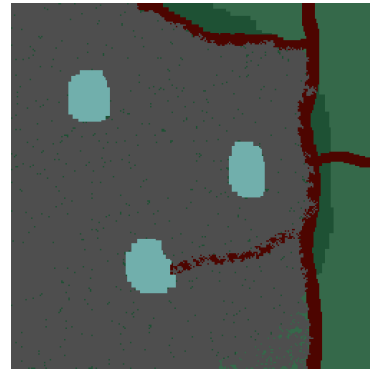


FIGURE 9 – Avec tranchées

En comparant la FIGURE 8 et la FIGURE 9 on constate que la présence de tranchées permet de bloquer la propagation du feu. Il est important de noter que ce résultat est observé pour un nombre d'itérations limité (ici 200), et qu'une simulation plus longue pourrait montrer une propagation au-delà des tranchées.

### 3.2.2 Vent faible/Vent fort

Dans un deuxième temps, comparons deux situations avec un vent orienté dans la même direction mais de vitesse différente.

En comparant la FIGURE 10 et la FIGURE 11 on observe qu'un vent plus fort diminue l'efficacité des tranchées ou chemins. Ce résultat est empiriquement logique : un vent plus intense permet au feu de se propager plus rapidement et donc de franchir plus facilement l'obstacle.

Il est tout de même intéressant de mentionner que, même lorsque le feu parvient à traverser la tranchée, moins de 25% de sa surface (dans la zone impactée par le feu) est brûlée à la fin de la simulation.

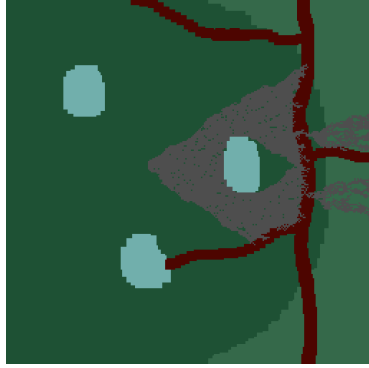


FIGURE 10 – 15 m/s de vent



FIGURE 11 – 30 m/s de vent

### 3.2.3 Tranchée mince/Tranchée large

Dans un dernier temps, comparons deux largeurs de tranchée.

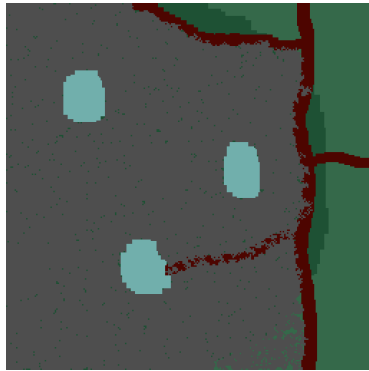


FIGURE 12 – Tranchée de 8m

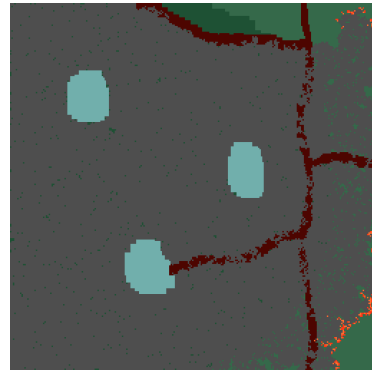


FIGURE 13 – Tranchée de 4m

En comparant la FIGURE 12 et la FIGURE 13 on constate qu'une tranchée plus large est plus efficace pour limiter la propagation du feu. Entre ces deux simulations, la grande tranchée verticale a été réduite horizontalement, et l'impact de ce changement est clairement visible.

De plus, on observe sur la FIGURE 13 que le front de feu est toujours présent, ce qui suggère qu'une simulation plus longue entraînerait la combustion d'une zone encore plus étendue.

## 3.3 Possibles améliorations

Bien que nous ayons obtenu des résultats intéressants avec ce modèle intermédiaire, qui prend en compte le type et la densité de la végétation ainsi que le vent, il est possible d'aller plus loin dans les simulations.

Il serait donc envisageable d'intégrer d'autres paramètres importants tels que l'élévation, l'hu-

midité ou la température.

De plus, on pourrait considérer d'autres transformations pour la gestion des incendies. Par exemple, la réduction de la densité de végétation, dont la FIGURE 7 a permis d'observer l'impact positif sur la limitation de la propagation. Il serait également pertinent d'affiner la modélisation de l'interaction avec les cours d'eau et les lacs, qui sont actuellement considérés comme des obstacles infranchissables, alors qu'ils peuvent influencer la progression du feu de manière plus complexe.

## 4 Annexe

### 4.1 Choix d'implémentation

Pour implémenter les différents modèles, nous avons choisi le langage **C99**. Ce choix a été principalement motivé par deux aspects :

- Le bas niveau du **C** offre un gain significatif en temps d'exécution par rapport à d'autres langages. Ceci est particulièrement avantageux compte tenu du volume important de calculs nécessaires pour simuler l'évolution des états dans nos grilles.
- Le **C**, étant plus largement répandu que l'**OCaml**, bénéficie d'un écosystème de bibliothèques externes plus vaste et plus mature.

Nous avons utilisé les bibliothèques non standard suivantes :

- **SDL2** : Pour la création d'une interface graphique permettant de visualiser l'évolution des simulations en temps réel.
- **cJSON** : Pour la manipulation de fichiers **JSON**, format utilisé pour représenter et importer les configurations initiales des grilles.
- **png** : Pour la création et la manipulation de fichiers **PNG**, afin d'exporter les résultats des simulations sous forme d'images.

Afin de faciliter la création et la configuration des grilles initiales, nous avons développé un site web interactif en **React.JS**. Cette interface web permet de concevoir visuellement les forêts initiales avant de les exporter au format **JSON** pour être utilisées par notre simulateur en **C**.