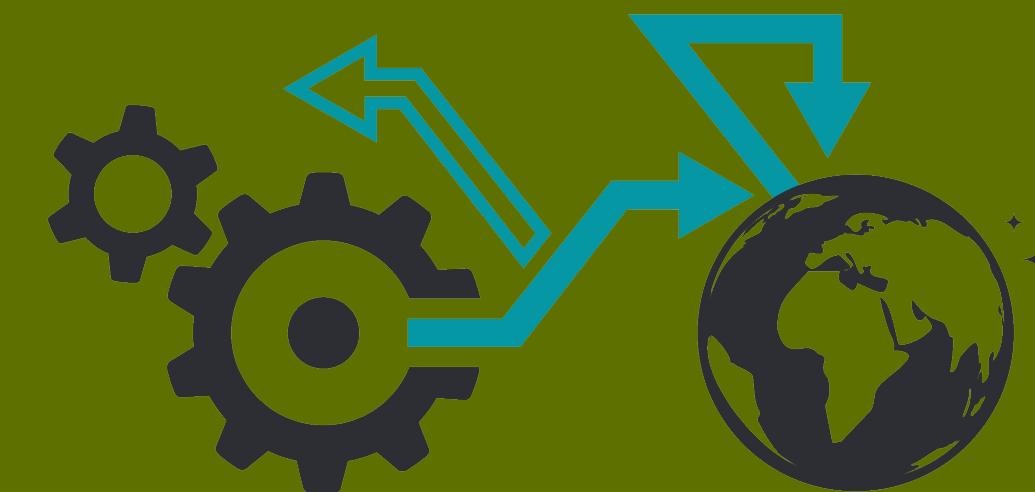


## Projet TOUNDRA

# PRÉSENTATION D'UN TRAVAIL DE VISUALISATION



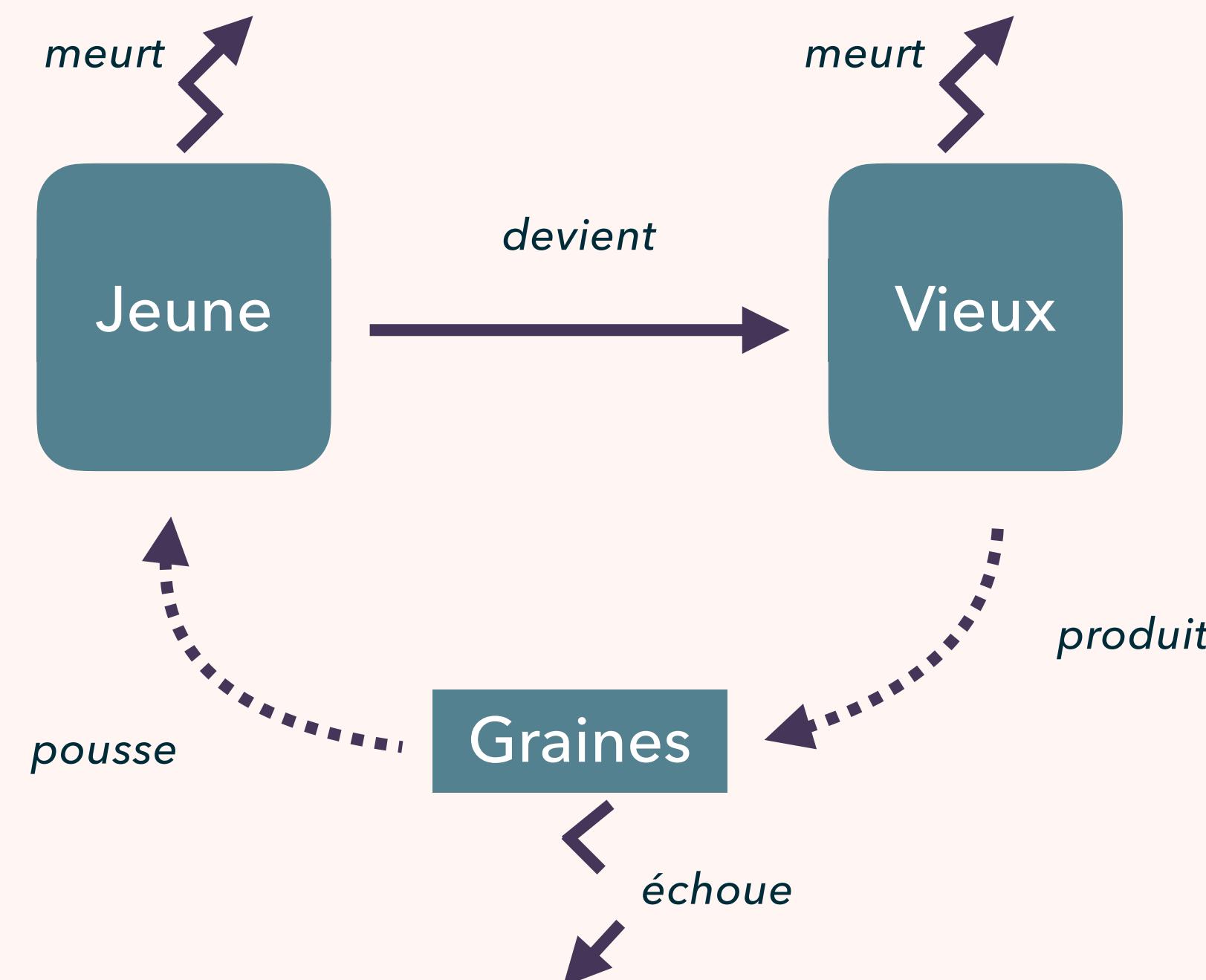
# POINT DE DEPART

- **Modèle d'évolution de forêts avec adjonction de feux (processus hybride)**
- **Premier modèle à mon arrivée : 1 espèce + vieux/jeune**

$$\begin{cases} \frac{\partial u}{\partial t}(t, x) = \beta \delta w(t, x) - \gamma(v(t, x))u(t, x) - fu(t, x), & t > 0, \quad x \in \Omega, \\ \frac{\partial v}{\partial t}(t, x) = fu(t, x) - hv(t, x), & t > 0, \quad x \in \Omega, \\ \frac{\partial w}{\partial t}(t, x) = d\Delta w(t, x) - \beta w(t, x) + \alpha v(t, x), & t > 0, \quad x \in \Omega. \end{cases}$$

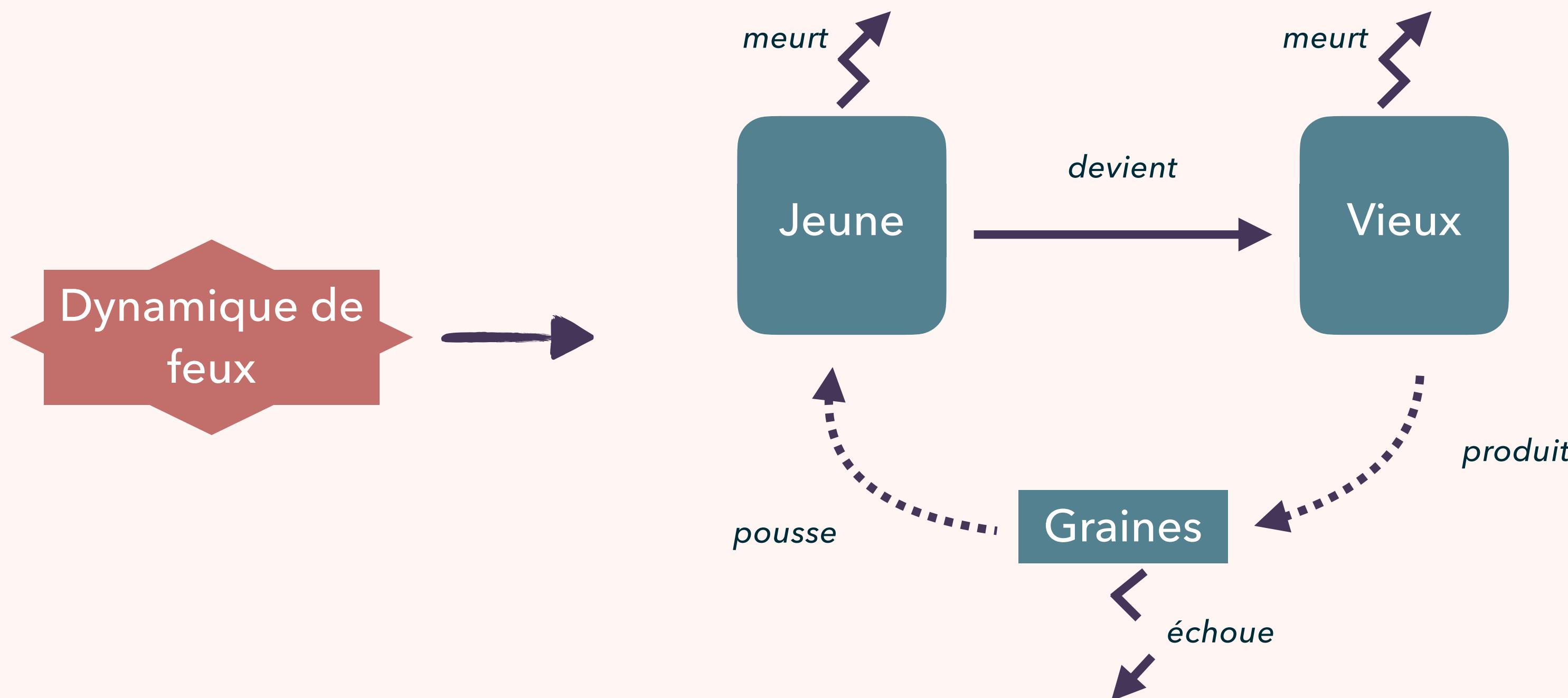
# POINT DE DÉPART

- Modèle d'évolution de forêts avec adjonction de feux (processus hybride)
- Premier modèle à mon arrivée : 1 espèce + vieux/jeune



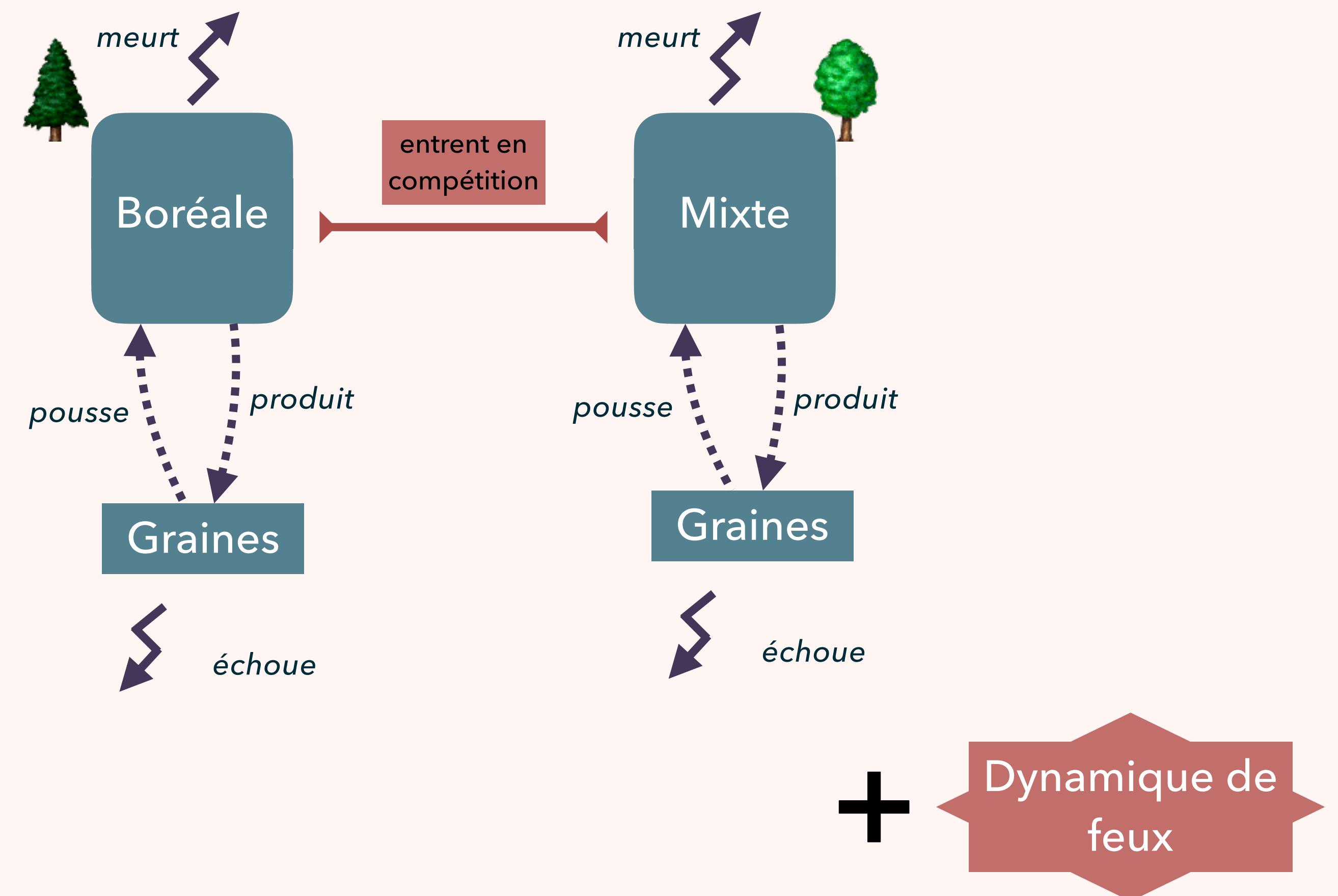
# POINT DE DÉPART

- Modèle d'évolution de forêts avec adjonction de feux (processus hybride)
- Premier modèle à mon arrivée : 1 espèce + vieux/jeune

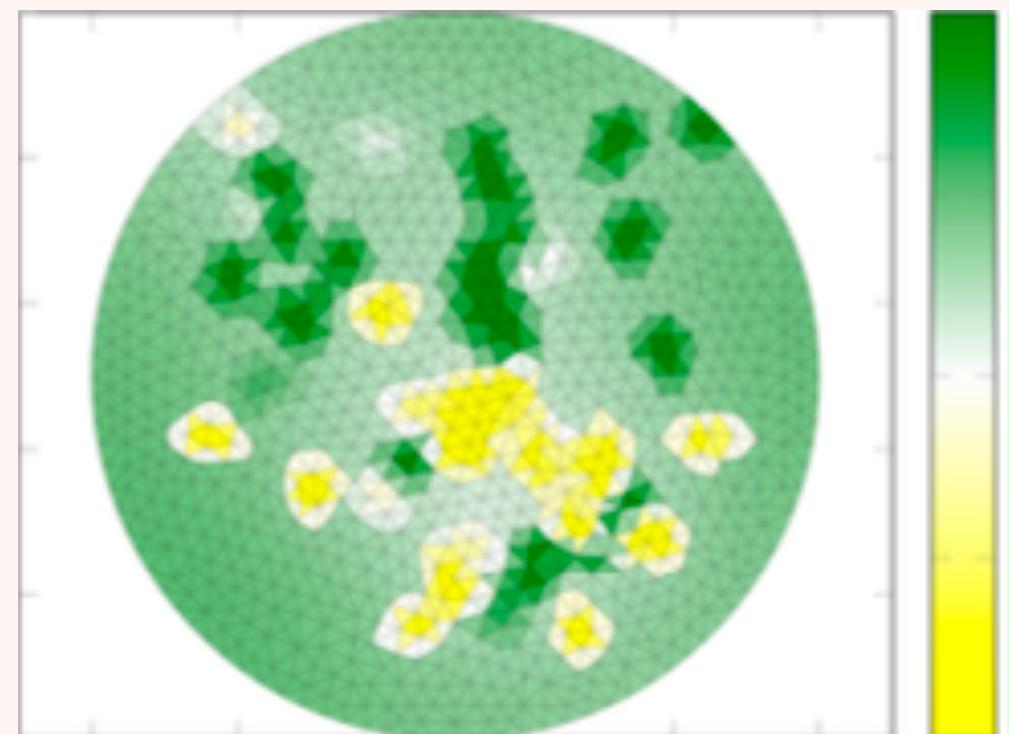
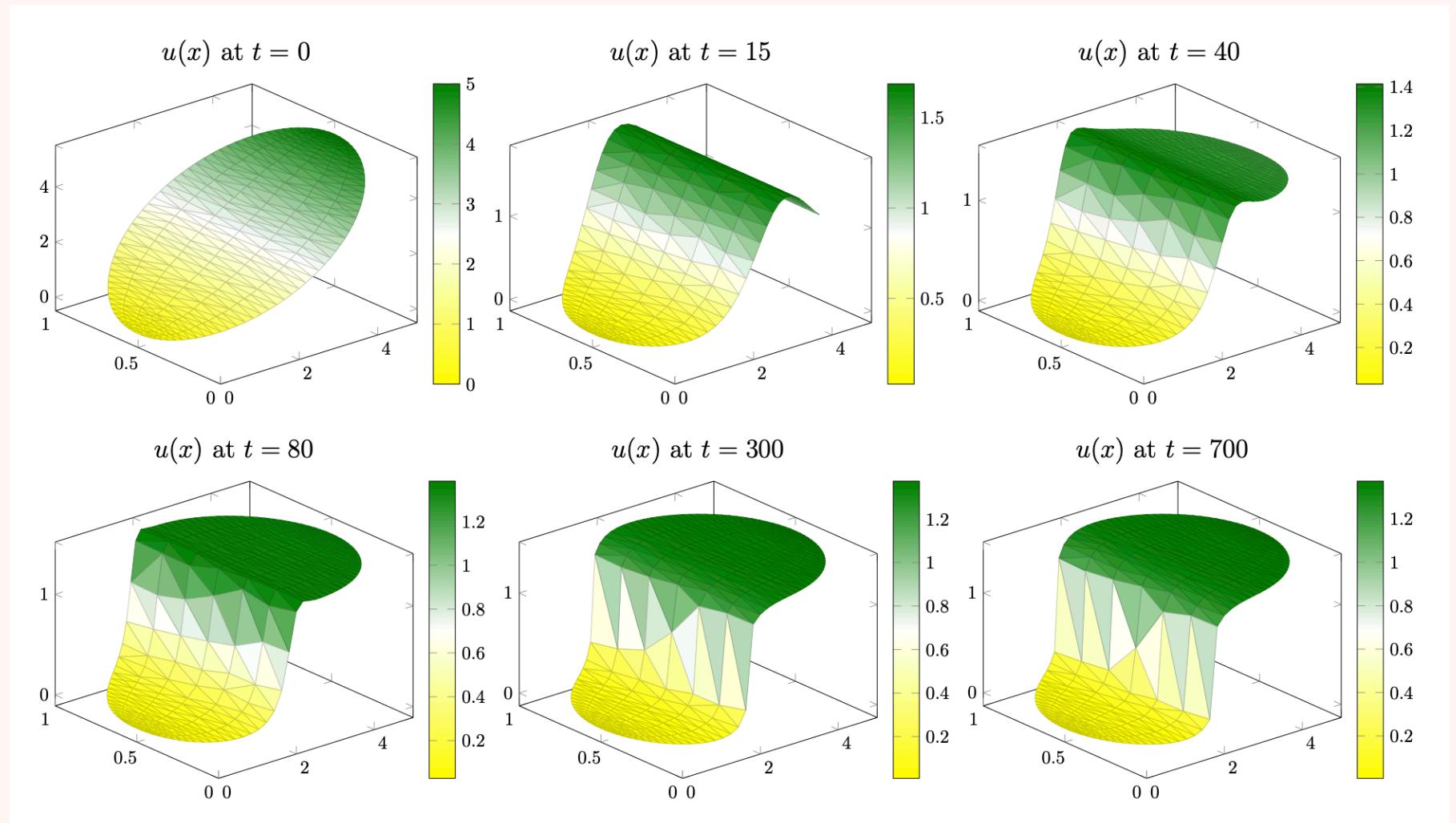


# D'AUTRES MODÈLES

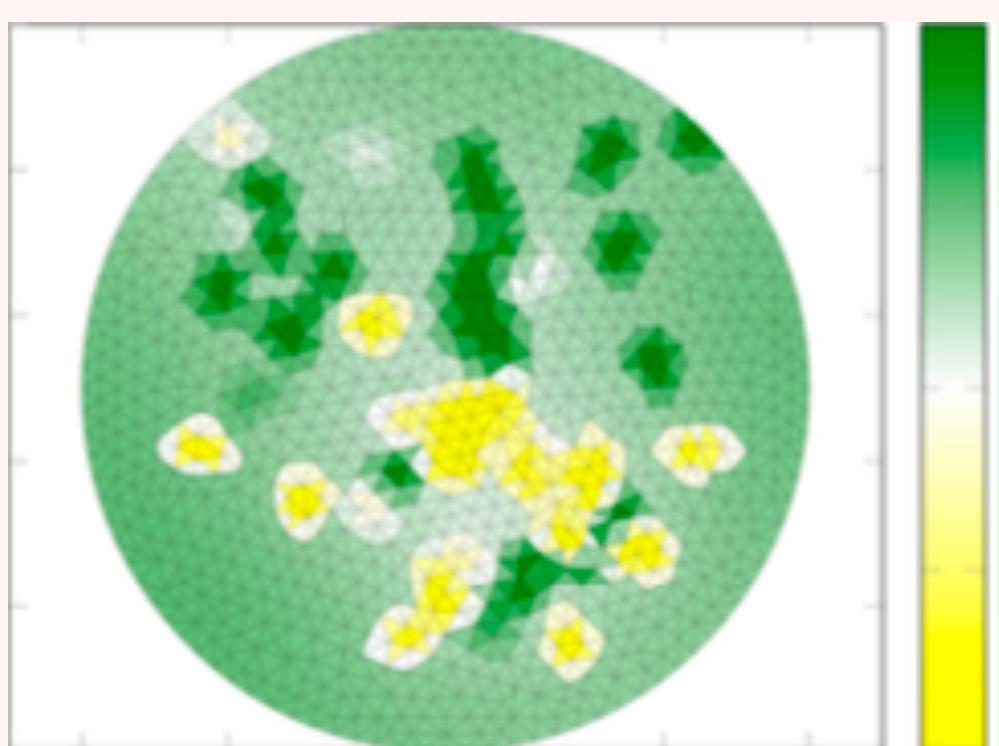
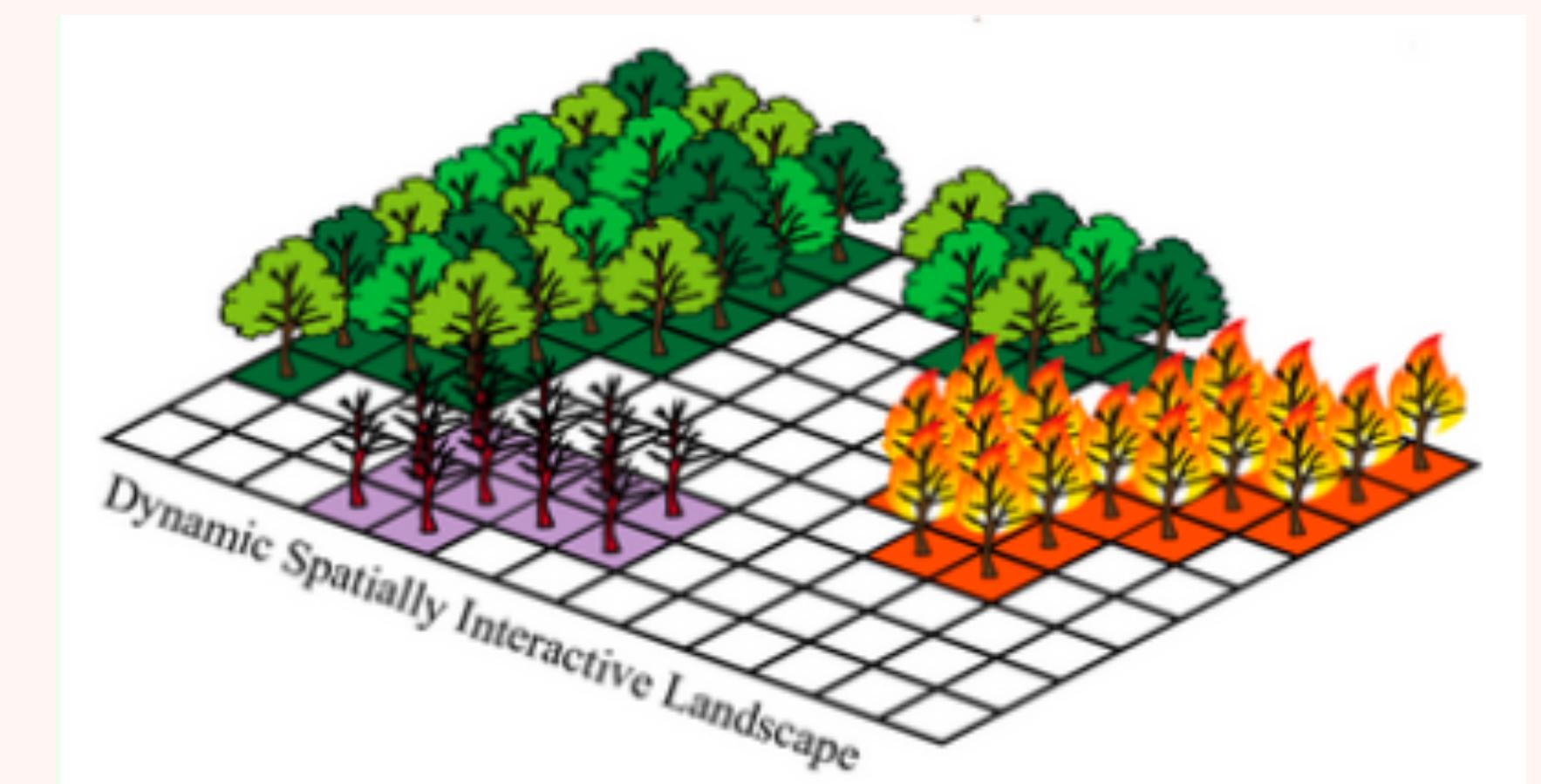
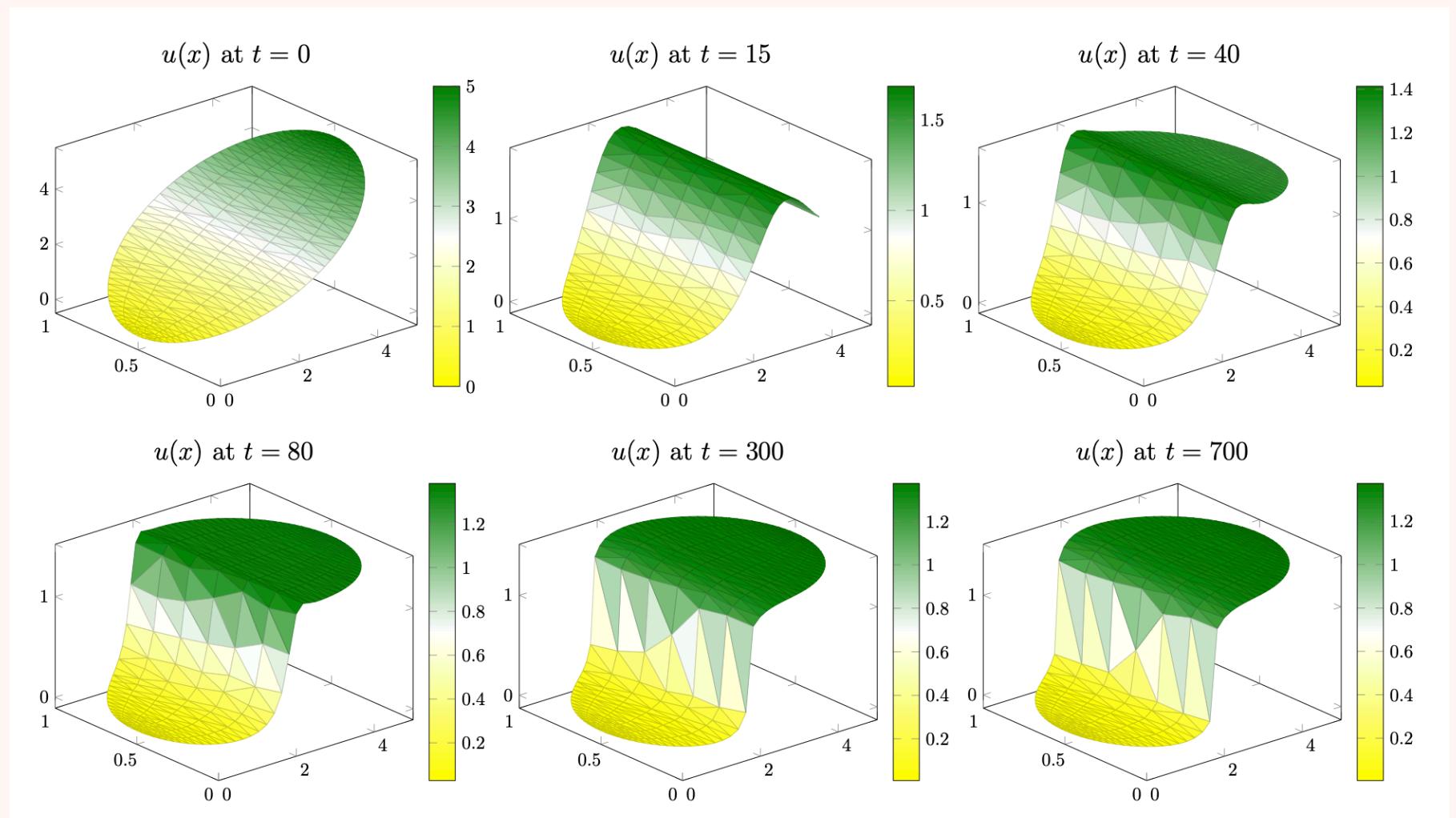
- **Deux équations** : 1 espèce d'arbre, pas d'âge
- **Quatre équations** : 2 espèces d'arbres, pas d'âge



# PRÉSENTER LES RÉSULTATS

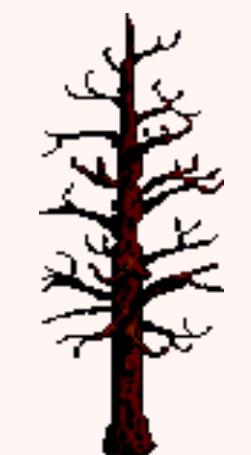
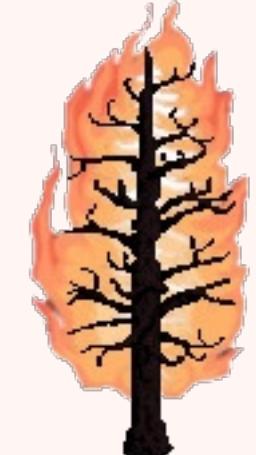
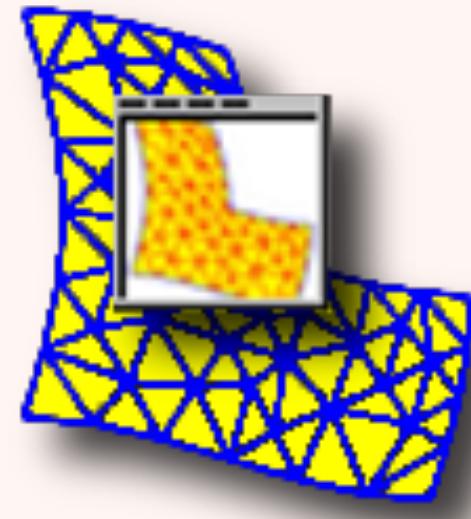


# PRÉSENTER LES RÉSULTATS



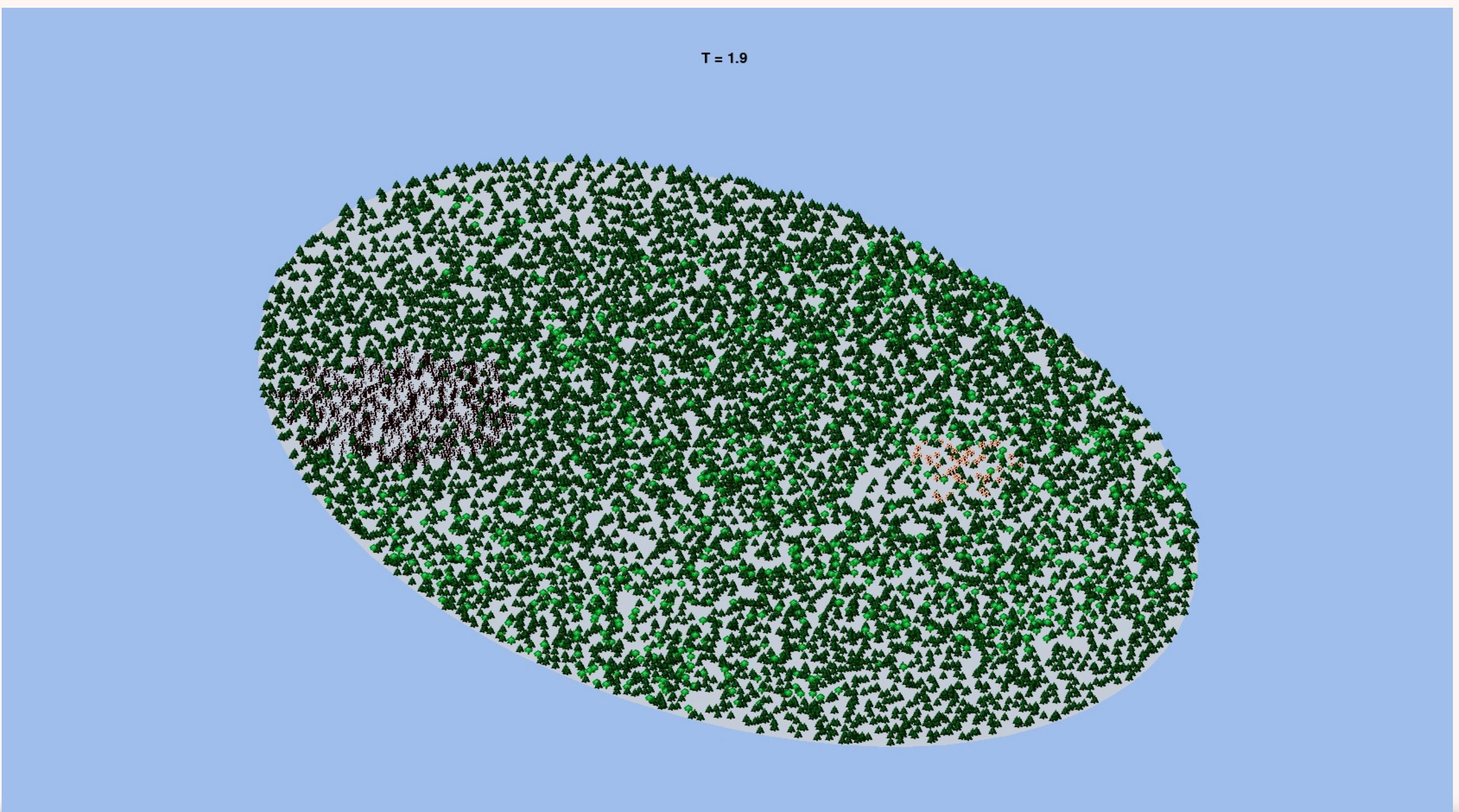
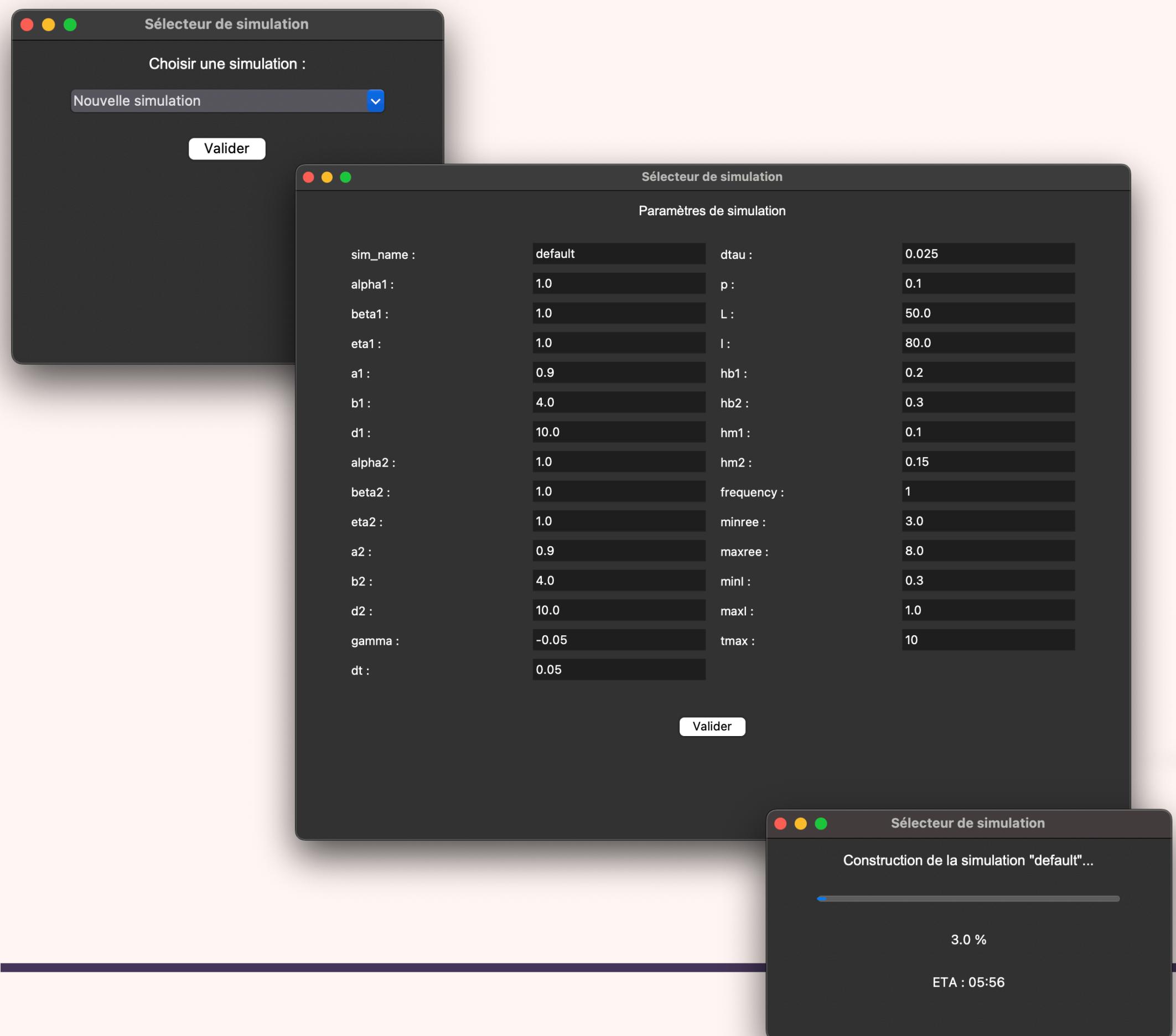
# PRÉSENTER LES RÉSULTATS

- **Simulations avec FreeFem ou Différences Finies**
- **Rendu avec PyGame**
- **Trois « applications » : FreeFem 4 équations, DF 2 équations, DF 4 équations**



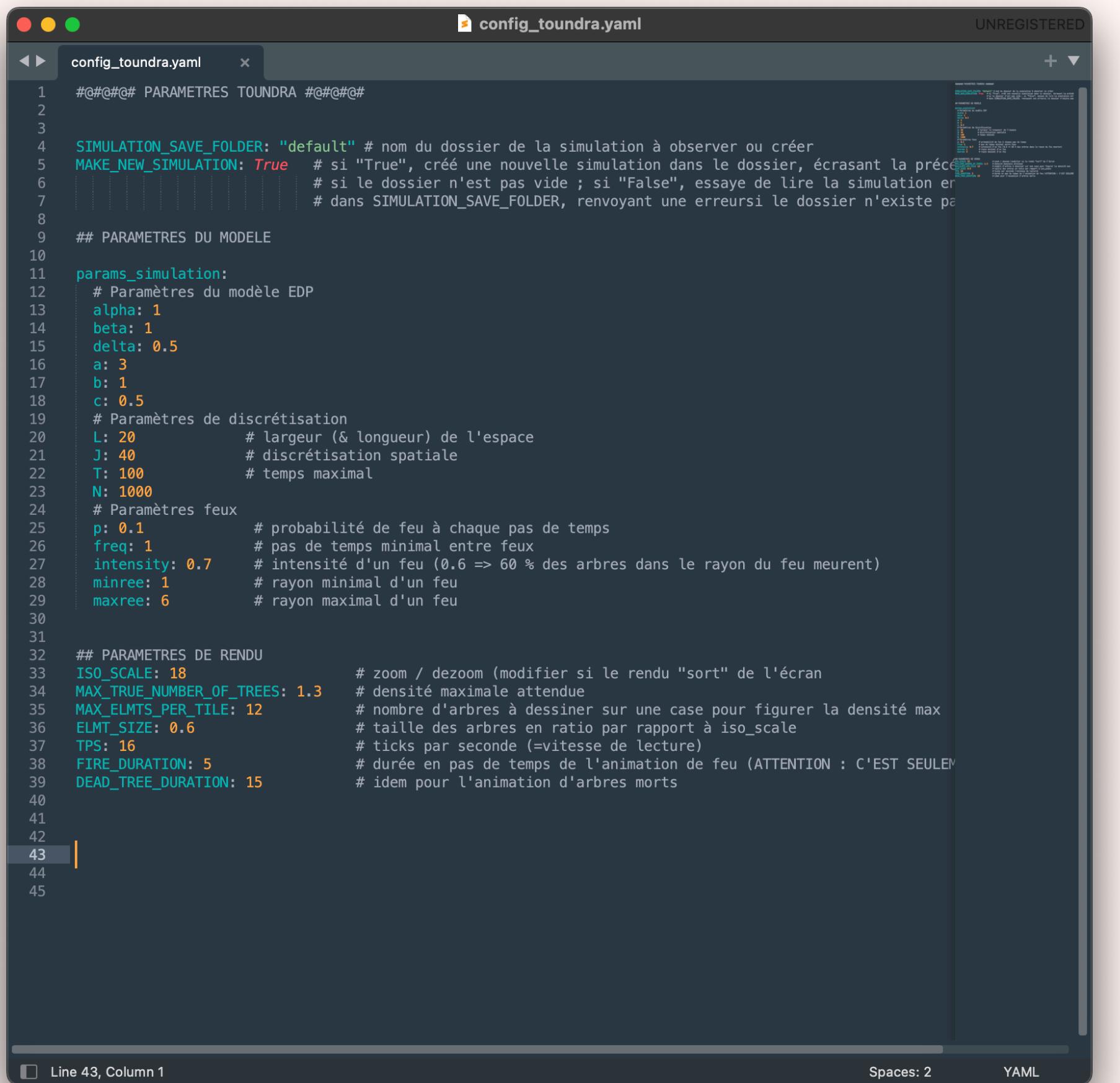
# TOUNDRA2

➤ FreeFem++ en subprocess  
et rendu pygame, menu à  
interface TK

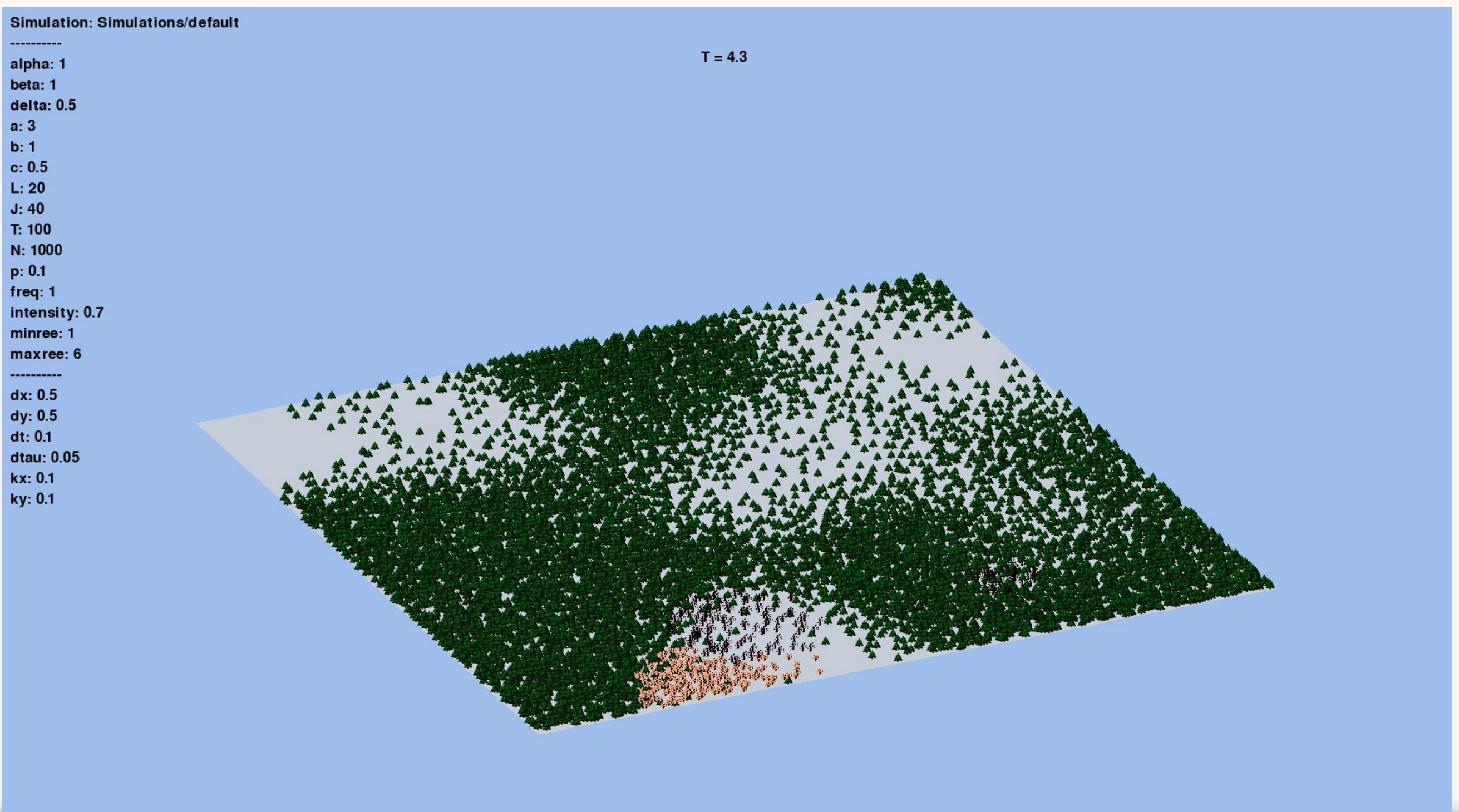


# TOUNDRA3

➤ modèles 2 équations,  
DF, config par .yaml

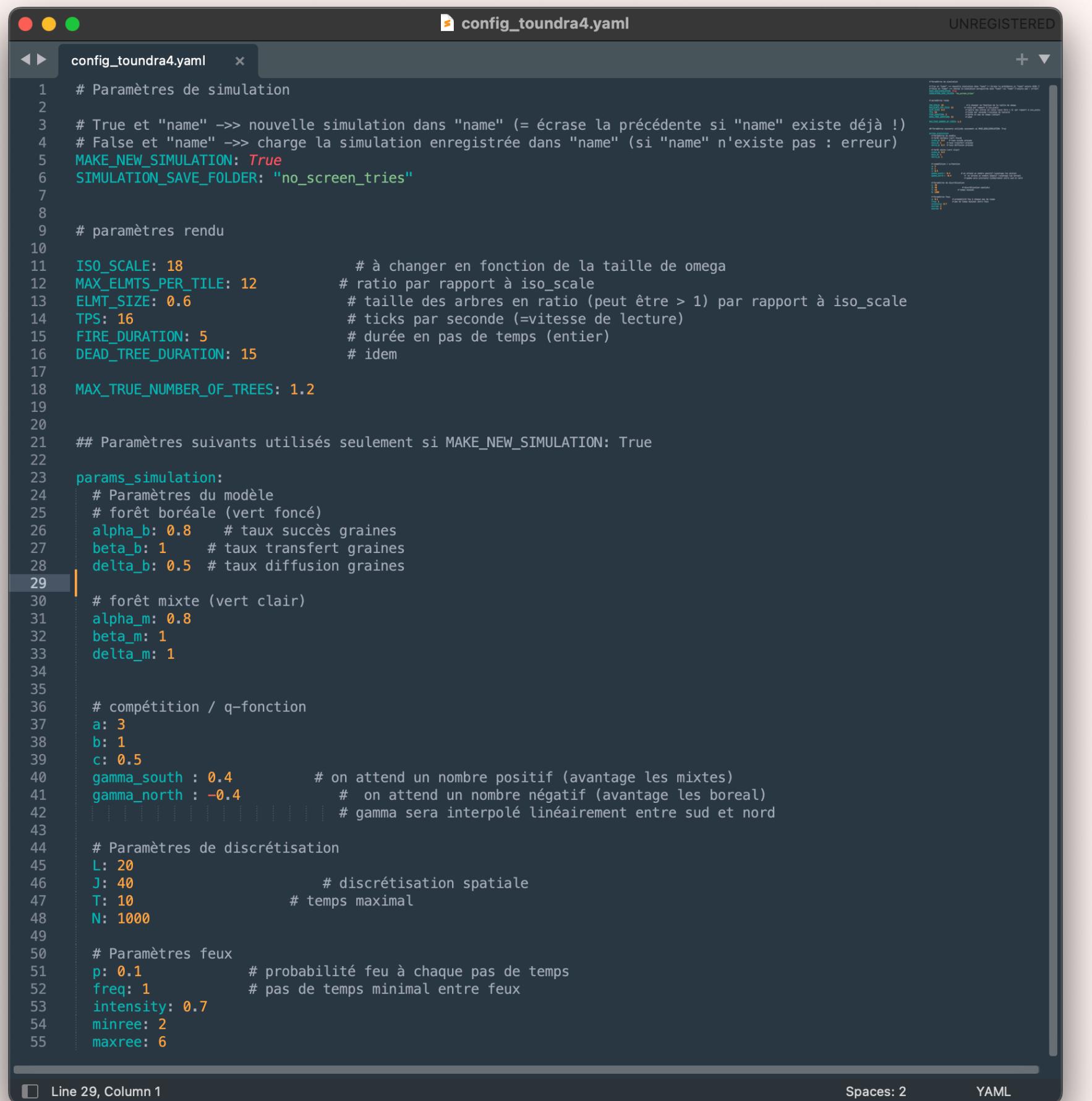


```
config_tundra.yaml
PARAMETRES TUNDRA
SIMULATION_SAVE_FOLDER: "default" # nom du dossier de la simulation à observer ou créer
MAKE_NEW_SIMULATION: True # si "True", crée une nouvelle simulation dans le dossier, écrasant la précédente si le dossier n'est pas vide ; si "False", essaie de lire la simulation existante dans SIMULATION_SAVE_FOLDER, renvoyant une erreur si le dossier n'existe pas
## PARAMETRES DU MODELE
params_simulation:
  # Paramètres du modèle EDP
  alpha: 1
  beta: 1
  delta: 0.5
  a: 3
  b: 1
  c: 0.5
  # Paramètres de discrétisation
  L: 20 # largeur (& longueur) de l'espace
  J: 40 # discrétisation spatiale
  T: 100 # temps maximal
  N: 1000
  # Paramètres feux
  p: 0.1 # probabilité de feu à chaque pas de temps
  freq: 1 # pas de temps minimal entre feux
  intensity: 0.7 # intensité d'un feu (0.6 => 60 % des arbres dans le rayon du feu meurent)
  minree: 1 # rayon minimal d'un feu
  maxree: 6 # rayon maximal d'un feu
## PARAMETRES DE RENDU
ISO_SCALE: 18 # zoom / dezoom (modifier si le rendu "sort" de l'écran
MAX_TRUE_NUMBER_OF_TREES: 1.3 # densité maximale attendue
MAX_ELMTS_PER_TILE: 12 # nombre d'arbres à dessiner sur une case pour figurer la densité max
ELMT_SIZE: 0.6 # taille des arbres en ratio par rapport à iso_scale
TPS: 16 # ticks par seconde (=vitesse de lecture)
FIRE_DURATION: 5 # durée en pas de temps de l'animation de feu (ATTENTION : C'EST SEULEMENT POUR LES FEUX EN COURS)
DEAD_TREE_DURATION: 15 # idem pour l'animation d'arbres morts
```



# TOUNDRA4

➤ modèle 4 équations, DF,  
config par .yaml



```
config_toundra4.yaml
UNREGISTERED

# Paramètres de simulation
# True et "name" -> nouvelle simulation dans "name" (= écrase la précédente si "name" existe déjà !)
# False et "name" -> charge la simulation enregistrée dans "name" (si "name" n'existe pas : erreur)
MAKE_NEW_SIMULATION: True
SIMULATION_SAVE_FOLDER: "no_screen_tries"

# paramètres rendu
ISO_SCALE: 18          # à changer en fonction de la taille de omega
MAX_ELMTS_PER_TILE: 12 # ratio par rapport à iso_scale
ELMT_SIZE: 0.6          # taille des arbres en ratio (peut être > 1) par rapport à iso_scale
TPS: 16                 # ticks par seconde (=vitesse de lecture)
FIRE_DURATION: 5         # durée en pas de temps (entier)
DEAD_TREE_DURATION: 15   # idem

MAX_TRUE_NUMBER_OF_TREES: 1.2

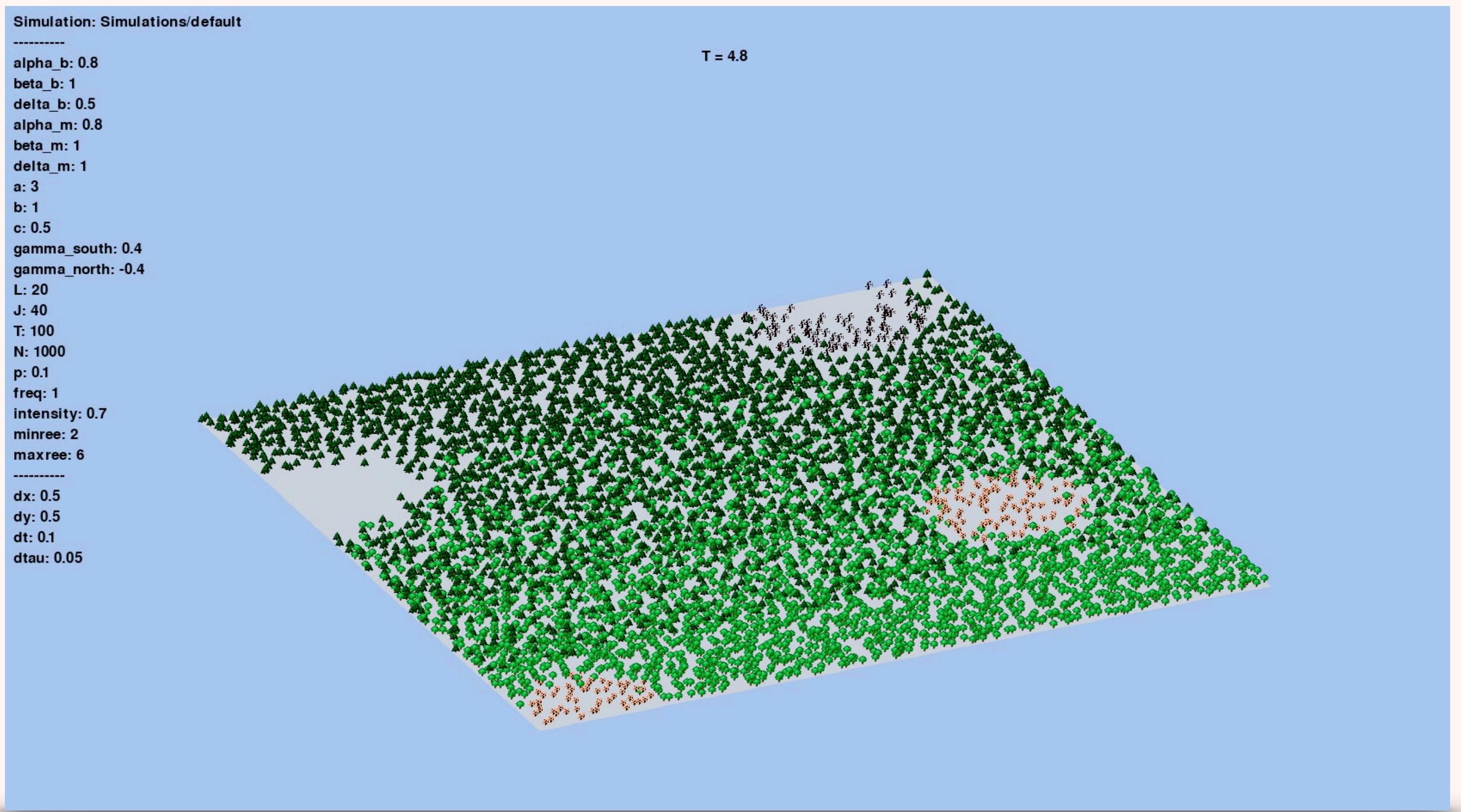
## Paramètres suivants utilisés seulement si MAKE_NEW_SIMULATION: True
params_simulation:
# Paramètres du modèle
# forêt boréale (vert foncé)
alpha_b: 0.8          # taux succès graines
beta_b: 1              # taux transfert graines
delta_b: 0.5            # taux diffusion graines
# forêt mixte (vert clair)
alpha_m: 0.8
beta_m: 1
delta_m: 1

# compétition / q-fonction
a: 3
b: 1
c: 0.5
gamma_south : 0.4      # on attend un nombre positif (avantage les mixtes)
gamma_north : -0.4     # on attend un nombre négatif (avantage les boreal)
                      # gamma sera interpolé linéairement entre sud et nord

# Paramètres de discrétisation
L: 20                  # discrétisation spatiale
J: 40                  # temps maximal
T: 10
N: 1000

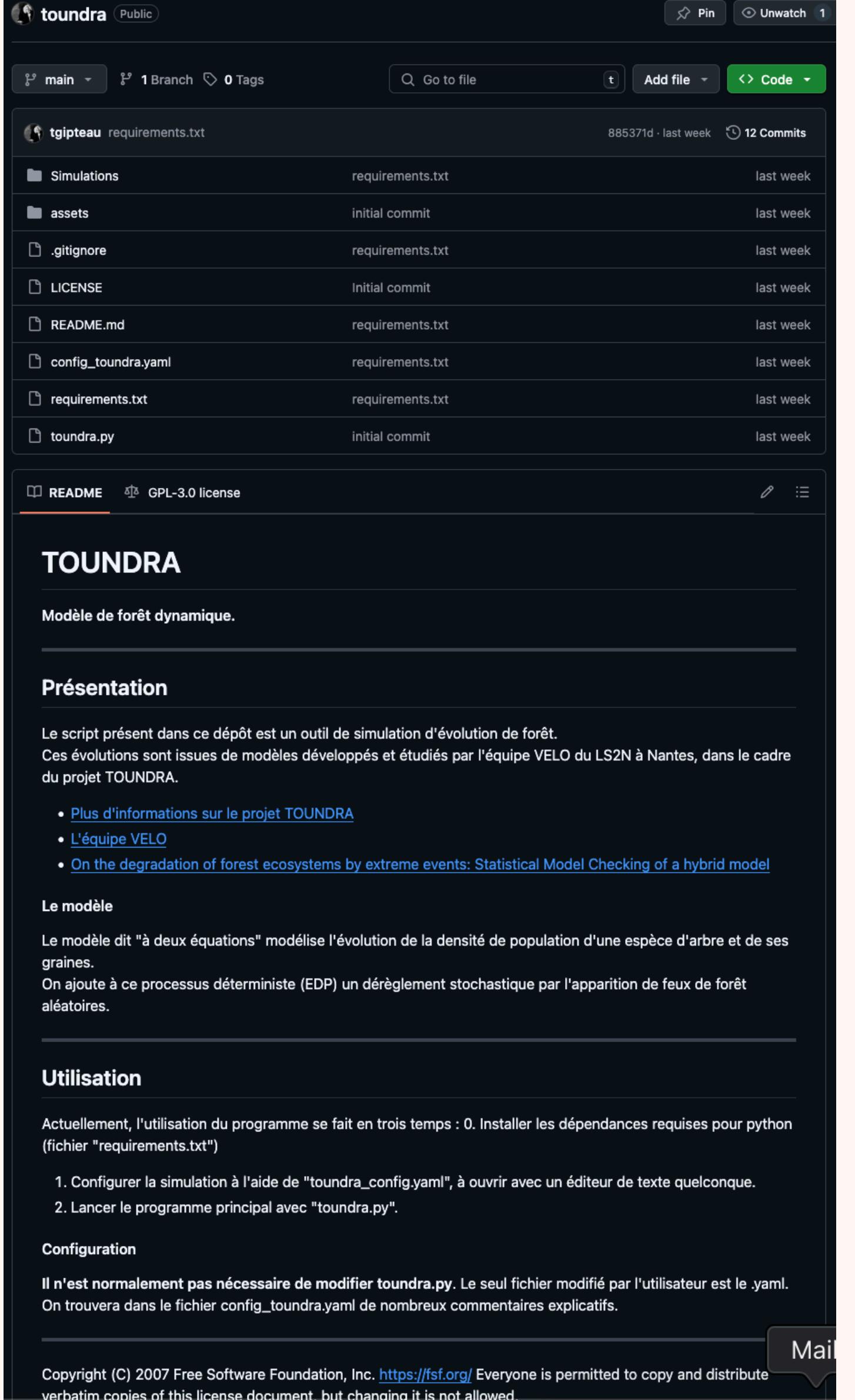
# Paramètres feux
p: 0.1                # probabilité feu à chaque pas de temps
freq: 1                # pas de temps minimal entre feux
intensity: 0.7
minree: 2
maxree: 6

Line 29, Column 1
Spaces: 2
YAML
```



# LA SUITE

- Pour la visu : Distribuer les applications ? (site pythonanywhere, executables standalone...)
- Pour moi : travailler sur une partie plus théorique (modélisation par chaînes de Markov)



The screenshot shows a GitHub repository named 'toundra' with 1 branch and 0 tags. The repository was last updated 885371d · last week with 12 commits. The file structure includes 'main', 'Simulations', 'assets', '.gitignore', 'LICENSE', 'README.md', 'config\_toundra.yaml', 'requirements.txt', and 'toundra.py'. The 'README.md' file contains the following content:

```
TOUNDRAGit repository for the TOUNDRAGit repository for the TOUNDRA project. The project is a Python script for simulating forest dynamics. The script is licensed under the GPL-3.0 license. The project is maintained by the VELO team at LS2N in Nantes. The README file provides information on the project's purpose, the underlying model, and how to use the software. It also links to the project's website and the team's GitHub profile. The configuration file 'config_toundra.yaml' is also mentioned as being modified by users.
```

**TOUNDRAGit repository for the TOUNDRAGit repository for the TOUNDRA project. The project is a Python script for simulating forest dynamics. The script is licensed under the GPL-3.0 license. The project is maintained by the VELO team at LS2N in Nantes. The README file provides information on the project's purpose, the underlying model, and how to use the software. It also links to the project's website and the team's GitHub profile. The configuration file 'config\_toundra.yaml' is also mentioned as being modified by users.**

**Présentation**  
Le script présent dans ce dépôt est un outil de simulation d'évolution de forêt. Ces évolutions sont issues de modèles développés et étudiés par l'équipe VELO du LS2N à Nantes, dans le cadre du projet TOUNDRA.

- [Plus d'informations sur le projet TOUNDRA](#)
- [L'équipe VELO](#)
- [On the degradation of forest ecosystems by extreme events: Statistical Model Checking of a hybrid model](#)

**Le modèle**  
Le modèle dit "à deux équations" modélise l'évolution de la densité de population d'une espèce d'arbre et de ses graines. On ajoute à ce processus déterministe (EDP) un dérèglement stochastique par l'apparition de feux de forêt aléatoires.

**Utilisation**  
Actuellement, l'utilisation du programme se fait en trois temps : 0. Installer les dépendances requises pour python (fichier "requirements.txt")

1. Configurer la simulation à l'aide de "toundra\_config.yaml", à ouvrir avec un éditeur de texte quelconque.
2. Lancer le programme principal avec "toundra.py".

**Configuration**  
Il n'est normalement pas nécessaire de modifier toundra.py. Le seul fichier modifié par l'utilisateur est le .yaml. On trouvera dans le fichier config\_toundra.yaml de nombreux commentaires explicatifs.

Copyright (C) 2007 Free Software Foundation, Inc. <https://fsf.org/> Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.