

431 avenue Gaston Berger

13625 Aix-En-Provence

Téléphone 04 42 93 90 43

Fax 04 42 93 90 74

[iut-aix-informatique@univ-amu.fr](mailto:iut-aix-informatique@univ-amu.fr)

www.iut.univ-amu.fr

Modélisation Mathématique

Projet automate cellulaire

Dossier final

Aurélien ROBINEAU, Damien PIEDANNA,

Nicolas MEYNIEL, Axel PISANI

Groupe 3A&B – 2eme année

Enseignent responsable : Patricia GAITAN

Module M3202C – Modélisation mathématique

Mercredi 7 Novembre 2018

Table des matières

[Présentation du projet 3](#_Toc527018504)

[Le code 4](#_Toc527018505)

[Génération de la matrice et des cases 4](#_Toc527018506)

[Outil et source 5](#_Toc527018507)

## Présentation du projet

Nous avons choisi de modéliser, dans le langage de programmation Python, un automate cellulaire, une matrice, où chacune de ces cases représentent un groupe de personnes. La globalité de la grille représente une zone géographique généré de manière randome et se faisant infecter par plusieurs virus au coure du temps, c’est-à-dire, jour par jour.

Il y a trois types de case :

* les cases saines, donc non infecter par un virus (verte)
* les cases infecter par un virus (rouge)
* les cases dépourvues de population (blanche)

Lors de la compilation du programme, le terminal Linux s’ouvre pour afficher, la génération de la matrice.

On clique sur une case saine pour initialiser la grille avec un départ d’infection. Le virus commence par infecter la case courante, au jour 0, avant de progresser vers les cases qui lui sont voisines. Les cases voisines étant les 8 cases autour de la case courante.

Cependant, on ne peut pas infecter une case dépourvue de population (blanche).

Chaque case voisine de la case courante possède ça propre probabilité d’être infectée par la case courante au jour 1.

Le but de cette automate cellulaire est de rendre toute cette zone infectée par les virus, avec le moins de survivant possible.

Pour ce faire nous avons pris en compte :

* l’âge moyen de chaque cellule, constituée d’une population de tout âge (0 à 105 ans)
* le temps en jours passé proche d’une cellule infectés
* le taux d’infection d’un virus

## Le code

Tout le programme a été codé de manière orienté objet.

### Génération de la matrice et des cases

Pour générer la matrice nous avons créé une classe Grille qui contient :

* une méthode def \_\_init\_\_ qui va :
  + Initialiser la matrice à une Hauteur et une Largeur respectivement égale à nbCelluleHauteur et nbCellouleLargeur qui sont 2 des paramètres de cette méthode.
  + Générer la taille de chaque Case en fonction de la hauteurPx (hauteur en pixel). C’est-à-dire qu’au plus il y aura de cellule au plus elles seront petites avec la même taille et inversement proportionnel.

## Outil et source

Pour réaliser notre projet d’automate cellulaire, nous avons constamment travaillé en équipe.

Nous avons utilisé les logiciels :

* CoCalc : est open source hébergé par SageMath inc
* Sublime Text : est open source qui est exécutable sous le bash Linux et prend en compte le langage Python (tout comme 44 langages au totale)
* Microsoft Word : pour toute la rédaction du présent dossier