 413 avenue Gaston Berger

13625 Aix-en-Provence

Tél. 04 42 93 43

Télécopie 04 42 93 74

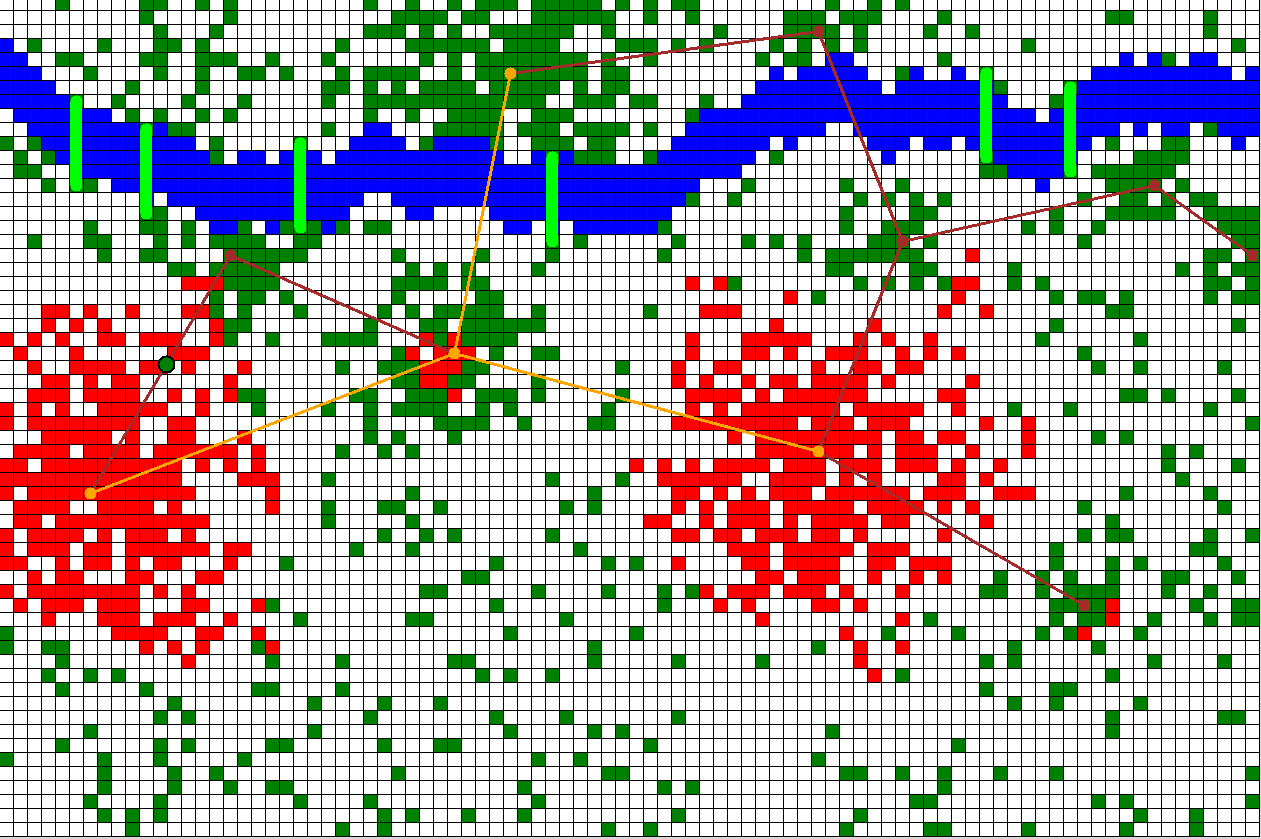
[Sylvie.seropian@univ-amu.fr](mailto:Sylvie.seropian@univ-amu.fr)

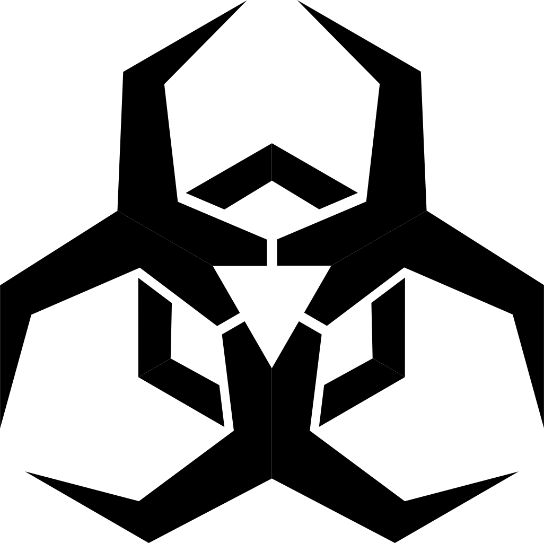
https://iut.univ-amu.fr/

Module 3202C

**Modélisation mathématique**

Automates cellulaires





Nicolas MEYNIEL, Damien PIEDNNA, Axel PISANI, Aurélien ROBINEAU

Groupe 3B – 2nd année

Module M3202C – Modélisation mathématique

Enseignant responsable : Patricia GAITAN

Mercredi 7 Novembre 2018

Table des matières

**Choix du sujet3**

**Les automates cellulaires 3**

**Evolution du projet4**

**Etat final5**

**Tapez le titre du chapitre (niveau 2)5**

**Tapez le titre du chapitre (niveau 3)6**

**Choix du sujet**

Lorsque les premiers sujets nous ont été proposé nous étions partis sur la réalisation de formes géométriques en 3D et en mouvement mais cela nous paraissait trop classique, nous voulions réaliser une représentation d’un fait réel tel qu’un feu ou des mouvements de populations. Là nous est venu l’idée d’utiliser les automates cellulaires et de représenter la propagation d’un virus dans une population avec des facteurs définis.

**Les automates cellulaires**

Les automates cellulaires sont utilisés pour modéliser des systèmes complexes. Ils consistent en une grille régulière de « cellules » contenant chacune un « état », choisi parmi un ensemble fini et qui peut évoluer au cours du temps. L'état d'une cellule au temps t+1 est fonction de l'état au temps t d'un nombre fini de cellules appelé son « voisinage ». À chaque nouvelle unité de temps, les mêmes règles sont appliquées simultanément à toutes les cellules de la grille, produisant une nouvelle « génération » de cellules dépendant entièrement de la génération précédente. Les exemples sont nombreux : trafic routier, réseaux informatiques, formation d’essaims d’oiseaux, occupation de nids d’abeille, tissus cellulaires…

Du côté informatique, les automates cellulaires peuvent également être vus comme un modèle abstrait représentant le fonctionnement des ordinateurs (précisément, ils constituent un modèle de calcul ayant la puissance des machines de Turing).

**Evolution du projet**

Au début, nous n’avions qu’une simple grille peu fonctionnelle avec des cellules reparties aléatoirement et de couleurs différentes représentant des populations saines(vertes) ou infecte(rouge).

Ensuite, diverses fonctionnalités ont été ajoute tel que pouvoir choisir quelles cellules de la grille allaient être infecte par le virus ou non grâce au clics souris, et pouvoir choisir le nombre de cellules de la grille en hauteur et en largeur.

Par la suite nous avons implémenté la fonction qui permet de propager le virus à travers la population à partir d’un point donné. Ce point de départ détecte les cellules de populations autour de lui et avec des probabilités définies pas nous-même, déterminait si le virus allait se propager ou non.

Le projet évoluant nous avons rajoute des zones peuplées, des villages, villes et métropoles représentés graphiquement par des zones ou le nombre de cellules de populations y est plus dense. Au centre d’une ville la population sera donc plus dense qu’à l’extérieur et cela influencera la propagation du virus. Des fleuves ont aussi été rajoute avec des ponts qui permettent le passage d’un côté à l’autre grâce à des ponts.

La fonction de propagation du virus a été amélioré avec une fonction qui s’appelle l’incidence proportionnelle, elle comprend deux composantes :

- Un décompte du nombre de contacts ayant lieu à l’instant T,

- Et une description de la probabilité qu’un tel contact, lorsqu’il a lieu, se traduise par la transmission de la maladie

Les cellules de populations ont été perfectionnés en y ajoutant un âge moyen variant de 0 à 105 ans qui permettra d’interagir avec la fonction de propagation afin de lui donner plus de réalisme.

Pour rendre notre simulation encore plus réaliste nous avons ajouté des déplacements tels que les routes, les trains ou encore l’avion(qui n’apparait qu’entre deux métropoles) qui permettent aux populations de se déplacer entres différentes zones urbaines et ainsi d’y répandre le virus, mais ce n’est pas tout, nous avons aussi ajouté un menu permettant à l’utilisateur de définir la taille souhaité de la simulation et de choisir le virus qui se propagera sachant que le virus interagi différemment avec les différentes classes d’âge des populations .

**Etat final**

Au final, la simulation comporte tous les éléments suivant :

-Différents types de cellules:

-les cellules saines (verte)

-les cellules infectées (rouge)

-les cellules vide (blanche)

-les cellules guéries (verte claire)

-les cellules mortes (grises)

-les cellules d’eau (bleu)

-Différentes zones urbaines créées aléatoirement:

-des zones peuplées

-des villages

-des villes

-des métropoles

-Des moyens de déplacements :

-des routes (marron) entre toutes les zones urbaines sauf les zones peuplées

-des lignes de train (orange) entre les villes et la métropole

-des voies aériennes (rose) entre deux métropoles

-des ponts pour piétons (vert fluo)

-Un menu au lancement de la simulation permettant de choisir la taille en hauteur et largeur de la simulation ainsi que le virus qui se propageras (sachant que certains sont plus ou moins létal).

-Les trois clics souris on chacun une action dédiée :

-le clic gauche infecte la cellule cible

- le clic droit transforme la cellule cible en cellule saine si elle est infectée

-le clic molette permet d’avancer d’une étape lorsque la simulation est en pause

-Une fonction de propagation qui permet au virus (précédemment choisis par l’utilisateur) de se répandre. Pour déterminer si une cellule va en infecter une autre on utilise l’incidence proportionnelle qui à partir du nombre de cellules saines proches, du nombre d’infecté proche et des propriétés du virus détermine si au prochain tour des cellules seront infecte ou non. Ensuite il y a une probabilité propre à chaque virus d’en mourir ou d’en guérir au bout d’un certain temps.

**Les maths dans le code**

**Algorithme de Prim**

L'algorithme de Prim est un algorithme dit glouton, qui calcule un arbre couvrant minimal d’un graphe connexe valué et non orienté. En d'autres termes, cet algorithme trouve un sous-ensemble d'arêtes formant un arbre sur l'ensemble des sommets du graphe initial, tel que la somme des poids de ces arêtes soit minimale.

***AURELIEN IL FAUT FAIRE DES TRUCS ICI***

Dans notre projet, chaque sommet est représenté par le centre des villages, villes et métropoles. Afin de créer des liaisons, ici représentés pour les axes routiers, ferroviaires et aériens, nous avons implémenté l’algorithme de Prim et utilisé pour chaque axe. Le poids minimal est la distance entre chaque ville.

**Probabilités**

**Matrices**

**Incidence proportionnelle**

En épidémiologie, l'incidence décrit, avec la prévalence, l'importance d'une maladie dans une population et signifie chaque nouveau cas d'une maladie par unité de temps dans une population définie.

Dans le projet nous utilisons la formule de l’incidence proportionnelle qui nous permet que calculer la probabilité qu’une cellule soit infecte par une cellule infecte voisine en fonction du nombre d’infecte proche, du nombre de sains proche, du coefficient r0 du virus (qui correspond au taux de reproduction de base d'une maladie infectieuse) et de la moyenne d’âge de la cellule

**Densité des zones urbaines**

La population de la zone urbaine (hors Zone Peuplée) doit être de moins en moins dense plus l'on approche du bord de la zone. On veut qu'au centre, la population soit très dense. Il aura donc 100% de chance que la cellule au centre de la zone soit peuplée. On veut qu'en bordure de zone une cellule ai 25% de chance d'être peuplée. Si on représente cela par une fonction linéaire on a :

f(x) = ax+b (pourcentage de chance d'être peuplée en fonction de la distance de la cellule au centre de la zone)

f(0) = 100

f(rayonZone) = 25

donc

f(0) = a\*0+b = 100

<=> b = 100

donc

f(rayonZone) = a\*rayonZone+100 = 25

<=> a\*rayonZone = -75

<=> a = -75/rayonZone

donc

f(x) = (-75/rayonZone)x + 100

**Fonctionnement général du programme**

***AURELIEN IL FAUT FAIRE DES TRUCS ICI***