

# Projet C++

B Deporte

Décembre 2020

## 1 Bases épidémiologiques, équations SIR

La propagation du virus est un phénomène épidémiologique connu - la population est ici divisée en trois catégories :

- les individus Sains, ou Susceptibles d'être infectés.
- les individus Contagieux, qui peuvent potentiellement infecter les individus sains, suivant la fréquence des contacts, la mise en place de gestes barrières, etc. NB : les individus Contagieux le sont pendant une durée de contagiosité fixe.
- les individus Recovered, qui ont été contagieux, et ne le sont plus à l'issue de la durée de contagiosité. Ces individus sont alors immunisés à l'infection. On considère dans le modèle codé que l'immunité acquise est définitive. NB : en tout état de cause, un individu 'Recovered' peut également être décédé, mais on n'a pas codé cette issue dans le modèle.

On note  $S_n, C_n, R_n$  la taille des populations de chacune des trois catégories jour  $n$ .

Les 'équations de propagation' du modèle SIR (pour Susceptibles, Infected, Recovered) discret s'écrivent :

$$c_n = I_n * R0 * c * S_n / N \quad (1)$$

$$I_{n+1} = I_n + c_n - I_n[last] \quad (2)$$

$$S_{n+1} = S_n - c_n \quad (3)$$

$$R_{n+1} = R_n + I_n[last] \quad (4)$$

Avec :  $c_n$  le nombre de nouveaux cas du jour, calculé à partir du nombre de contagieux  $I_n$ , du coefficient de propagation (= nombre d'infections potentielles par jour par contagieux)  $R0$ , et rapporté à la part de la population susceptible d'être infectée  $S_n/N$  ( $N$  population totale). (NB :  $c$  est le coefficient d'atténuation, entre 0 et 1)

On a aussi  $I_n[last]$  le nombre de contagieux en 'dernier jour' de contagiosité.  $I_n$  diminue de ce nombre, et  $R_n$  augmente de ce nombre.

Ce modèle simple permet de retrouver facilement : le nombre maximum de nouveaux cas journaliers (= pic épidémiologique), la durée de l'épidémie (lorsque le nombre de nouveaux cas arrive à zéro), ainsi que la part de la population ayant été infectée à l'issue de l'épidémie (= taux 'd'immunité collective').

On peut aussi considérer un modèle SIRS, quand les Recovered perdent leur immunité et redeviennent Susceptibles à l'issue de leur période d'immunité, considérer une propagation par tranches d'âge, voire une charge virale journalière (= contagiosité) non constante pendant la période de contagiosité des Contagieux.

Voir le papier ici :

<https://www.sciencedirect.com/science/article/abs/pii/S0163445320302425>

## 2 Codage

Le modèle reprend les notions d'héritage de classes, de polymorphisme, et de pointeurs, vues en cours.

La classe Population est la classe parent. Elle ne contient qu'un attribut protected 'nombre', avec ses accesseurs 'get-nombre' et 'set-nombre'. La méthode 'affichage' est codée void de façon à avoir le polymorphisme avec les classes filles.

Les classes Susceptibles, Contagieux, et Recovered héritent de la classe Population.

La classe Contagieux a un constructeur dans lequel est créé un tableau dynamique, avec pointeur, du nombre de contagieux infectés depuis  $i$  jours : `cont[i]`. Le destructeur est explicité avec 'delete' pour libérer la mémoire.

Les trois classes contiennent des méthodes permettant de coder l'itération du modèle du jour  $n$  au jour  $n + 1$  :

- 'Susceptibles.evolution' : méthode par laquelle on diminue la taille de la population des Susceptibles, avec une borne inférieure à 0.
- 'Contagieux.evolution' : méthode par laquelle on incrémente d'un jour toutes les catégories de contagieux, en retournant la dernière (=sortie de la période de contagion).
- 'Contagieux.add-new-inf' : accesseur direct à `cont[0]`, qui permet de rentrer les nouveaux cas dans la population Contagieux.
- les méthodes '....affichage()' permettent le polymorphisme avec les pointeurs pointant sur les différentes instances des classes.

Le moteur du modèle est codé dans `main()`. On utilise trois pointeurs `p-s`, `p-c`, `p-r`, sur chacune des trois instances `s`, `c`, `r` de classes, de façon à permettre le polymorphisme.

Il n'y a pas de difficulté dans le codage de l'algorithme lui-même.

La boucle while d'itération s'arrête dès lors que le nombre de nouveaux cas tombe à zéro, et affiche les paramètres clefs de l'épidémie : pic journalier de nouveaux cas, nombre de jours de l'épidémie, taux de population ayant été infectée.