**Benjamin A. ANTUNES – David R.C. HILL**  
Université Clermont-Auvergne, CNRS, Mines de Saint-Étienne, Clermont-Auvergne-INP   
LIMOS UMR 658 - 63000 – ISIMA Clermont-Ferrand, France.

[David.Hill@uca.fr](mailto:David.Hill@uca.fr) – [Benjamin.Antunes@uca.fr](mailto:Benjamin.Antunes@uca.fr)

Résumé : (à la fin)

# **1. Un aperçu de plusieurs approches de modélisation utilisables en l’épidémiologie.**

Quelle que soit l’approche retenue, il serait illusoire de prétendre faire des prédictions en matière épidémiologique. Par contre, il est important d’améliorer notre compréhension des phénomènes épidémiologiques et, dans ce cadre, la simulation est un des rares outils permettant d’appréhender les systèmes complexes. En effet, nous ne sommes pas dans un contexte de modèles physiques déterministes et maîtrisés. De plus, il existe également des modèles déterministes complètement élémentaires qui montrent de façon surprenante et flagrante les limites de nos connaissances (Zwirn 2000 ; 2006) et l’importance des simulations pour explorer le fonctionnement de ces systèmes que l’on nomme complexes (Wolfram 2018). De fait, Christophe Langton propose avec sa fourmi un automate cellulaire bidimensionnel avec un jeu de règles excessivement simples dont le comportement (symétrique, chaotique, infi), ne peut se réaliser qu’à travers une simulation (Gajardo et al. 2002). Il est important de connaître ce type de système avant de se lancer comme des apprentis sorciers dans la programmation du vivant, de loin bien plus obscure que la programmation de la fourmi de Langton qui déjà nous dépasse.

# **2. Related work**

- différents modèles (Refs)

- Why SMA ?

# **3. Présentation du modèle**

- Les paramètres (Refs)

Ce modèle est développé en C++. La reproductibilité des résultats est assurée par l’accessibilité au code via un repository GitHub (lien : xxx) et un notebook Jupyter permettant l’exécution du code de manière très simple. Il sera néanmoins possible d’ajuster les paramètres afin d’observer les différents impacts de ceux-ci. Les paramètres, stockés dans un fichier de configuration, sont les suivants :

- ResistanceInjectionValuesByAge : 8 valeurs flottantes comprises entre 0 et 1, déterminants le taux de résistance à la réinfection en fonction de l'âge. Il y a huit tranches d’age : 0 - 15, 15 - 25, 25 - 35, 35 - 45, 45 - 55, 55 - 65, 65 - 75, 75 +. A partir de 90 jours après l’infection (source ?), l’immunité va diminuer de 0.001 (source ?) chaque jour.

- MaxResistanceInjectionValuesByAge : 8 valeurs flottantes comprises entre 0 et 1, déterminants le taux maximum de résistance après une vaccination complète en fonction des huit tranches d’âges. Cette résistance diminuera de manière similaire que précédemment.

- MinResistanceInjectionValuesByAge : 8 valeurs flottantes comprises entre 0 et 1, déterminants le taux minimum de résistance après une vaccination complète en fonction des huit tranches d’âges. Cette résistance diminuera de manière similaire que précédemment.

- TableTauxMortaliteByAge : 8 valeurs flottantes comprises entre 0 et 1, déterminants le taux théorique de mortalité par tranche d’âge. C’est sur ces valeurs que l’on s’appuie pour déterminer le taux de passage à l’hôpital, puis en réanimation et enfin à la mort.

//- TauxContaminationVoisin : n’est plus utilisé je pense

- NbPlaceHospital : Valeur entière déterminant le nombre maximal de place disponible à l’hôpital.

- NbPlaceReanimation : Valeur entière déterminant le nombre maximal de place disponible en réanimation.

- R0 : Valeur flottante permettant de fixer le nombre de contamination d’un individu durant la maladie au départ de la simulation. Ce paramètre n’est plus utilisable directement, il a été utilisé pour fixer un R0 de départ.

- TauxMortRea : Valeur flottante déterminant le pourcentage de chance de mourir lorsque l’on est en réanimation.

- Size : Valeur entière déterminant la carte de la carte. La carte est un carré, et ce paramètre défini la longueur des côtés.

- NbMalade : Valeur entière déterminant le nombre d’humains malade au départ de la simulation.

- NbHumain : Valeur entière déterminant le nombre d’humain présent au total dans la simulation au départ. Le nombre d’humain sain au départ sain au départ est donc le nombre d’humain total moins le nombre d’humain malade.

- NbIteration : Valeur entière déterminant le nombre d’itération que va effectuer la simualtion.

- NbMultMortToHosp : Valeur entière. Le taux de chance d’être hospitalisé en fonction de l tranche d’age après avoir contracté la maladie est basé sur le taux de mortalité défini en paramètre. NbMultMortToHosp est un facteur de multiplication. Dans notre cas il est fixé à 4. Par exemple, si la probabilité de mourir pour une personne de + 75 ans est de 15%, alors ces chances d’être hospitalisé sont de 60%.

- IsVaccin : Valeur entière étant 0 ou 1 (booléen). Lorsque la valeur est fixée à 1, alors au départ de la simulation, un pourcentage de la population sera complétement vacciné.

- HistogrammeContamination : Tableau de 11 valeurs flottantes déterminant les probabilités de contamination des voisins en fonction du stade de la maladie. Nous considérons que les humains sont contagieux deux jours avant les symptômes, puis neuf jours après, soit onze jours au total.

- PourcentAsymptomatique : Valeur flottante determinant le taux de chance de rester asymptomatique après deux jours de maladie.

- TauxDeChanceDeMourirHospitalHull : Valeur flottante déterminant la probabilité qu’un humain a de mourir s’il doit aller à l’hôpital mais qu’il n’y a plus de place disponible.

- TauxVaccination : Valeur flottant déterminant le taux de vaccination de humains au départ de la simulation (si le paramètre IsVaccin est mis à 1).

#######

* Ajout parametre taux de personne par age ?
* L’utilisation des parametres avec le taux de mort est bizarre je trouve : On a le taux de mortalité dans la simu, mais on ne l’utilise pas pour faire mourir les gens, mais pour les hospitaliser, ce qui au final fait qu’on a un taux de mortalité différent. Qui est d’ailleurs maintenant plus proche de la réalité que le taux en paramètre qui est peut etre trop élévé maintenant (moins de mort qu’au debut de l’épidémie). Donc c’est bizarre comme fonctionnement.
* Nom de parametre mélange français et anglais, c’est moisi ?
* On a un taux de mortalité un peu trop bas
* Mais en meme temps, ce sont les asymptomatiques qui contaminent le plus de gens car ils se déplace, mais dans la réalité ils sont surement beaucoup moins contagieux que les symptomatiques, donc HistogrammeContamination pourrait etre différent. Mais alors si on contamine moins, on aura encore moins de mort à la fin. (Mais ajustable avec le R0, on rajoute est iteration).
* Ce principe de R0 n’est peut etre pas hyper clair.
* Avec la vaccination / infection, on est protegé contre la ré-injection. Néanmoins, on voit qu’avec le vaccin, on est apriori pas vraiment protégé contre la ré-infection mais plutôt contre les formes graves/hopital/mort. Donc, peut etre modifier ce fonctionnement ? (Actuellement, on est protégé contre la réinfection, mais si on est réinfecté, alors tout se passe comme les autres).
* Après avoir regardé rapidement Didier Raoult (Je ne suis pas expert en épidémie, d’autres le sont), il suggère qu’il n’y a pas eu plusieurs vagues du meme virus, mais bien plusieurs vagues car plusieurs virus (cf debut : <https://www.youtube.com/watch?v=wCBZ0NYqewU> ). Dans mon modele a depart j’avais qu’une seule vague rapide, quand la densité était assez elevé. Mais, après avoir baissé la densité avec un R0 de 3, ce n’est pas une seule vague. Devrait-on travailler sur la modélisation afin que l’on est réellement 1 seule vague sur 4 mois en moyenne, avec biensur des fluctuations du a l’aléatoire, avec une timeline des variants permettant d’observer plus vagues ? Comment faire ça ? Puisque dans notre cas, une vague s’arrete car tout le monde est contaminé. Or dans la vraie vie ce n’est pas le cas, et c’est d’ailleurs cela qui permet à la vague suivante d’exister. Car si tout le monde était contaminé, variant ou pas, nous serions trop immunisés pour faire repartir une nouvelle vague, je suppose ici que le système immunitaire est capable de se défendre contre les variants. Donc le seul moyen pour le modeliser serait d’avoir une resistance propre a chaque variant dans notre simulation, mais dans ce cas le nombre de contamination ne serait pas du tout representatif de la realité. D’ailleurs, c’est déjà le cas puisque dans la vraie vie seulement 10% des français ont réellement attrapé la maladie. Une explication pourrait etre que Raoult se trompe, il n’y a pas eu plusieurs vagues de virus différents, mais bien plusieurs vagues du meme virus, sauf que un variant, grace a un avantage evolutif, va supplenter un autre et va le faire disparaitre. Mais les contaminations ont lieu en continu sans arret. Auquel cas, dans notre modelisation, il n’y a aucun interet a modeliser les variants, mais garder un esprit de plusieurs vagues. Raoult ne se trompe pas vraiment, mais en gros j’ai le sentiment que les variants ne change rien : Il y a des vagues de variant qui ecrasent les autres grace a leurs avantages evolutifs, mais si des variants n’étaient pas apparu, alors l’épidémie ne se serait pas arrêtée net en faisant qu’une seule vague, le meme variant en aurait fait plusieurs. Qu’en pensez vous ?

#######

- Les choix de modélisation (temps, espace, et utilisation des paramètres…) – Refs

Ce modèle part d’une base écrite en Python par XXXX (Source XX). (Utile de préciser ça ? Surement pas). Le modèle est basé sur une grille carrée dans laquelle des humains vont se déplacer (Paramètre : Size). Chaque itération de la simulation correspond à un jour (Paramètre : NbIteration). Chaque jour, des comportements propres aux états des humains vont être mis en place : Dans un premier temps, les humains sains vont se déplacer sur la carte. Ils se déplacent de manière aléatoire potentiellement sur toute la carte. Pour la création de l’aléatoire, nous utilisons le générateur Mersenne Twister de Makoto Matsumoto. Les humains contaminés vont, quant à eux, se déplacer plusieurs fois sur la carte. Le nombre de déplacement est basé sur un R0 de 3, ce qui nous donne 12 déplacements journaliers pouvant potentiellement contaminer d’autres individus (Paramètre : HistogrammeContamination). Si les individus contaminés le sont depuis plus de deux jours, alors un choix s’opère : Soit l’individu reste asymptomatique et il peut continuer à se déplacer et contaminer (Paramètre : pourcentAsymptomatique), soit l’individu devient symptomatique, auquel cas il ne se déplace plus et ne contamine plus (il se confine). Au bout du 3eme jours, un test est effectué pour les humains confiné afin de savoir s’ils doivent être hospitalisé ou non en fonction de leur âge (Paramètres : MultMortToHosp). Si l’hôpital est plein, alors l’individu à une chance de mourir (Paramètre : TauxDeChanceDeMourirHospitalHull ). Au 4eme jours, les humains hospitalisés ont une chance d’entrer en réanimation. Les humains hospitalisés restent à l’hôpital du 4eme au 12eme jours (### a reduire ?). Une fois en réanimation, les humains ont une probabilité de mourir (Param : TauxMortRea) indépendamment de leur âge. On considère qu’une fois en réanimation, tout le monde à approximativement les mêmes chances de s’en sortir. Les individus peuvent rester en réanimation ensuite entre 5 et 21 jours (Source XX).

# **4. Résultats et discussion**

Ce modèle

# **5. Conclusion**

- Bilan

- Perspectives

# **7. Références**

(Gajardo et al. 2002) GAJARDO, Anahi, MOREIRA, Andre, et GOLES, Eric. Complexity of Langton's ant. Discrete Applied Mathematics, 2002, vol. 117, no 1-3, p. 41-50.

(Wolfram 2018) WOLFRAM, Stephen. Complex Systems Theory 1. In : Emerging syntheses in science. CRC Press, 2018. p. 183-190.

(Zwirn 2000) ZWIRN, Hervé. Limites de la connaissance (Les). Odile Jacob, 2000.

(Zwirn 2006) ZWIRN, Hervé. Systèmes complexes (Les): Mathématiques et biologie. Odile Jacob, 2006.

(Kretzschmar 2008) Kretzschmar Mirjam, « Infectious disease modeling », International encyclopedia of public health, éd. Kris Heggenhougen, 2008, p. 569-576.

(Krämer et al 2010) KRÄMER, Alexander, KRETZSCHMAR, Mirjam et KRICKEBERG, Klaus (ed.). Modern infectious disease epidemiology: Concepts, methods, mathematical models, and public health. Springer Science & Business Media, 2010.

(Hethcode 2000) HETHCOTE Herbert W., « The Mathematics of Infectious Diseases », SIAM Review, volume 42, numéro 4, pages 599-653, 2000.