



RAPPORT TRAVAUX PRATIQUE SMA

MASTER I - SYSTEMES INTELLIGENTS ET
MULTIMEDIA (SIM)

SIMULATION DE LA PROPAGATION D'UNE ÉPIDÉMIE DANS UNE VILLE

Etudiant :

KANA NGUIMFACK KEVIN

Superviseur :

Pr. MANH HUNG NGUYEN

8 mai 2020

Table des matières

1	Introduction	4
2	Analyse du problème	4
2.1	Définition	4
2.2	Présentation des composants géographique GIS	4
3	conception de notre système simulation	7
3.1	Modèle 01	7
3.1.1	Objectifs du modèle	7
3.1.2	Description du modèle	7
3.1.3	section global du modèle	8
3.1.4	species road	9
3.1.5	species building	9
3.1.6	species people	9
3.1.7	Observation résultats	10
3.1.8	Modèle 02	11
3.1.9	Objectifs du modèle	11
3.2	Description du modèle	11
3.2.1	section global du modèle	11
3.2.2	species hospital	11
3.2.3	species doctor	12
3.2.4	Observation résultats	13
3.3	Modèle 03	13
3.3.1	Objectifs du modèle	13
3.3.2	Description du modèle	13
3.3.3	scénarios avec distance sociétale élevé 10 mètres	14
3.3.4	scénarios avec distance sociétale très faible	14
4	Conclusion	16

Liste des tableaux

1	Variables globales modèle 01	8
---	--	---

Table des figures

1	vue globale portion géographique de Berlin	5
2	vue réseau routier de notre environnement	5
3	vue des différents secteurs de l' environnement	6
4	vue d'ensemble de l'environnement de simulation	6
5	exemple simulation modèle 01	10
6	exemple simulation modèle 02	13
7	scénarios avec distance sociétale très grande	14
8	étape 1 : initialisation scénarios avec faible distance sociétale	15
9	étape 2 : après une certaine durée scénarios avec faible distance sociétale .	15

1 Introduction

Depuis des années, le monde est de plus en plus dévasté par des épidémies à l'instar de la peste, le choléra, la grippe espagnole, la grippe asiatique. Et aujourd'hui le coronavirus baptisé **COVID-19** qui fait des ravages plus que les précédentes depuis décembre 2019. Dès lors les recherches en médecine ne cessent d'être approfondir et les stratégies sont de plus recherchées pour mieux cerner ces épidémies. Les systèmes de simulations s'inscrivent comme un excellent moyen à fin d'organiser au mieux une stratégie de défense contre ces épidémies. C'est dans ce contexte que nous sommes amenés à concevoir un système de simulation de propagation dans une ville d'une épidémie. Notre travail se décline en trois parties, une partie analyse qui présente notre approche, une partie conception qui présente l'implementation de notre système et une dernière interprétation des résultats obtenus.

2 Analyse du problème

Dans le cadre de notre travail, dont le but est de réaliser une simulation de la propagation d'une épidémie, nous utilisons à cet effet **GAMA** un environnement de développement de modélisation et de simulation basé sur des agents. Nous sommes amenés à réaliser une simulation basée sur des **agents** et **GIS** (Geographic Information System).

2.1 Définition

Notre environnement choisi pour simuler la propagation d'une épidémie, est la ville de Berlin en Allemagne, plus précisément un échantillon de cette ville du fait des limites des capacités de notre ordinateur. Nous admettons que cette épidémie se transmet par dans l'air et l'infection est toutefois possible si un individu normal se situe au maximum à deux mètre d'un autre individu infecté. Et les individus infectés peuvent guérir s'ils sont pris en charges dans un hôpital, ou soient mourrir malgré la prise en charge. Les agents de notre environnement sont des personnes, des routes, des batiments et des lieux (représentant les partitions d'une ville).

- **Personnes** : représentent un ensemble d'individus dans notre environnement, identifié par son âge, son habitation, son district, ses losirs, son lieu d'étude, son lieu commercial, son lieu de services et son état de santé (en forme ou infecté). Chaque personne est affiliée à un centre hospitalié en guise d'assurance maladie.
- **Médecins** : représente un cas particulier de personnes dont le but est de guérir les personnes malades. un médecin prend fonction au sein d'un hôpital et detient tous les capacités necessaires pour soigner des personnes infectées par l'épidémie.

La section suivante est dédiée exclusivement à la présentation des données géographiques.

2.2 Présentation des composants géographique GIS

Dans cette section nous présentons l'espace géographique d'une portion de la ville de Berlin.

Nous avons utilisé le logiciel **QGIS** pour nos manipulations. la figue suivante montre portion de notre environnement avec routes, bâtiments et lieux.

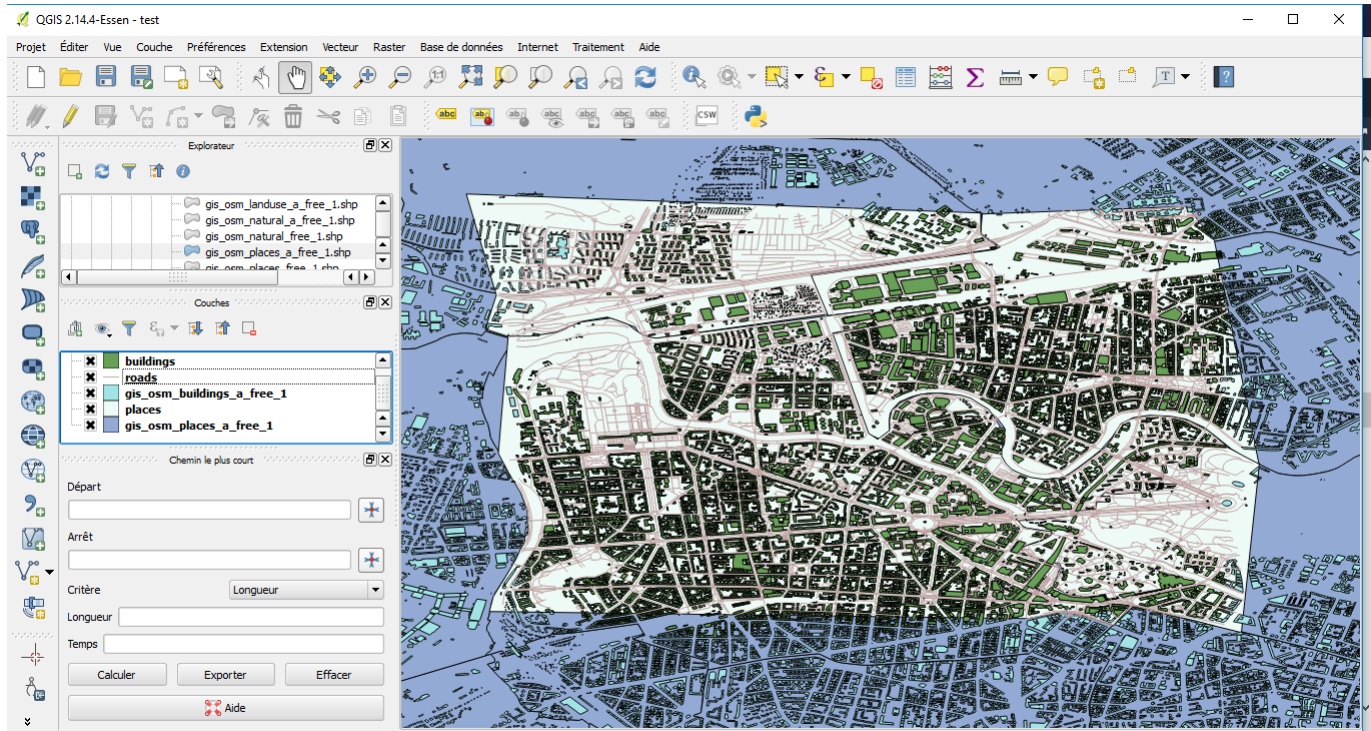


FIGURE 1 – vue globale portion géographique de Berlin

L'ensemble des routes constitue notre réseau routier. C'est l'ensemble des chemins de notre environnement.

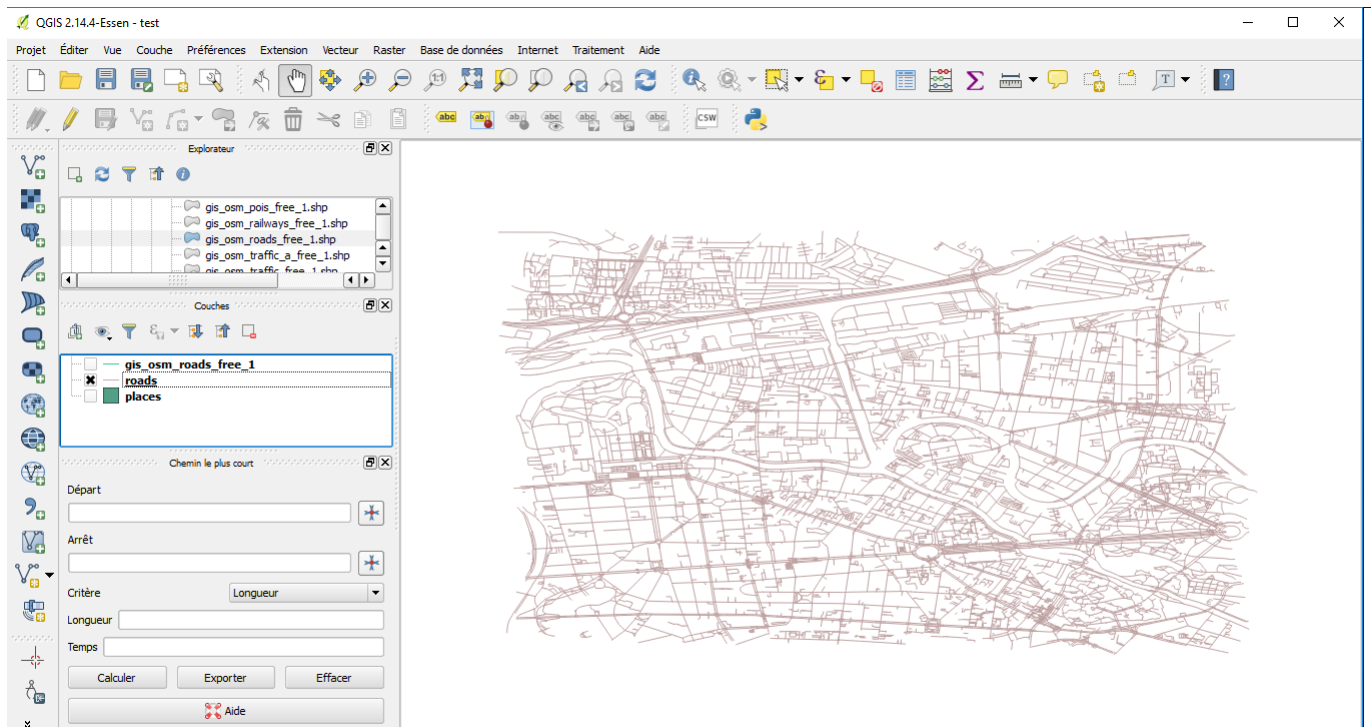


FIGURE 2 – vue réseau routier de notre environnement

Notre environnement est partitionné en secteurs représentant des lieux.

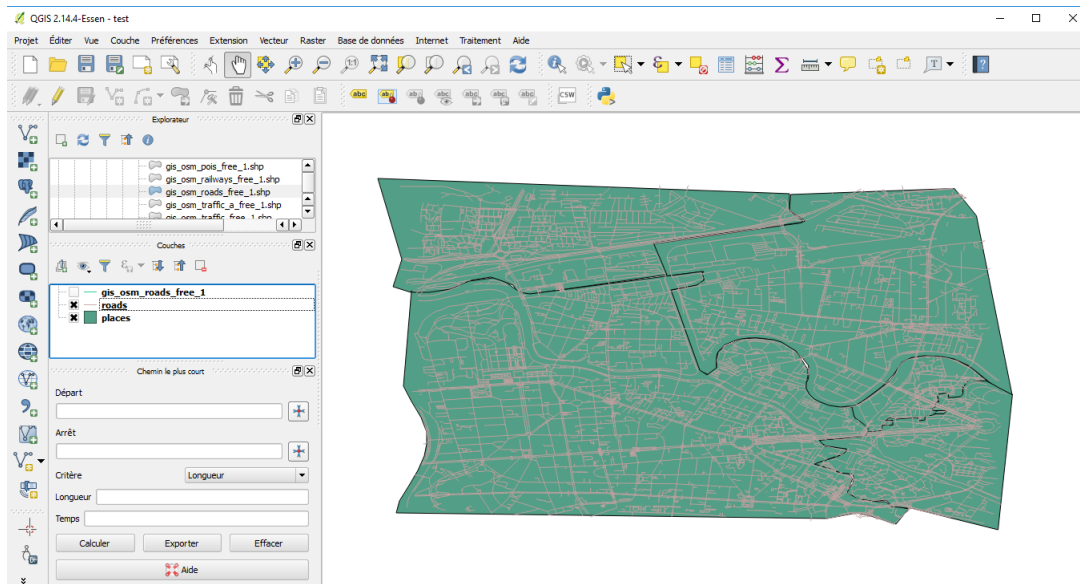


FIGURE 3 – vue des différents secteurs de l' environnement

Les bâtiments représentent des habitations, des services, des écoles, des détenteurs (sports, parc, espaces enfants, etc...), des centres commerciaux (supermarché, restaurants, etc...) et des hopitaux.

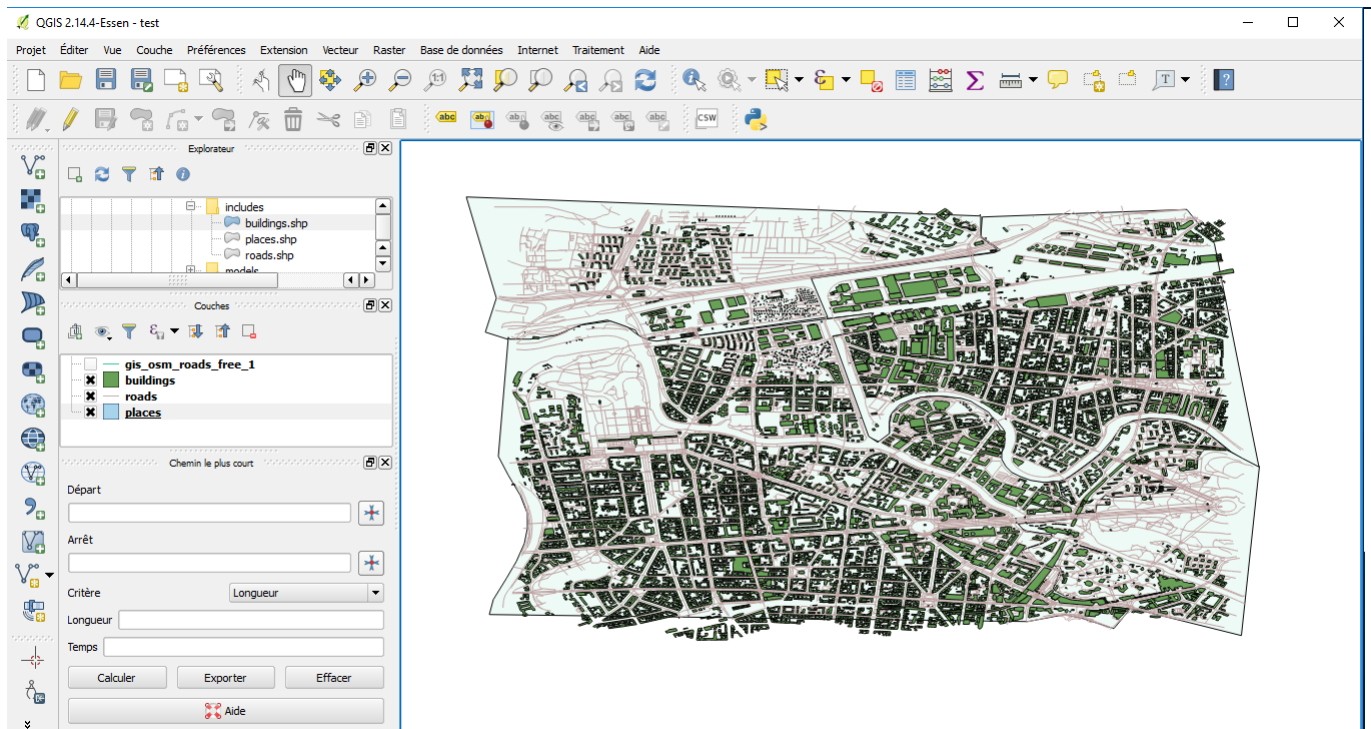


FIGURE 4 – vue d'ensemble de l'environnement de simulation

3 conception de notre système simulation

Notre conception se décline en trois modèles. Chaque modèle représente un fichier de notre projet avec extension **gaml** et dont le contenu comporte trois type d'éléments. Le premier élément est la section **global** qui représente un agent particulier, notre **environnement**, c'est à l'intérieur de ce dernier que nos agents sont instanciés. Le deuxième, les **species** pour représenter les différentes agents dans l'environnement, chaque agent est défini par un ensemble d'attributs, d'actions, de comportements et sa géométrie (sa représentation). Et le dernier est la section **experiment** pour afficher les agents, il se définit par des entrées (**inputs**) et des sorties (**outputs**).

3.1 Modèle 01

3.1.1 Objectifs du modèle

- Dans cette première partie de notre projet de simulation, il est question pour nous de :
- Intégrer les fichiers de données **GIS** au sein de notre environnement
 - Créer les agents de notre monde, les **personnes**

3.1.2 Description du modèle

Les fichiers GIS représentés par les **shapefiles** sont importés dans le modèle dans la section **global**, les **routes**, **bâtiments** et les **lieux**. Un ensemble de spécifications est pris en compte notamment :

- étape de simulation (**step**) 10 minutes.
- une date de début de simulation fixée ici par exemple au **2020-01-01-00-00-00**.
- notre environnement est représenté par une topologie de type **graphe**
- les personnes se déplacent sur les routes en choisissant toujours le plus court chemin.
- chaque personne est décrite par quatre niveaux représentés par les couleurs **cyan** (en forme), **violet** (infectée au niveau 1), **rouge** (infectée au niveau 2) et **noire** (mort).
- chaque personne a au moins un domicile, un travail, un divertissement, une fréquentation commerciale et affiliée à un hôpital.
- initialement chaque personne se trouve dans son habitation
- chaque personne se déplace à vitesse constante
- les personnes sont définies par une heure de début d'activité entre 6 et 8 heures, et une heure de fin d'activités entre 17 et 21 heures.
- chaque personne a une pose déjeunée qui est la moyenne du temps de début et fin d'activités
- les personnes étudiantes vont à l'école, les travailleurs au travail et les plus âgés dans les lieux de divertissement.
- chaque personne reste dans un bâtiment un certain nombre de temps puis choisit une autre activité.
- travailleurs et étudiants finissent leur activités avant de choisir une autre activité.
- chaque personne devra être de retour dans son habitation au maximum à l'heure maximale de fin d'activité.
- les personnes âgées ont plus de chance d'être infectées.
- la probabilité de contracter la maladie est initialisée à **0.3**

- la probabilité de se déplacer est initialisée **0.05**.
- chaque personne infecté peut transmettre la maladie dans un voisinage maximum 2 mètres.

3.1.3 section global du modèle

le modèle se décrit comme suite, dans la partie **global** nous avons un ensemble de variables.

Définition des Variable		
type	variables	signification
float	step	etape de simulation
int	nb_people	nombre de personnes
int	nb_people_infected	initialisation nombre de personnes infectées
float	move_probability	probabilité de sortir du bâtiment
int	distance_neighbors	distance maximale de contamination
float	contract_disease_probability	probabilité pour une personne d'être infectée
date	starting_date	date debut simulation
int	min_activity_start	heure minimale debut activité
int	max_activity_start	heure maximale debut activité
int	min_activity_end	heure minimale fin activité
int	max_activity_end	heure maximale fin activité
int	min_speed	vitesse minimale
int	max_speed	vitesse maximale
file	roads_shape_file	données géographique du réseau routier
file	places_shape_file	données géographique des districts de la ville
file	buildings_shape_file	données géographique d'emplacement des bâtiments
graph	roads_topology	réseau routier de la ville
list	works_list	liste des différents type de service
list	works_list	liste des différents type bâtiments de service
list	schools_list	liste des différents type d'écoles
list	commercials_list	liste des différents centres commerciaux
list	residences_list	liste des différents type de maison
list	hobbies_list	liste des divertissements
list	residences_list	liste des différents type d'hôpitaux
list	residences_list	liste des différents lieux
geometry	shape	définition du monde à partir des shapefiles

TABLE 1 – Variables globales modèle 01

Dans le bloc **init**, on procède à l'instanciation des agents, **people**, **road**, **building** et **place**. Dans notre monde, les ecoles ont une couleur **jaune**, les services en **bleu**, les hôpitaux en **vert**, les habitations en **marron**, les centres commerciaux de couleur **olive** et les hobbies en **orange**.

3.1.4 species road

notre agent **road** a un seul attribut **display_shape** de type **geometry** qui represente le shapefile de la route . notre réseau routier est un ensemble de route représenté par un graphe, et chaque route est une arête du graphe dont les extrémités sont les noeuds. Les routes ont des lignes en représentation en 3D de couleur noire.

3.1.5 species building

Chaque building est identifié par :

- **type_building** : le type du bâtiment (école, service,etc...).
- **name_building** : le nom du batiment, utile pour effectuer des filtres lors de l'instanciation.
- **color** : couleur du bâtiment.

Notre agent a une représentation en 3D de couleur initiale **grise** à partir du shapefile buildings.

3.1.6 species people

Chaque agent est défini par les attributs suivants :

- **speed** : la vitesse de l'agent de type **float**. Elle est constante, et se déduit entre **min_speed** et **max_speed**.
- **age** : l'âge de l'agent entre 10 et 100 ans, de type **int**.
- **is_infected** : vrai si l'agent est infecté et faux sinon, de type **bool**.
- **color** : la couleur de l'agent.
- **building_work** : de type **building**, le bâtiment de travail de l'agent.
- **building_living** : de type **building**, le bâtiment de résidence de l'agent.
- **building_school** : de type **building**, le bâtiment d'étude de l'agent.
- **building_eating** : de type **building**, le bâtiment de restauration de l'agent.
- **building_hobby** : de type **building**, le bâtiment de détente de l'agent.
- **building_school** : de type **building**, le bâtiment d'étude de l'agent.
- **building_hospital** : de type **building**, le bâtiment d'assurance santé de l'agent.
- **building_goal** : de type **building**, le bâtiment cible de l'agent.
- **goal** : de type **point**, la destination de l'agent.
- **tmp** : de type **point**, la destination précédente de l'agent (hôpital, travail, école).
- **begin_activity**. de type **int**, et se déduit entre **min_activity_start** et **max_activity_start**.
- **end_activity**. de type **int**, et se déduit entre **min_activity_end** et **max_activity_end**.

Chaque agent est défini également par un ensemble de comportements définis par le mot clé **reflex** :

- **runActivity** : exécuté lorsque **current_date.hour**(heure courante) est égal **begin_activity**. En fonction de l'âge la variable **goal** est initialisée par une cible, soit par l'école pour les étudiants, soit par le travail pour les travailleurs et soit par les divertissements pour les plus âgés.
- **backHome** : exécuté lorsque **current_date.hour** est égal **begin_activity** ou bien **current_date.hour** inférieur à **max_activity_end**. Il oblige l'agent de retrouve son habitation si l'heure maximale **max_activity_end** est atteinte.

- **breakToEat** : exécuté à une heure moyenne de l'activité de l'agent, pour qu'il parte manger. Toutefois sa localisation précédente est conservée dans la variable **tmp**.
- **moveToBuilding** : exécuté quand l'agent à un but. Une fois arrivée à destination la variable **goal** est remise à nulle (**nil**).
- **stayInBuilding** : exécuté lorsque l'agent est à l'intérieur d'un bâtiment et l'heure de fin des activités non atteinte.
- **spreadingDisease** : exécuté lorsque l'agent est infecté, alors il peut infecter son voisinage avec une probabilité défini **contract_disease_probability** pour les personnes de moins de 30 ans. Et l'inverse pour les personnes de moins de 70 ans et les autres de plus de 70 ans sont d'office infecté. Nous partons de l'hypothèse que le taux d'infection est d'autant plus faible que la population. Toutefois Les personnes pouvant être infectées sont des personnes en forme de couleur **cyan**.

chaque agent est représenté par une sphère dans notre monde 3D. La couleur de l'agent lors d'une instanciation est soit **cyan** si en bonne santé. Et s'il est infecté, il peut avoir la couleur **violet** avec une probabilité de **contract_disease_probability**, sinon 80% pour la couleur **rouge** et sinon décède directement.

3.1.7 Observation résultats

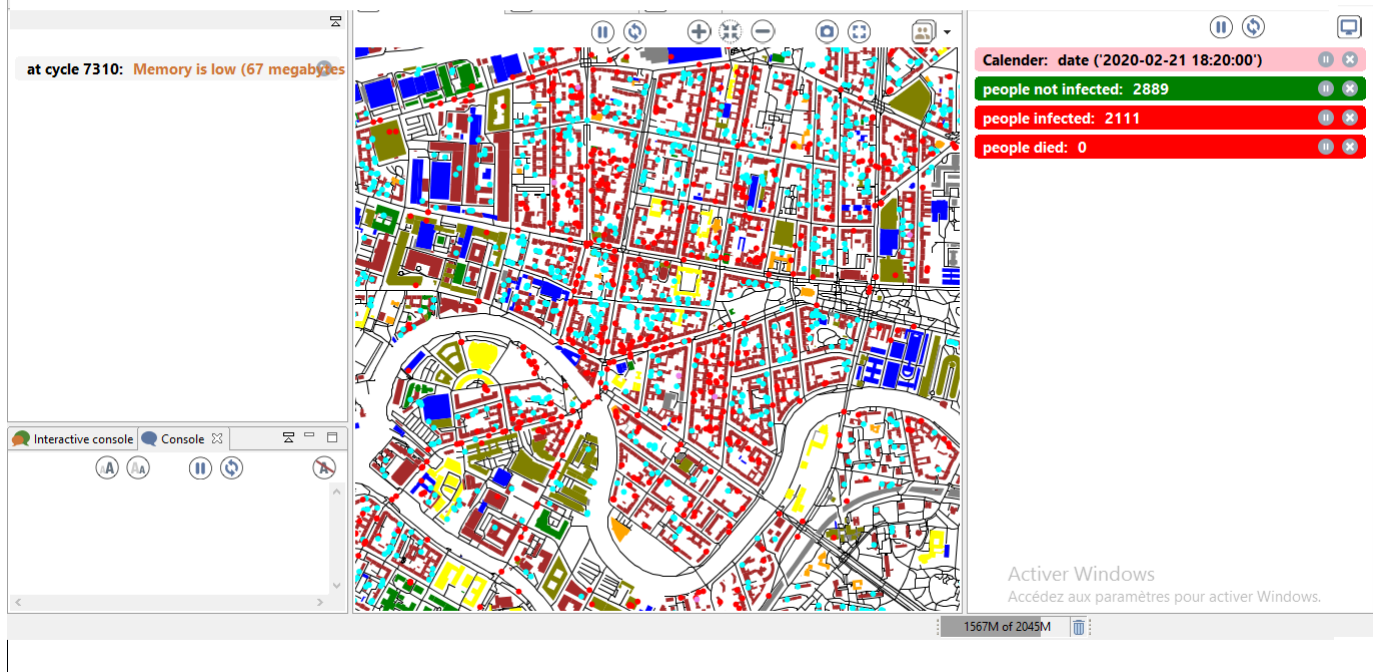


FIGURE 5 – exemple simulation modèle 01

3.1.8 Modèle 02

3.1.9 Objectifs du modèle

Le modèle précédent étant achevé, dans cette deuxième partie de notre projet, il est question pour nous de :

- Créer les médecins capable de guérir les personnes infectées. la vitesse de guérison dépendre du niveau de santé et du niveau de la maladie de chaque personne.
- S'assurer que toute personne infectée, se rend à l'hôpital pour des soins
- Créer des agents tels que des hôpitaux qui combattent la maladie.
- Initialiser l'épidémie dans une place en ville
- Mettre une zone en quarantaine pour un nombre de personne préalablement défini.

3.2 Description du modèle

Un ensemble de spécifications pour ce modèle est également à prendre en compte :

- une personne infectée a la possibilité de guérir
- la personne infectée devra se rendre dans un hôpital qui combat la maladie
- un médecin peut guérir en fonction de l'état de santé et du niveau de maladie de chaque personne.
- tous les hôpitaux combattent la maladie
- les hôpitaux peuvent échanger les informations concernant la maladie
- un médecin ne peut contracter la maladie, car toutes les mesures de protection sont prises par ce dernier.
- le médecin est une personne, donc possède les attributs et comportements d'un agent **people**
- l'agent **doctor** hérite de l'agent **people**
- un médecin peut mettre une zone en quarantaine
- un médecin prend fonction dans un hôpital
- un hôpital est un agent qui communique avec les autres hôpitaux.

3.2.1 section global du modèle

Dans cette section une mise à jour est faite. De nouvelles variables sont définies :

- **threshold_for_quarantine** : de type **int**, initialisée avec une valeur seuil de personnes infectées pour la mise d'une zone en quarantaine.
- **nb_doctors** : de type **int**, initialisée avec le nombre de médecin à affecter par hôpital.

Dans la section **init**, les instances d'hôpitaux associés aux bâtiments ont été créés. De même un nombre **nb_doctors** de docteurs a été créé dans chaque agent **hospital**. Chaque agent **doctor** est instancié avec initialisation d'attributs hérités de l'agent **people**, mais notons bien que le nombre de **doctor** ne font partir de la population **people**. Dans cette section également nous initialisons le nombre de personnes infectées dans une zone spécifique, ce nombre est variable en fonction du nombre de personne dans la zone ciblée.

3.2.2 species hospital

Cet agent hérite de l'agent **building** et chaque building initialement référencé comme un hôpital, devient un agent **hospital**. L'agent **hospital** en plus d'hériter des attributs

de la classe **parente**, défini un nouvel attribut **Hospitals** représentant la liste des agents **hospital**.

L'agent détient la capacité de communiquer avec d'autres agents **hospital**, cette capacité lui est attribué grâce au **skill fipa**, le protocole **fipa-contract-net** est utilisé pour cette interaction. Deux comportements définissent cette communication :

- **send_infos** : exécuté avec une probabilité de 50%, permettant à l'agent de transmettre des informations sur la maladie aux autres.
- **receive_infos** : exécuté par l'agent pour acquiescer une reception d'informations des autres agents lorsque la variable **proposes** propre au skill représentant les messages entrants est non vide.

3.2.3 species doctor

La définition de cet agent apporte quelques mise à jours aux agents du modèle précédent.

- **is_quarantine** : attribut de booléen initialisé à faux est ajouté à l'agent **place**, permettant si de dire si la zone a été placée en quarantaine.
- **stayInBuilding** : le reflex de l'agent **people** est légèrement modifié en redirectionnant les personnes en zone de quarantaine vers leur habitation respective.
- **runActivity** : Desormais une personne saine se voit attribuer un objectif, toute personne malade devra se rendre à l'hôpital pour être pris en charge.
- **spreadingDisease** : le reflex exécuté en cas d'infection avec la possibilité d'infecter des personnes à une distance de valeur **distance_neighbors** mais aussi l'agent infecté doit s'y rendre à l'hôpital dans lequel il est affilié. Il ne peut infecter les personnes soignants, seulement ceux de couleur **cyan**.

Un agent **doctor** hérite de l'agent **people**, en plus des attributs hérités , un nouvel attribut **quarantine_zone** représentant la liste des agents **place** est ajouté. Certains comportements hérités sont redéfinis.

- **runActivity** : dans lequel la variable **goal** est mise à **nil**, car chaque agent **doctor** est initialement dans l'hôpital de prise de service.
- **stayInBuilding** : donc le corps est vide car l'agent doit rester administrer les soins aux malades.
- **backHome** et **breakToEat** : dans lesquels la variable **goal** est mise à **nil**, nous estimons que tous ceux dont il a besoin se trouve dans son lieu de service et garanti que l'hôpital soit opérationnel 24 heures (cela élimine la relève).

De nouveaux comportements sont également définis :

- **cure** : exécuté tant l'agent se trouve dans son service, permettant une administration des soins aux malades. Leur vitesse de guérison est fonction du niveau de leur maladie, un malade de niveau 1 à plus de chance de guérir que celui de niveau 2. Le niveau de maladie peut en pirer jusqu'à la mort du malade.
- **quarantine** : exécuté avec une probabilité de 50%, permettant de mettre une zone en quarantaine lorsque le nombre de personnes infectés est supérieur à la valeur seuil **threshold_for_quarantine**. La variable **goal** contient les coordonnées de la zone et la liste des zones en quarantaine est mise à jour.
- **cure_in_zone** : exécuté quand la liste est non vide, l'agent se rend dans la zone en quarantaine pour soigner les personnes infectées.

Les médecins sont représentés par une sphère de couleur **blanche**.

3.2.4 Observation résultats

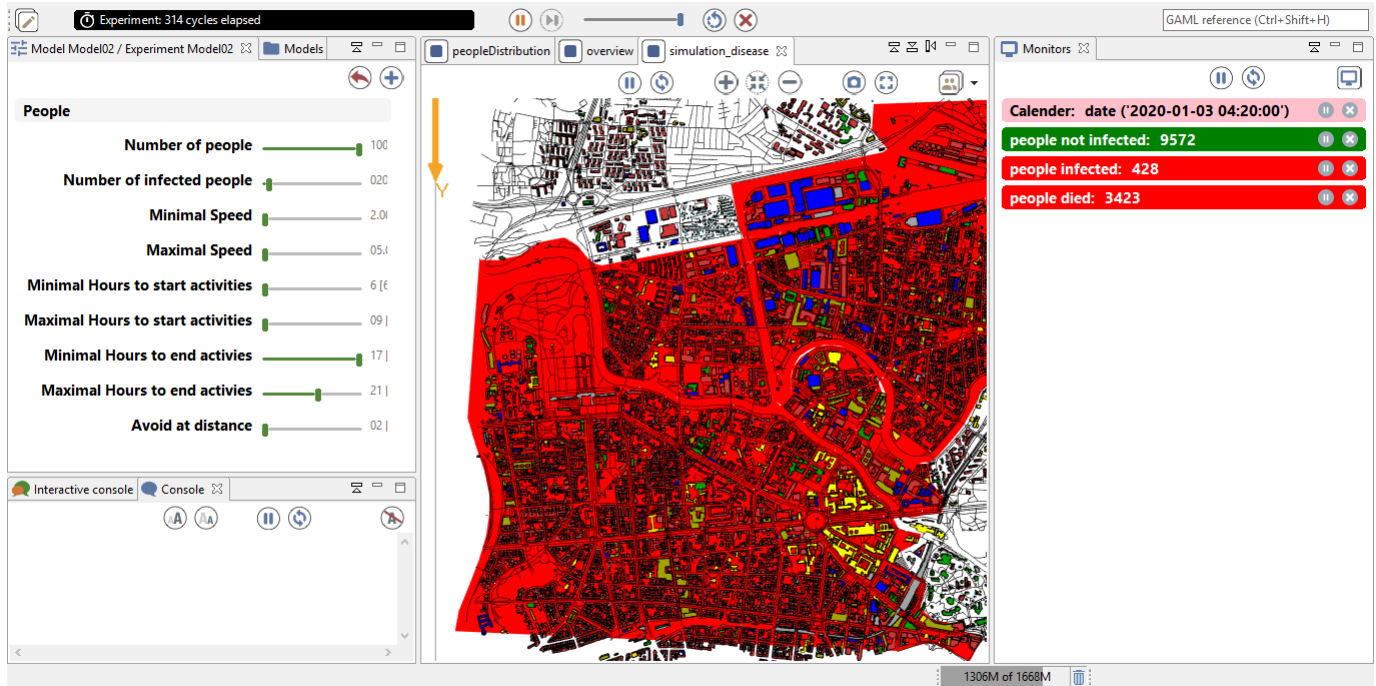


FIGURE 6 – exemple simulation modèle 02

3.3 Modèle 03

3.3.1 Objectifs du modèle

Il est question dans cette partie de :

- Proposer des scénarios pour lutter efficacement contre la maladie.
- Définir une distribution des hôpitaux pour lutter efficacement contre la maladie
- Fournir une étude statistique des différents scénarios.

3.3.2 Description du modèle

Pour atteindre ses objectifs un ensemble de spécifications sont définies :

- à fin de rendre notre lutte efficace, nous supposons que tous les centres de santé ont acquis le nécessaire pour lutter efficacement contre l'épidémie.
- plus le rayon de contamination est grand plus l'épidémie prend de l'ampleur.
- plus la distance sociale est petite, plus le combat contre l'épidémie fournit des résultats satisfaisants.
- Plus la population devient consciencieuse, en prenant des mesures de précaution tels que le port des masques car la maladie est transmise par les airs et le respect des règles d'hygiène. Plus le taux de contamination baisse.

3.3.3 scénarios avec distance sociale élevé 10 mètres

. Nous constatons que le nombre de cas positive ne cesse d'accroître constamment. Alors plus le champ de contamination est grande , voir infini, donc nous avons moins de chance de maîtriser l'épidémie.

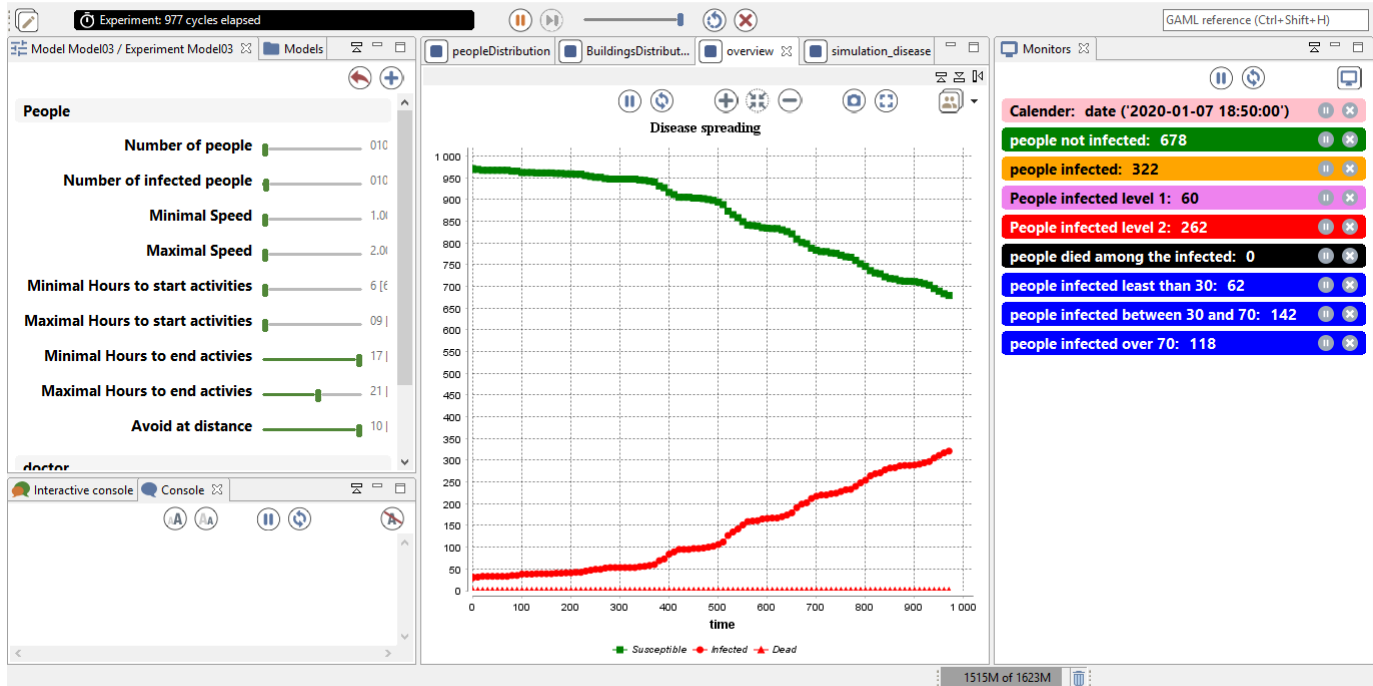


FIGURE 7 – scénarios avec distance sociale très grande

3.3.4 scénarios avec distance sociale très faible

L'épidémie progresse faiblement, et on a plus de chance de vaincre rapidement la maladie par une prise en charge rapide des personnes infectées.

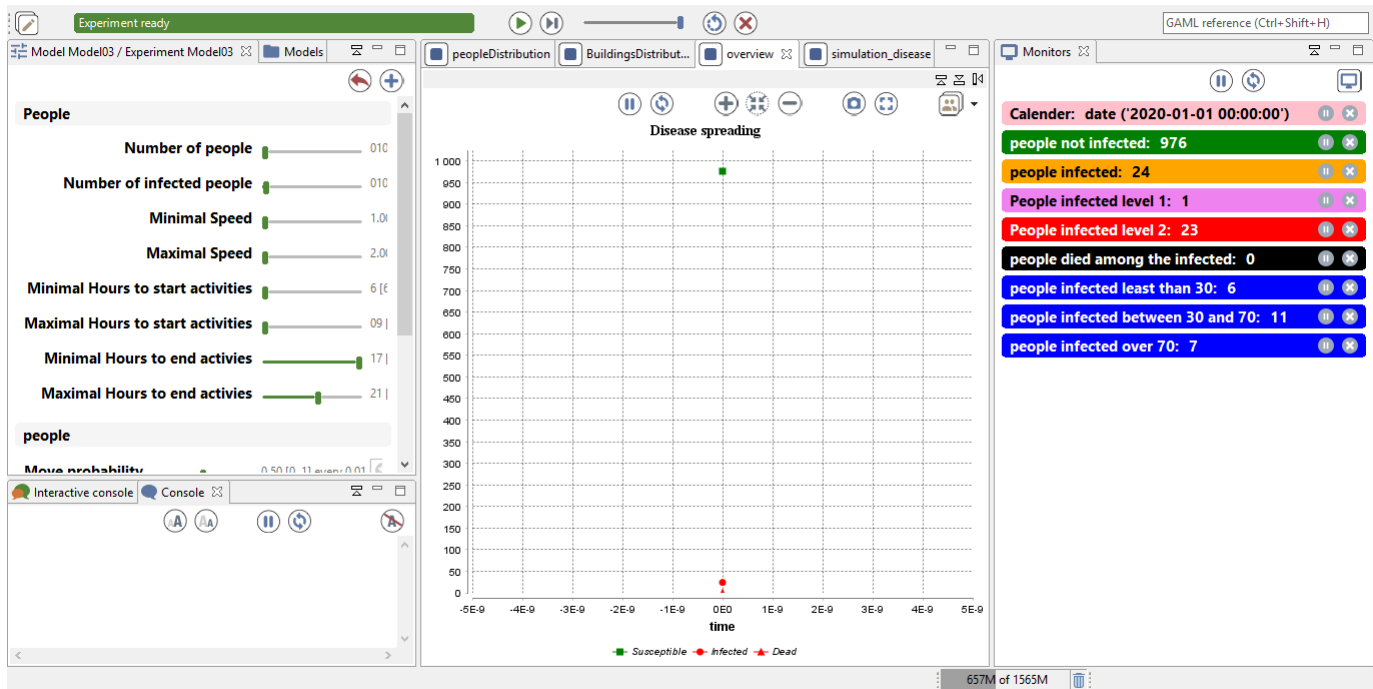


FIGURE 8 – étape 1 : initialisation scénarios avec faible distance sociétale

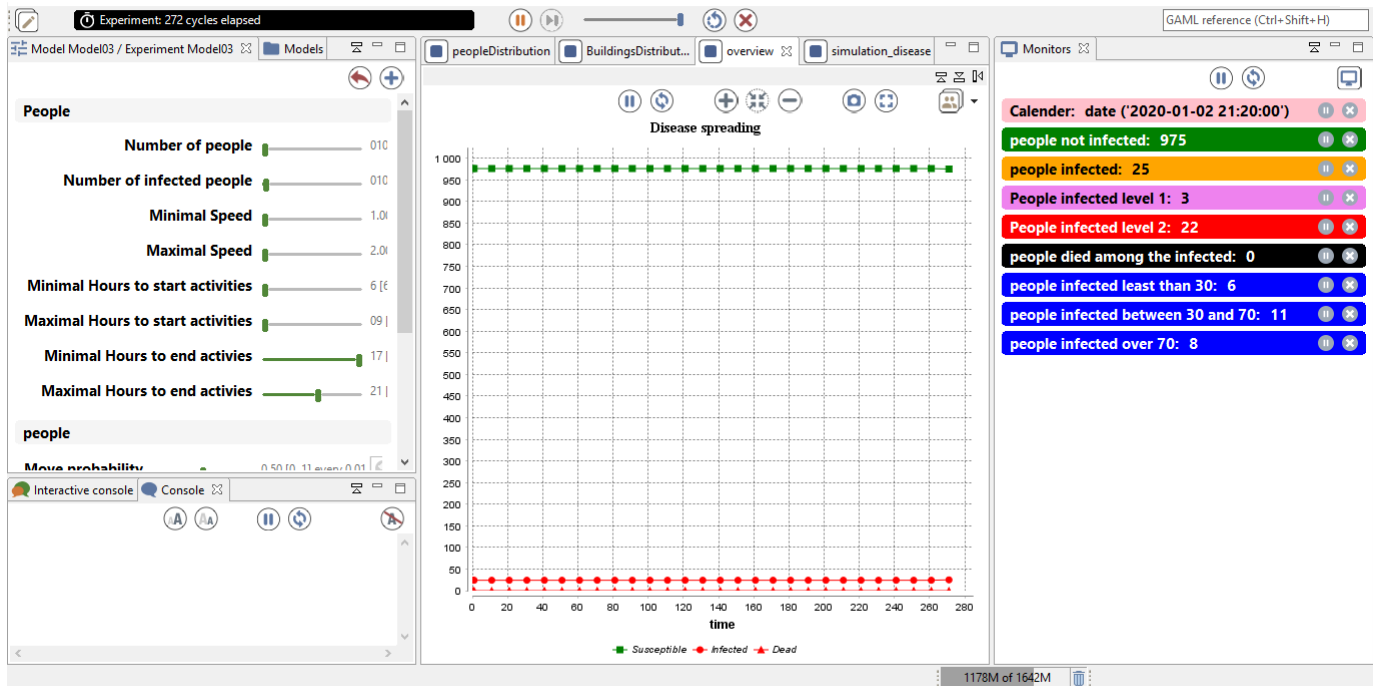


FIGURE 9 – étape 2 : après une certaine durée scénarios avec faible distance sociétale

4 Conclusion

En guise de conclusion , nous retiendrons que la meilleur façon de lutter contre une épidème, est le confinement. Cela de lutter plus efficacement avec résultats immédiats. Alors une amélioration possible de notre travail serait de confiner toutes les personnes saines dans leur habitations et ne sortiront de nouveau que lorsque tous les cas positives passeront à négatif.