



**HAL**  
open science

## Chapter 3: Macro Models, Micro Models and Network Based coupling

Arnaud Banos, Nathalie Corson, Éric Daudé, Benoit Gaudou, Sébastien Rey-Coyrehourcq

► **To cite this version:**

Arnaud Banos, Nathalie Corson, Éric Daudé, Benoit Gaudou, Sébastien Rey-Coyrehourcq. Chapter 3: Macro Models, Micro Models and Network Based coupling. Arnaud Banos; Christophe Lang; Nicolas Marilleau. Agent-based Spatial Simulation with Netlogo, Volume 2: Advanced Concepts, 2, Iste Press, pp.63-84, 2017, 978-1785481574. 10.1016/B978-1-78548-157-4.50003-0 . hal-01829756

**HAL Id: hal-01829756**

**<https://hal.science/hal-01829756>**

Submitted on 23 Mar 2021

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

---

## Macro Models, Micro Models and Network-based Coupling

---

### 3.1. Introduction

In this chapter, we will discuss coupling models with different scales to describe the propagation of a virus within a population. This population is distributed throughout a set of cities connected by airline routes. Population movements between cities enable the virus to travel, carried by infected individuals. In each city, the description of virus propagation is based on an SIR-type model (presented in more detail in section 3.2). The first model, called EpiSim [DAU 14], is presented in section 3.3, first in its aggregate variant, and then in its individual-based variant. Comparing these two approaches will allow us to discuss the advantages and limitations of each of them. In the second section, we will present two approaches for coupling models based on networks. The first approach considers a network of coupled systems of equations (section 3.5). The second combines an agent-based model of microscale components with a model of macroscale components based on systems of equations (section 3.6). This coupled model, called MicMac, is presented in more detail in [BAN 16]. The added value of micro/macro coupling is examined in the article [BAN 15b].

This chapter picks up where the last chapter left off. In the previous chapter, we considered a weak coupling approach combining distinct traffic models

with different scales. This chapter instead focuses on an integrative coupling approach that associates two different dynamics modeled at different scales.

### 3.2. Description of the equation-based SIR model

The dynamics of an epidemic within a population can be formulated with an SIR model that describes the evolution of the number of susceptible ( $S$ ), infected ( $I$ ) and recovered ( $R$ ) individuals within the population. This model is described by the system [3.1] (see [KER 27]).

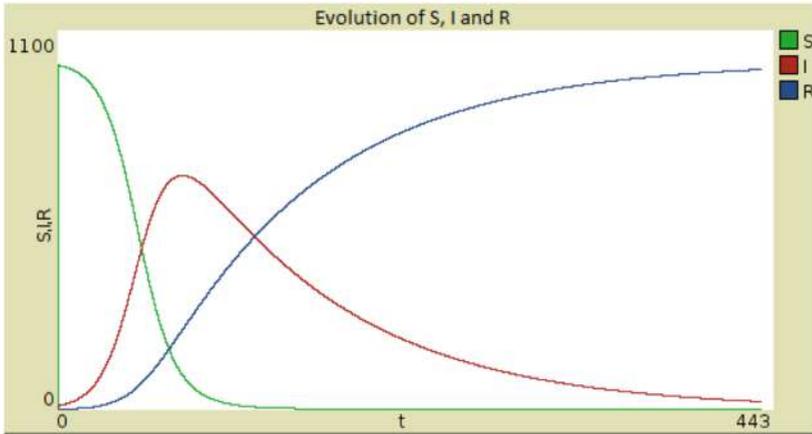
$$\begin{cases} \frac{dS}{dt} = -\frac{\beta}{N}IS \\ \frac{dI}{dt} = \frac{\beta}{N}IS - \alpha I \\ \frac{dR}{dt} = \alpha I \end{cases} \quad [3.1]$$

In this model, each city has a population ( $P$ ) divided into three groups  $S$ ,  $I$  and  $R$ . If the population has not been previously exposed to the virus, then  $P = S$ , and everybody in the city is susceptible. The population is constant, i.e. demography is not taken into account. The transition from the group of susceptible individuals to the group of infected individuals is described by the term  $\frac{\beta}{N}SI$  where  $\frac{\beta}{N}$  is the contamination rate in the event of contact between a susceptible individual and an infected individual. The transition from the group of infected individuals to the group of immune (recovered) individuals is described by  $\alpha I$ , where the  $\alpha$  term gives the proportion of infected individuals that recover at each time step.

This system of equations describes the evolution of the number of susceptible, infected and recovered individuals over time for fixed contamination and recovery rates, as shown in Figure 3.1.

There are various different approximation methods for numerically calculating the solutions of this kind of system, i.e. determining the evolution of  $S$ ,  $I$  and  $R$  over time. As discussed previously (Volume 1, Chapter 5, p. 178 [BAN 15a]), the system dynamics module in NetLogo uses Euler's method. We will use this module and hence this method for section 3.3 of this chapter. In section 3.3, the numerical method that we will use is called fourth-order

Runge–Kutta ([BAN 15a] p. 183). It is implemented directly in NetLogo. Finally, in section 3.6, we will use the Scala extension presented in Chapter 1 of this book.



**Figure 3.1.** Evolution of the number of susceptible, infected and recovered individuals over time with  $\alpha = 0.2$ ,  $\frac{\beta}{N} = 0.5$  (hence,  $R_0$  is equal to  $\frac{\beta}{\alpha} = 2.5$ ) and initial conditions  $S_{init} = 1000$ ,  $I_{init} = 10$  and  $R_{init} = 1$ . The chosen method of numerical integration is Runge–Kutta 4, with a resolution of  $10^{-3}$ . For a color version of the figure, see [www.iste.co.uk/banos/netlogo2.zip](http://www.iste.co.uk/banos/netlogo2.zip)

In the following sections, whenever an SIR system is used to describe the population of a city  $i$  (for  $i \in 1, \dots, NBNodes$ ), we will denote the S, I and R components of the populations respectively, by  $S_i$ ,  $I_i$  and  $R_i$ .

### 3.3. Equation-based and agent-based propagation model: EpiSim

In this section, we will aim to recreate the global dynamics of the aggregate SIR model with an agent-based model. We will analyze the conditions under which each of these two models converge and diverge. We will begin by individualizing the SIR model without spatial constraints, and then we will gradually add mechanisms for local interactions.

#### 3.3.1. Distributed and non-spatial SIR model

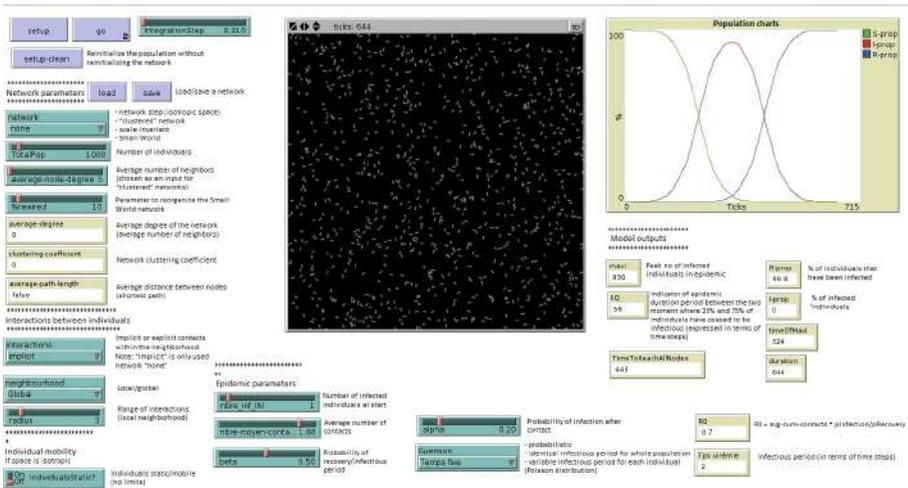
The first stage of modeling is to develop the processes of the aggregated mathematical model using an agent-based formalism

(EpiSim\_Model.e.nlogo), while retaining the fundamental assumptions of the initial approach (EpiSim\_Math\_Model.e.nlogo). To shift from the first formalism to the second, we need to redistribute the SIR model at the individual scale. Each agent is therefore assigned a description of its epidemiological state:  $S \rightarrow I \rightarrow R$ . Similarly to the aggregate version, these transitions are not symmetric.

In this first stage, space is not considered, and each agent can potentially interact with everyone, depending on the contact rate  $\beta$ . The graph of interactions is said to be trivial. Each agent has two transition functions:

$$P(S \rightarrow I) = \beta \frac{I}{N} \quad \text{and} \quad P(I \rightarrow R) = \alpha$$

This non-spatial model, shown in Figure 3.2, is agent-based, but still reproduces the same dynamics as the equation-based SIR model given equivalent initial conditions.



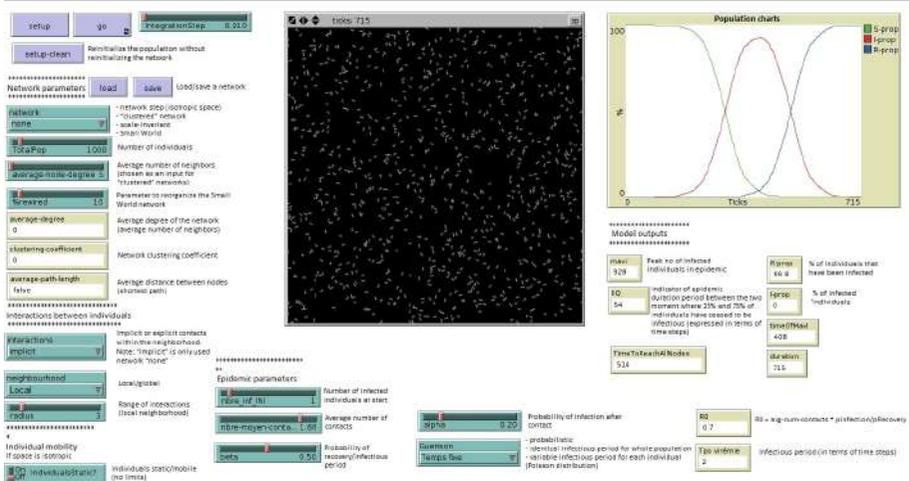
**Figure 3.2.** Screenshot of the dynamics of the non-spatial model. The Network parameter is set to “None”, the interactions are “implicit” (each agent can potentially interact will every other agent), the neighborhood is “global” and individual mobility (IndividualStatic?) is currently irrelevant

### 3.3.2. Spatially distributed SIR model with local interactions

In this second version of the model, the individual probabilities of transition are calculated as a function of a local fixed-radius neighborhood of agents (fixed Euclidean distance around the agent). At each time step, the agents have unlimited mobility within the domain. In this spatial version of the model, individuals are randomly distributed over space at initialization and at each time step. Contacts occur randomly within the neighborhood of each agent, which can be configured to be more global or local depending on the choice of spatial constraints. Thus, no *a priori* structure is defined for potential contacts. The graph of contacts that did actually occur during the simulation could, however, be reconstructed *a posteriori*, to examine its topology.

$$P(S \rightarrow I) = \beta \frac{I_{local}}{N_{local}} \quad \text{and} \quad P(I \rightarrow R) = \alpha$$

Under these conditions, the model shown in Figure 3.3 once again allows us to obtain dynamics similar to those produced by the equation-based SIR model.

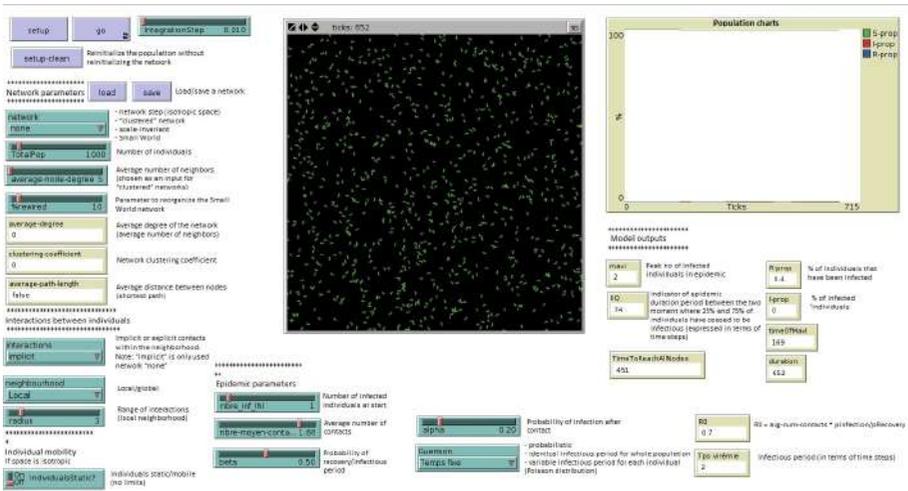


**Figure 3.3.** Screenshot of the dynamics of the spatial model with local neighborhoods. The Network parameter is still set to “None”, interactions remain “implicit”, the neighborhood is now “local” with a fixed radius of 3 and individual mobility IndividualStatic? is disabled

### 3.3.3. Spatially distributed model with local neighborhoods and explicit contact between individuals

We continue this approach with a fully distributed and behavioral model, shown in Figure 3.4. Each agent  $S$  experiences a number  $n$  of contacts (parameter avg-num-contact), which follows a Poisson distribution with mean  $\beta$ . Each agent therefore effectively enters into contact with  $n$  agents from its neighborhood  $V$ . If the state of one of these  $n$  neighbors is  $I$ , then  $S \rightarrow I$ .

$$P(S \rightarrow I) = 1 \quad \text{if} \quad \sum_V I \geq 1 \quad \text{and} \quad P(I \rightarrow R) = \alpha$$



**Figure 3.4.** Screenshot of the dynamics of the spatial model with local neighborhoods and interactions only between individuals. The Network parameter is still set to “None”, interactions are now “explicit”, the neighborhood remains “Local” with a fixed radius of 3 and individual mobility (IndividualsStatic?) is either enabled or disabled

The dynamics produced by the simulations of this model deviate from previous dynamics, and infection rates are lower than previous versions, or completely absent. One way of verifying this is to vary the radius, IndividualsStatic? mobility and avg-num-contact parameters to trigger the dynamics of an epidemic. These results allow us to improve the model by relaxing the hypotheses of global spatiality and global interaction. The next

model therefore takes into account the movements of agents and the localized nature of the contacts that could potentially result in virus transmission.

### ***3.3.4. Spatially distributed SIR model with a network of interactions***

With this next version, the structure of potential contacts can be entered as a model input to specify the network of interactions between agents. Each agent is represented by a node, and its potential contacts are represented by a set of connections making up its neighborhood. Hence, mobility is no longer relevant here, since the network of interactions does not change over the course of the simulation. At each time step, each agent comes into contact with other agents selected from the set of level 1 neighbors within the network, limited to the average number of contacts retained from the previous version of the model (fixed parameter  $\beta$  which follows a Poisson distribution). In other words, if the size of the neighborhood is less than `avg-num-contact`, then this parameter takes the value of the size of the neighborhood. Other networks can be constructed in order to study the role of topology in the dynamics of virus propagation:

- Regular networks: networks such that all vertices have the same number  $k$  of neighbors, also described as  $k$ -regular networks.

- Random networks with fixed degree distributions: networks whose vertices have an average of  $k$  neighbors. This number varies locally around the mean for each vertex. The number of edges is defined beforehand to guarantee the desired degree distribution, then each edge is connected to two randomly selected vertices.

- Small world networks: intermediate networks between regular networks and random networks. To generate these networks, we start with a regular network and randomly reassign a certain percentage of its edges.

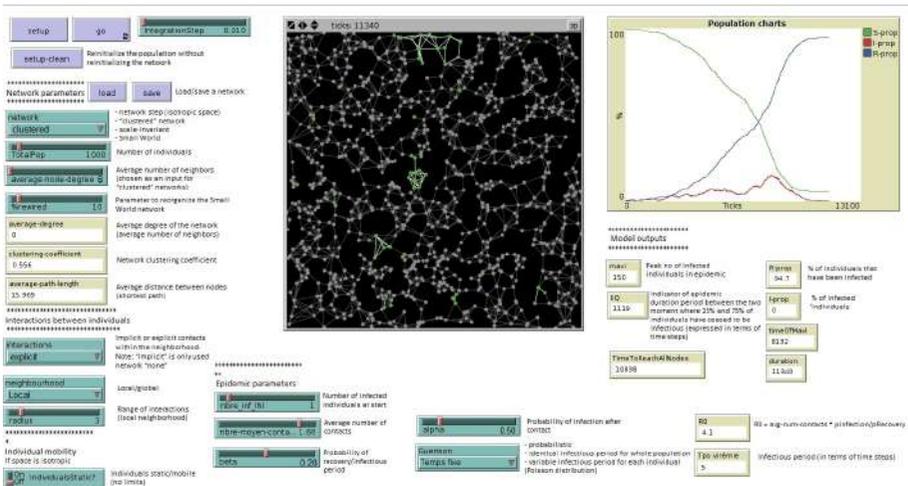
- Scale-free networks (scale-invariant): directed acyclic or star networks with a strongly heterogeneous distribution for the number  $k$  of neighbors of each node. This distribution follows a power law.

These networks can be characterized by global topological indicators such as:

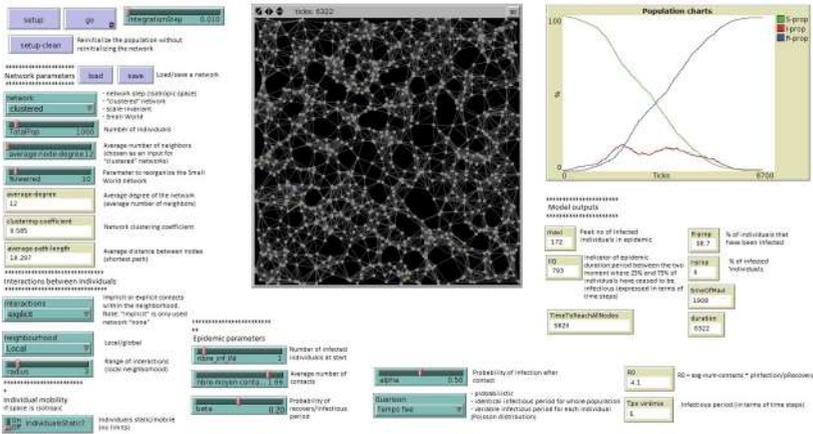
- the average degree  $K$ : the average number of incident edges to a vertex;

- the clustering coefficient  $C$ : this measures the degree to which the network contains clusters, i.e. groups of vertices strongly linked together and weakly connected to the rest of the network;
- the average of the  $C(x)$  evaluated at each vertex  $x$ : this is the number of existing edges within the neighborhood  $K(x)$  of the vertex  $x$  divided by the number of possible edges in this neighborhood;
- the average length of shortest paths: the average of the shortest values for the distance between any two vertices of the network.

We will present the results of simulations performed with a clustered network, such that  $k$ -degree equal to 6 (Figures 3.5 and 3.6) and 2 test parameter values corresponding to the high/low values for  $L$ . For example, when the network has a high  $k$ -degree, the clustering coefficient  $C$  is high, and the average of the shortest distances  $L$  tends to be low. By contrast, the lower the  $k$ -degree of the network, the higher the average of the shortest distances tends to be.



**Figure 3.5.** Screenshot of the dynamics of the spatial model, network with interindividual interactions. The value of *avg-num-contact* remains fixed at 1.66. The *Network* parameter is set to “clustered”, with *average-node-degree* equal to 6. The  $\alpha$  (0.50) and  $\beta$  (0.20) parameters have been adjusted to obtain a higher  $R_0$  (4.1)



**Figure 3.6.** Screenshot of the dynamics of the spatial model, network with interindividual interactions. The value of  $avg\_num\_contact$  remains fixed at 1.66. The *Network* parameter is set to “clustered”, with *average-node-degree* equal to 12. The  $\alpha$  and  $\beta$  parameters have been adjusted to obtain a higher  $R_0$  (4.1)

The simulation calculates several indicators as outputs: the proportion, of the entire population, of the contaminated individuals at peak contamination (MaxI); the interquartile range (IQR), which provides an indication of the duration of the epidemic; and the cumulative percentage of infected individuals once the epidemic is over (Recovered). In general, the simulations confirm the initial intuition that the more the network is connected, the faster the virus propagates through it, and the greater the number of infected nodes.

Studying the influence of the  $R_0$  parameter on these indicators and varying the  $k$ -degree parameter shows the “compensating” effect of this parameter relative to how the degree of network connectivity affects the propagation dynamics. Recall that the  $R_0$  parameter is equal to the probability of infection multiplied by the average number of contacts divided by the probability of recovery. The simulations show a strong time shift as a function of the  $k$ -degree parameter: the epidemic “takes off” more quickly as the network connectivity increases, for fixed  $R_0$ . This can be observed by comparing the two experiments shown in Figure 3.5 (11340 ticks) and Figure 3.6 (6322 ticks). A simple experiment (varying  $R_0$ , varying the  $k$ -degree parameter) allows this initial intuition to be quickly verified. Also, the  $R_0$  threshold value for a “lightning” epidemic (hyper-fast epidemic that reaches all vertices within a

short period) decreases as the connectivity increases. Note also that the value of  $R_0$  required for the epidemic to affect almost all of the population may be less than 1 if the network is strongly connected.

### 3.4. Coupling SIR models based on networks

Assume now that we have the same nodes as the previous models, representing cities, but that the edges between these cities represent connections by air. We also define a mobility rate  $g$  for each city, representing the proportion of outbound travelers from that city. This proportion may vary across the groups of susceptible, infected and recovered individuals. We could, for example, assume that in the case of a symptomatic disease, infected individuals will cease to travel, whereas asymptomatic individuals will continue to do so. With this assumption, the number of infected individuals is a function of the proportion of asymptomatic cases observed for a given infectious disease. The mobility rates are, respectively, denoted by  $g_i$ ,  $g_s$  and  $g_r$ . The mobile population of city  $i$  is therefore given by  $g(S_i + I_i + R_i)$ , or by  $(g_s S_i + g_i I_i + g_r R_i)$  in the case where the mobility varies as a function of infectious state.

Once these mobility rates have been defined, individuals from one city can travel to adjacent cities. The weight  $m_{ij}$  of each edge is the fraction of outbound travelers from node  $i$  headed toward node  $j$ . The sum of the  $m_{ij}$  corresponding to each of the outbound edges at a given node must be equal to 1:

$$\sum_{j=1, j \neq i}^N m_{ji} = 1 \text{ with } m_{ii} = 0$$

Thus, these nodes and edges define a network of cities, and the population moves through them. To model these movements, we propose two approaches.

### 3.5. SIR coupling without scaling: Metapop model

#### 3.5.1. Presentation of the Metapop model

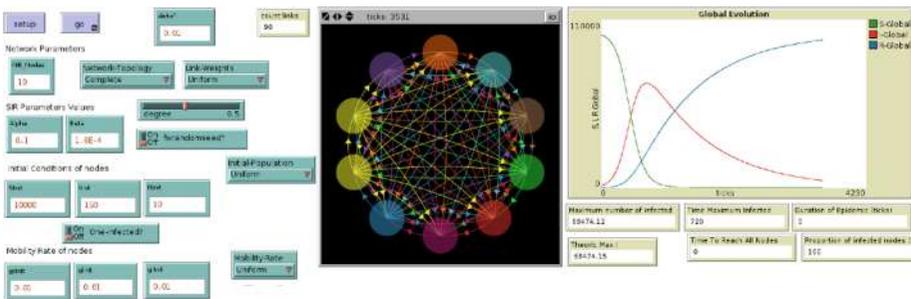
The metapopulation approach considered here takes into account the flow of travelers from one city to another at each time step of the simulation. Trips

are assumed to be instantaneous (the Euclidean distance between any two cities is zero) since the model does not naturally include a notion of time and its dynamics are exclusively defined by the number of integration steps. We are therefore only interested in the epidemic-related dynamics of the city network, as shown by the model in Figure 3.2.

$$\begin{cases} \frac{dS_i}{dt} = -\frac{\beta}{N}I_iS_i - g s_i S_i + \sum_{j=1}^n g s_j m_{ji} S_j \\ \frac{dI_i}{dt} = \frac{\beta}{N}I_iS_i - \alpha I_i - g i_i I_i + \sum_{j=1}^n g i_j m_{ji} I_j \\ \frac{dR_i}{dt} = \alpha I_i - g r_i R_i + \sum_{j=1}^n g r_j m_{ji} R_j \end{cases} \quad [3.2]$$

Note that at each node, these dynamics are given by the initial dynamics of the epidemic after subtracting all outbound travelers (hence the term  $-g s_i S_i$  in the group of susceptible individuals) and adding all inbound travelers from adjacent cities  $j$  (hence the term  $\sum_{j=1}^n g s_j m_{ji} S_j$  in the susceptible group). Movements are instantaneous, and at each step of the simulation the total population is given by the sum of the populations at each node.

The interface of the metapopulation model is presented in Figure 3.7.



**Figure 3.7.** Interface of the metapopulation model, allowing the user to define the network topology as well as the mobility rates and the initial distribution of the population. This allows us to observe not only the dynamics of single nodes, but also those unfolding at the level of the whole network. For a color version of the figure, see [www.iste.co.uk/banos/netlogo2.zip](http://www.iste.co.uk/banos/netlogo2.zip)

### 3.5.2. Numerical integration of a network of coupled SIR using Runge-Kutta implemented in NetLogo

The fourth-order Runge-Kutta method is used to numerically approximate the solutions of systems of ODEs. Recall that it works by averaging four estimates, which makes it more precise than other approaches such as the Euler method. Its implementation in NetLogo is shown below for an SIR system: in the following code, the values of S, I and R are `currentS`, `currentI` and `currentR`. Multiple stages are required to calculate these values for the next time step, which are specified in full detail in Chapter 5 of Volume 1 [BAN 15a].

---

```
to calcK [coef]
2  set tmpS currentS + coef * currentKS
   set tmpI currentI + coef * currentKI
   set tmpR currentR + coef * currentKR
   set nextKS (- beta) * tmpI * tmpS
   set nextKI beta * tmpI * tmpS - alpha * tmpI
7  set nextKR alpha * tmpI
   end

to stepK [coef]
   set nextS nextS + coef * nextKS
12  set nextI nextI + coef * nextKI
   set nextR nextR + coef * nextKR
   set currentKS nextKS
   set currentKI nextKI
   set currentKR nextKR
17 end

;; deltaT is the integration step

to RKstepNetwork
22  (foreach (list 0 (1 / 2) (1 / 2) 1) (list (1 / 6) (1 / 3) (1 / 3) (1 /
    6)) [
    calcK ?1 * deltaT
    stepK ?2 * deltaT
  ])

27  ;; update of the values of currentS, currentI and currentR
   set currentS nextS
   set currentI nextI
   set currentR nextR
end
```

---

This same method can be applied to the Metapop model with coupled SIR. We must simply account for inflow and outflow at each of the nodes.

---

```

to calcK [coef]
  set tmpS currentS + coef * currentKS
  set tmpI currentI + coef * currentKI
4   set tmpR currentR + coef * currentKR
  set nextKS (- beta / (tmpS + tmpI + tmpR)) * tmpI * tmpS
  set nextKI (beta / (tmpS + tmpI + tmpR)) * tmpI * tmpS - alpha * tmpI
  set nextKR alpha * tmpI
end
9

to coupling

; Outflow = sum of mij for each out-link of the nodes
14 ; this outflow is multiplied by the population of the nodes later

let sumOutS 0
let sumOutI 0
let sumOutR 0
19
  ask my-out-links
  [
    set sumOutS sumOutS + mij
    set sumOutI sumOutI + mij
24   set sumOutR sumOutR + mij
  ]

; Inflow : sum of mij * gs * tmpS, mij * gi * tmpI, mij * gr * tmpR for
      each in-link
; where tmpS, tmpI and tmpR are the number of S, I, R of the source node
      during the RK4 procedure
29
let sumInS 0
let sumInI 0
let sumInR 0

34 ask in-link-neighbors
  [
    let mi [Mij] of out-link-to myself
    set sumInS sumInS + gs * tmpS * mi
    set sumInI sumInI + gi * tmpI * mi
39   set sumInR sumInR + gr * tmpR * mi
  ]

; Update of the number of S, I and R in each node, taking into account
      outflows and inflows

44 set nextKS nextKS - sumOutS * gs * tmpS + sumInS
  set nextKI nextKI - sumOutI * gi * tmpI + sumInI
  set nextKR nextKR - sumOutR * gr * tmpR + sumInR

```

```
end

49 to stepK [coef]
    set nextS nextS + coef * nextKS
    set nextI nextI + coef * nextKI
    set nextR nextR + coef * nextKR
    set currentKS nextKS
54  set currentKI nextKI
    set currentKR nextKR
end

to RKstepNetwork
59  (foreach (list 0 (1 / 2) (1 / 2) 1) (list (1 / 6) (1 / 3) (1 / 3) (1 / 6))
    [
        ask nodes [calcK ?1 * deltaT]
        ask nodes [coupling]
        ask nodes [stepK ?2 * deltaT]
64  ])
    ask nodes
    [
        set currentS nextS
        set currentI nextI
69  set currentR nextR
    ]
end
```

---

### 3.5.3. Examples of results

[BAN 15b] presents the results of both models (Metapop and MicMac). We chose the following indicators to analyze these results:

- *MaxI*: the maximum number of infected individuals at any given moment;
- *TimeofMaxI*: the moment at which the maximum number of infected individuals occurred;
- *Duration*: the duration of the epidemic.

This choice of indicators allows us to characterize how the propagation of the disease though the population changes.

The first interesting result is that if we assume that the network is complete, the *S*, *I* and *R* populations and the mobility rates are uniformly distributed over all nodes, and the weights are equal on all edges of the network, then the MetaPop model is equivalent to an SIR system on the total population.

[BAN 15b] also shows the impact of different topologies on the disease spread. In summary, decreasing the diameter and the average length of paths in a network increases the value of  $MaxI$  and decreases the values of  $TimeofMaxI$  and  $Duration$ . This is characteristic of an increase in the rate of diffusion of a disease within a population.

### 3.6. SIR coupling with scaling: MicMac model

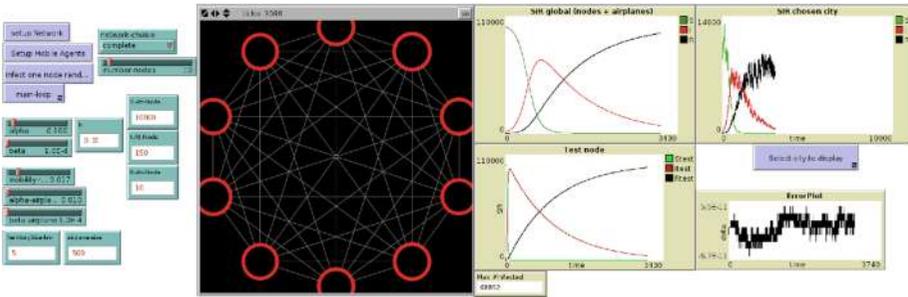
The MetaPop model assumes that the population moves instantaneously between cities. We can include this hypothesis in the MicMac model by introducing a new type of agent (typically representing aircraft) that transports individuals between cities. This model therefore introduces a change in scale and a new paradigm, since the epidemiological dynamics of city populations are described by equations, and the epidemiological dynamics between cities are modeled by single agents and their movements.

#### 3.6.1. Model presentation

The MicMac model uses the same city agents as above, with the same equation-based epidemiological dynamics (also described by the SIR model). The difference between the 2 models lies in the “mobility” component of the model, which is disaggregated and discrete: individuals are extracted from each city and travel to other cities by airplane. The flight duration depends on the distance, and is adjusted to the integration step. A preliminary calibration phase is performed for each simulation to synchronize the integration step, flight durations and distances, based on the duration of an observed epidemic. The same principle of conservation of population is satisfied: the total population is constant and at any given moment is equal to the sum of the populations in the nodes and in the airplanes.

The interface of this hybrid model is presented in Figure 3.8.

Whereas the Metapop model had instantaneous movements, in the hybrid model the dynamics of the epidemic need to be defined. Indeed, the disease continues to propagate during flights, inside the airplanes. Thus, at each time step, we need to update the number of individuals both in the cities and in the air. For now, we will use an ODE-based SIR model to describe contagion in the air.



**Figure 3.8.** Interface of the hybrid micro-macro model, allowing the user to define the network topology, mobility rates, and initial population distribution. We can observe the dynamics at each node, but also the dynamics of the whole network. At each time step, the difference between the total population in the network and the initial population is calculated to ensure that the total population remains constant while passing from the population to individuals and vice versa

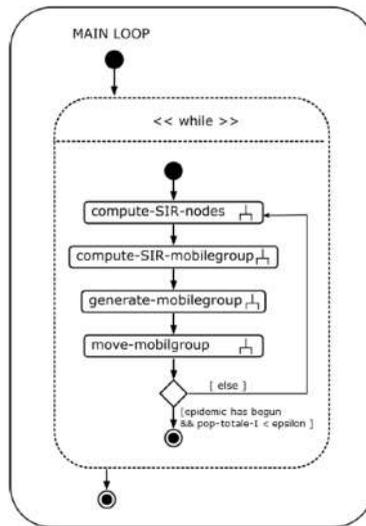
### 3.6.2. General description of the working principle of the coupling

Figure 3.9 gives a global overview of the dynamics of the MicMac model in the form of an activity diagram. The simulation loop is divided into 4 main stages, each of which has its own method for coupling the equation-based model with the agent-based model.

Firstly, the infectious state of the population in cities (nodes agents) and aircraft (mobilegroup agents) is updated. This dynamic is described by the system of SIR equations. Each step of the simulation corresponds to one integration step of the SIR system: consequently, the numerical solution of the system and the updates of the agent-based model are synchronized. The synchronization begins at model initialization by performing a calibration phase on a test node. The numerical solution is presented in the next section.

Next, new airplanes are created. Each node is assigned a continuous stock of population representing the proportion that desires to leave. Named stock-to-flight, this stock depends in particular on a mobility rate representing the fraction of the total population that can travel by plane at each time step. The stock is incremented at each simulation step. The algorithm creates planes according to the following principle: as many full planes as possible are created at each step of the simulation and each node. Thus, at each simulation step, if the value of stock-to-flight exceeds the capacity

of a plane, a plane agent is created with a destination city chosen randomly from the neighbors of the current node, containing a number of passengers equal to its capacity. The population of the aircraft is then extracted from the city of departure. The number of individuals in each state is proportional to the distribution within the city, and is calculated using a lottery algorithm specified in section 3.6.5.2. The number of people in the plane is then subtracted from the `stock-to-flight` of that node. While this value remains greater than the capacity of one plane, another plane is created by following the steps given above.



**Figure 3.9.** General activity diagram of the MicMac model

At each step of the simulation, the airplanes travel at a certain speed (and therefore cover a certain distance). This speed is also calibrated at initialization. Once a plane reaches its destination city, its population merges with the global population of the city (see section 3.6.5), and disappears. Thus, if the plane contained infected persons, they could potentially contaminate the population in the destination city, allowing the virus to travel from city to city.

### 3.6.3. Initialization: calibrating the model

At the beginning of the simulation, an SIR model is integrated on a reference node using the parameter values specified by the user. This node

contains the total population of the model, and uses the same stopping condition for calibration as the simulation (namely that proportion of infected individuals is below a certain threshold  $\epsilon$ ). The number of iterations of the RK4 method required to achieve this stopping condition is calculated, and the relation between the duration of the epidemic specified by the user and the number of RK4 iterations is calculated to determine the “duration” of each time step. Since the aspect of space is incorporated into the model by specifying the network structure, this preliminary operation allows the size of each edge to be derived, but also determines the traveling speed and therefore the transport time.

### ***3.6.4. Using the RK4 extension to perform numerical integration***

The SIR equations of this model are numerically integrated using the RK4 extension (see section 1.2.8 of this book), which is recalled directly in the code as follows:

---

```
set sir rungeKuta:compute-SIR S_Node I_Node R_Node galpha (gbeta / (S_Node  
+ I_Node + R_Node)) integrationStep
```

---

### ***3.6.5. Switching between the continuous and discrete parts of the model***

The MicMac model associates equation-based dynamics and agent-based dynamics. For the former, integrating the system of equations can produce non-integer values for each population stock. For the latter, by definition, the dynamics are expressed in terms of integer numbers of individuals. The interface between these dynamics therefore requires a method for transitioning between continuous and discrete settings.

#### ***3.6.5.1. Transition from discrete to continuous***

Converting discrete values to continuous values is trivial. Each plane contains an integer number of susceptible, infected and recovered persons. Once it arrives at a city, each plane unloads its passengers, which are added to the city stock for each state.

### 3.6.5.2. Transition from continuous to discrete: the lottery algorithm

The reverse situation arises when a plane is created to travel from one city to another, specifically when calculating its integer number of passengers. At this point, we encounter the following problem: how do we obtain integer numbers of susceptible, infected and recovered populations from three population stocks, represented as continuous values, while conserving the proportions of each stock?

To answer this question, we introduce a so-called “lottery” algorithm. We will present this algorithm here, as it is sufficiently general to be useful for a wide range of different problems. It is based on two subroutines: `find-state` and `generate-passengers`.

Given a set of (integer) values representing the number of individuals in each state, `find-state` randomly chooses a state with probability proportional to the number of individuals in this state. The function returns an integer value representing the selected state, corresponding to the index of the state selected from the list passed as a parameter.

---

```

to-report find-state [#roundedStock]

  let roundedPop sum #roundedStock
4   let random-value (random roundedPop) + 1

  let state 0
  let step-i 0

9   if roundedPop > 0 [
    foreach #roundedStock
      [
        if state = 0 [
          set random-value random-value - item step-i #roundedStock
14          ifelse random-value <= 0 [
            set state step-i + 1
          ][
            set step-i step-i + 1
          ]
        ]
      ]
19  ]
  ]

  report state
24 end

```

---

The function `generate-passengers` takes the list of the number of individuals in each possible state (`pop`) and the number of individuals in the group to be generated (`sample_number`) as parameters. It generates and returns a group of `sample_number` individuals, with the same proportions in each state as `pop`. This group has an integer number of individuals in each state. The returned population is then removed from the population passed as a parameter. This function takes two preliminary precautions:

- the population passed as a parameter is rounded down to the integer below for each state. This means that if one state has the value of 0.3, it will be rounded down to 0.

- it checks that the population passed as a parameter is larger than the number of agents in the expected output population, after rounding.

The `find-state` function is then called `int(sample_number)` times. Each time that it is called, one individual is extracted from the initial population and added to the population that will be returned. Consequently (because of the `find-state` algorithm), if the number of individuals (in the population passed as a parameter) in a certain state hits 0 at some point during the execution of the algorithm, no further individuals will be drawn in this state.

---

```
1 to-report generate-passengers [pop sample_number]

  let rounded_pop recompute-rounded-population pop
  let S_pop item 0 rounded_pop
  let I_pop item 1 rounded_pop
6  let R_pop item 2 rounded_pop

  let state 0
  let Si 0
  let Ii 0
11 let Ri 0

  if ((sum rounded_pop) >= int(sample_number))
  [
    ;; One returned state by find-state
16  repeat sample_number
    [
      ;; compute/recompute population at each turn
      set state find-state recompute-rounded-population (list S_pop
        I_pop R_pop)
21  if state = 1 [
```

```

        set Si Si + 1
        set S_pop S_pop - 1
    ]
26     if state = 2 [
        set Ii Ii + 1
        set I_pop I_pop - 1
    ]
31     if state = 3 [
        set Ri Ri + 1
        set R_pop R_pop - 1
    ]
    ]
36 ]

    report (list Si Ii Ri)
end

```

---

```

to-report recompute-rounded-population [pop]
  report (list int(item 0 pop) int(item 1 pop) int(item 2 pop))
end

```

---

### 3.6.6. Example results

As above for the MetaPop model, the results are taken from [BAN 15b]. We will also consider the same indicators as before. Similarly to the MetaPop model, the MicMac model can be equivalently rephrased as a system of SIR ODEs on the total population if we consider the special case of a complete network and uniformly distributed population over each node. The MicMac model also requires the assumption of instantaneous travel between cities (or zero distance between cities).

If we do not assume instantaneous travel or uniform distribution, the model no longer coincides with classical SIR nor MetaPop:

- $MaxI$  of MetaPop  $>$   $MaxI$  of MicMac;
- $TimeofMaxI$  of MetaPop  $<$   $TimeofMaxI$  of MicMac;
- $Duration$  of MetaPop  $<$   $Duration$  of MicMac;

These results show that diffusion unfolds more slowly in the MicMac model. This is in particular because a city with an infected individual will not necessarily infect its neighbors (whereas in the MetaPop model this city would always send at least a small proportion of infected individuals to its neighbors once it becomes infected).

The effect of the network topology on the dynamics of the epidemic, on the other hand, is essentially identical in both MicMac and MetaPop. The difference lies in the previous comment: propagation will be slower in the MicMac model because an infected node does not always infect its neighbors.

### 3.7. Conclusion and outlook

In a massively connected and highly mobile world, studying the diffusion of epidemics is of great social and scientific value. Modeling not just the spatial but also temporal character of propagation leads us to consider hybrid models. In this chapter, we presented a series of different models. The first of these, the equation-based SIR model, is a macroscopic model based on ordinary differential equations. We used the assumptions of this model as the basis for a second, agent-based model that allows us to reproduce similar behavior under certain conditions (EpiSim). Two approaches were considered to describe diffusion within a network of interconnected cities. The first approach, fully macroscopic, describes the population as a homogeneous group. Flows from one city to another are instantaneous, which excludes temporality or state changes ( $S$ ,  $I$ ,  $R$ ) during travel. The second, hybrid approach views cities as homogeneous groups of individuals but allows for heterogeneity in their movements. It also allows us to include a description of travel times and distances. Control strategies (quarantine, avoidance, risk culture, etc.) can be applied to both of these models (MetaPop and MicMac). All strategies applied to cities at a global scale can be tested with either of these models. However, strategies related to individual choices can only be tested with the MicMac model.