Documentation for the Toolbox to Simulate and Mitigate COVID-19 Propagation

Z.E.O FRIHI, J. BARREIRO-GOMEZ, S. E. CHOUTRI and AND H. TEMBINE

Learning & Game Theory Laboratory, New York University Abu Dhabi

Research Center on Stability, Instability and Turbulence, New York University Abu Dhabi

1. Description of the functions

The MatLab-based toolbox is mainly composed of six functions, which are explained in detailed next.

1.1. Construction of the continents data

The function `construct_continents` is in charge of generating the required data for each one of the regions in the world map. There are six main regions as follows:

- North America
- South America
- Africa
- Europe
- Asia
- Australia

1.1.1. Inputs

The function `construct_continents` `(I,resolution_grid)` has two main inputs as follows:

- `I`: this input corresponds to the image *continents.png* that is attached in the toolbox. This picture classifies the different regions by means of different colors.

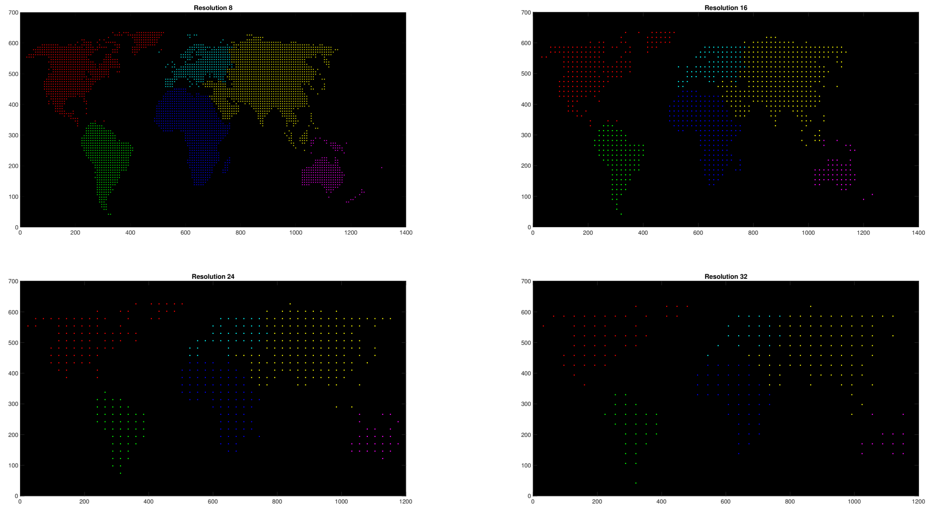

Figure 1: Different resolutions selected in the function `resolution_grid`.

- `resolution_grid`: this input determines the resolution of the discretization of the map, i.e., it is associated to the number of nodes considered in the map. Figure 1 shows the world map for four different resolutions.

1.1.2. Outputs

The function `[Datos_NorthAmerica,...,Datos_Oceania] =construct_continents` returns six outputs, each one of them corresponding to a region in the map. These outputs are the following:

- `Datos_NorthAmerica`
- `Datos_SouthAmerica`
- `Datos_Africa`
- `Datos_Europe`
- `Datos_Asia`
- `Datos_Oceania`.

Each output is a cluster composed of three components as follows:

- `Datos_NorthAmerica{1}`: this element returns the total number of nodes corresponding to the region North America. Similarly, the elements `Datos_SouthAmerica{1}`...
  `Datos_Oceania{1}` return the number of nodes in the continents South America ... Australia, respectively. Notice that this number directly depends on the resolution selected in `resolution_grid`.

- `Datos_NorthAmerica{2}`: this element of the cluster returns a matrix of the x-y coordinates for each one of the nodes introduced in
  `Datos_NorthAmerica{1}` for the region North America. Similarly, coordinates for the nodes of other regions are obtained from
  `Datos_SouthAmerica{2}`...
  `Datos_Oceania{2}`

- `Datos_NorthAmerica{3}`: this element of the cluster returns the corresponding filtered region from the image `I` (see Figure 2). The filtered regions South America ... Australia can be obtained using the functions `Datos_SouthAmerica{3}` ...
  `Datos_Oceania{3}`, respectively.

1.2. Publication of the sets $\mathcal{S}$, $\mathcal{I}$, $\mathcal{R}$, and $\mathcal{D}$ for the regions

The function `show_map` plots the discretized map by using the nodes constructed in the function
`construct_continents`.

1.2.1. Inputs

The fuction `show_map(S,I,R,D,C,color)` has six inputs as follows:

- `S`: this input corresponds to the set of susceptible nodes for any selected region, e.g., `S_NorthAmerica`. This nodes are plotted with a desired color selected by the user and introduced in the las input of this function `color`

- `I`: this input corresponds to the set of infected nodes for any selected region, e.g., `I_NorthAmerica`. These nodes are always plotted with orange color.

- `R`: this input corresponds to the set of recovered nodes for any selected region, e.g., `R_NorthAmerica`. These nodes are always plotted with yellow color.

- `D`: this input corresponds to the set of dead nodes for any selected region, e.g., `D_NorthAmerica`. These nodes are always plotted with red color.

- `C`: this input corresponds to the coordinates of the nodes for the whole world map, e.g., `C= [Datos_NorthAmerica{2}, ..., Datos_Oceania{2}`

- `color`: this input determines the color for the susceptible nodes, e.g., `[0,0,1]` plots the nodes with blue color.

1.2.2. Outputs

The function `show_map` does not have any numerical output. Instead, this function generates the plot of the map with the characteristic selected in the inputs. Figure 5 shows an example.

1.3. Computation of a neighborhood set around a node

This function allows computing a set around a certain node. This function is fundamental in order to compute the set of susceptible nodes around a sick node to propagate the virus.

1.3.1. Inputs

The function `neighbor(agent,radius,xpos,ypos)` has four inputs as follows:

- `agent`: this input is given by any node from the discretization of the map. This node can belong to any of the six regions.

- `radius`: this scalar value allows determining the size of the neighborhood. This parameter allows configuring either the local or long-distance propagation of the virus.

- `xpos`: this entry corresponds to the x coordinates for the whole world map, i.e., `xpos=C(:,1)`

- `ypos`: this entry corresponds to the y coordinates for the whole world map, i.e., `xpos=C(:,2)`

1.3.2. Outputs

The function `N = neighbor` has a unique output. This output `N` is a set of nodes corresponding to the neighborhood set of `agent`. See Figure 5.

1.4. Generation of the age distribution

The function `show_age` publishes the world map showing the distribution of three age ranges defined by `age1`, `age2`, and `age3`.

1.4.1. Inputs

The function `show_age(C,age1,age2,age3)` has four inputs described as follows:

- `C`: this input corresponds to the matrix of coordinates for all the nodes in the world map as in the function
`show_map(S,I,R,D,C,color)`,
i.e., `C= [Datos_NorthAmerica{2}, ...,`
`Datos_Oceania{2}`
- `age1`: set of nodes with age in the range $[0, 39]$
- `age2`: set of nodes with age in the range $[40, 69]$
- `age1`: set of nodes with age greater than 70

1.4.2. Outputs

The function `show_age` does not have any numerical output. Instead, this function generates the plot of the map with the age distribution for the three age ranges. Figure 6 shows the distribution of ages for the six regions and Figure 7 shows the distribution of the ages in the map for the whole world.

1.5. Computation of the flight connections

The function `flight_connections` computes the set of nodes belonging to the regions to which a node can travel. This function allows computing the long-distance propagation of the virus by means of flights.

1.5.1. Inputs

The function `flight_connections`
`(A,S_NorthAmerica,...,S_Oceania)` has seven inputs as follows:

- `A`: this input is an adjacency matrix $A \in \{0, 1\}^{6\times 6}$ describing the possible intercontinental transmision among the different regions. For example, Figure 8 shows a multi-region connectivity given by the following adjacency matrix

$$A = \begin{pmatrix} 0 & 1 & 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 1 & 0 \end{pmatrix}$$

and Figure 8 shows an scenario with $A = \mathbf{0}$, i.e., without intercontinental connections.

- `S`: this input corresponds to the set of nodes for each region, i.e., `S_NorthAmerica,...,S_Oceania`

1.5.2. Otputs

The function
`[fc_NorthAmerica,...,fc_Oceania]`
`=flight_connections` has six outputs corresponding to each region as follows:

- `fc_NorthAmerica`: this output is the set of nodes to which an agent belonging to the region North America can move to. In other words, a sick agent belonging to the region North America could infect an agent belonging to the set `fc_NorthAmerica`. This set is determined by using the adjacency matrix A.

- `fc_Oceania`: the interpretation of this set if the same as for the set `fc_NorthAmerica` but for the continent Australia.

1.6. Computation of the virus propagation

The function `update` is in charge of updating all the sets $\mathcal{S}$, $\mathcal{I}$, $\mathcal{R}$, and $\mathcal{D}$ for all the regions.

1.6.1. Inputs

The function
`update(S,I,R,D,xpos,ypos,`
`flight_connections,p,Age)` has nine inputs:

- `S`: this input corresponds to the set of susceptible nodes for any selected region, e.g., `S_NorthAmerica`.

- `I`: this input corresponds to the set of infected nodes for any selected region, e.g., `I_NorthAmerica`.

- `R`: this input corresponds to the set of recovered nodes for any selected region, e.g., `R_NorthAmerica`.

- `D`: this input corresponds to the set of dead nodes for any selected region, e.g., `D_NorthAmerica`.

- `xpos`: this entry corresponds to the x coordinates for the whole world map, i.e., `xpos=C(:,1)`

- ypos: this entry corresponds to the y coordinates for the whole world map, i.e., xpos=C(:,2)

- flight_connections: this is the set of agents that can be infected from other regions because of flights and connectivity of continents, e.g., fc_NorthAmerica

- p: this input is the set of transition probabilities p = [p1,p2,p31,p32,p33,p4,p5] in the Markov chain as follows:

  - p1: probability that a susceptible agent gets infected
  - p2: probability that the sick agent infects an agent from other regions (because of flight connectivity)
  - p31: probability of death for a sick agent (age-dependent) [0-39]
  - p32: probability of death for a sick agent (age-dependent) [40-69]
  - p33: probability of death for a sick agent (age-dependent) [70+]
  - p4: probability that a sick agent gets recovered
  - p5: probability that a recovered agent gets sick again

- Age: this is the vector of ages for each region

1.6.2. Outputs

The function
[S_dif,I_dif,R_dif,D_dif,S,I,R,D,FSA,Flag] = update has ten outputs as follows:

- S_dif: this output corresponds to the changes that the set of susceptible nodes had

- I_dif: this output corresponds to the changes that the set of infected nodes had

- R_dif: this output corresponds to the changes that the set of recovered nodes had

- D_dif: this output corresponds to the changes that the set of dead nodes had

- S: this output corresponds to the updated set of susceptible nodes for any selected region, e.g., S_NorthAmerica.

- `I`: this output corresponds to the updated set of infected nodes for any selected region, e.g., `I_NorthAmerica`.
- `R`: this output corresponds to the updated set of recovered nodes for any selected region, e.g., `R_NorthAmerica`.
- `D`: this output corresponds to the updated set of dead nodes for any selected region, e.g., `D_NorthAmerica`.
- `FSA`: this is a unique agent from the set `flight_connections`, for example `fc_NorthAmerica`, that is in risk of getting infected
- `Flag`: this value indicates if `FSA` got infected, i.e., if `Flag=1` then `FSA` got infected

2. Illustrative Examples

Once the main functions of the toolbox have been introduced, we present different examples in order to illustrate how they can be effectively used. The proposed examples show the usage of the functions progressively one by one until the last example presents a simulation of the propagation of the virus in the world divided into six different continents.

2.1. Example 1

The main objective of this example is to present the construction of the continents with different resolutions and computation of X-Y coordinates.

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%% Authors: J. Barreiro—Gomez, S. E. Choutri, and H. Tembine
%%%% Learning & Game Theory Laboratory (L&G Lab)
%%%% Center on Stability, Instability and Turbulence (SITE)
%%%% NYUAD 2020
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
clear all
close all
clc
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
I = imread('continents.png') ; % The reference image is ...
    imported
resolution_grid = 8 ; % This defines the resolution
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
[Datos_NorthAmerica,Datos_SouthAmerica,...
Datos_Africa,Datos_Europe,Datos_Asia,...
Datos_Oceania]...
= construct_continents(I,resolution_grid) ; % Data per ...
    continent are generated
%%%% Datos are organized as follows: (1) Number of nodes ...
    in the continent,
%%%% (2) Coordinates vector, and (3) Map of the continent
C = [Datos_NorthAmerica{2};
Datos_SouthAmerica{2};
Datos_Africa{2};
Datos_Europe{2};
Datos_Asia{2};
Datos_Oceania{2}] ; % This is the vector of coordinates ...
    for the entire world
xpos = C(:,1) ; % Position in x axis
ypos = C(:,2) ; % Position in y axis
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%% We plot the continents to show how the map was ...
    constructed.
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
fz = 20 ; %fontsize
figure(1)
subplot(2,3,1)
image(Datos_NorthAmerica{3})
colormap(gray)
title('North America','fontsize',fz)
set(gca,'FontSize',fz)
%
subplot(2,3,2)
image(Datos_SouthAmerica{3})
colormap(gray)
title('South America','fontsize',fz)
set(gca,'FontSize',fz)
%
subplot(2,3,3)
image(Datos_Africa{3})
colormap(gray)
title('Africa','fontsize',fz)
set(gca,'FontSize',fz)
%
subplot(2,3,4)
image(Datos_Europe{3})
colormap(gray)
title('Europe','fontsize',fz)
set(gca,'FontSize',fz)
%
subplot(2,3,5)
```

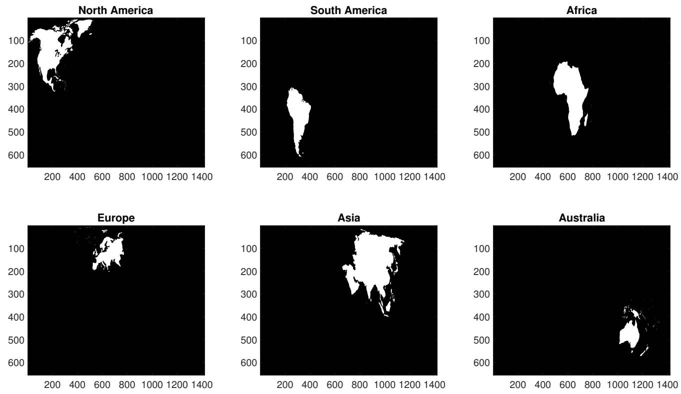

Figure 2: Figure 1 generated by the code in Example 1.

```
image(Datos_Asia{3})
colormap(gray)
title('Asia','fontsize',fz)
set(gca,'FontSize',fz)
%
subplot(2,3,6)
image(Datos_Oceania{3})
colormap(gray)
title('Australia','fontsize',fz)
set(gca,'FontSize',fz)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
figure(2)
image(Datos_NorthAmerica{3}+...
Datos_SouthAmerica{3}+Datos_Africa{3}+...
Datos_Europe{3}+...
Datos_Asia{3}+Datos_Oceania{3})
colormap(gray)
title('World','fontsize',fz)
set(gca,'FontSize',fz)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

2.2. Example 2

The main objective of this example is to present the construction of the following sets per continent: S = susceptible, I = infected, R = recovered, and D = dead, and how to plot the sets with different colors in the map.

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%% Authors: J. Barreiro-Gomez, S. E. Choutri, and H. Tembine
```

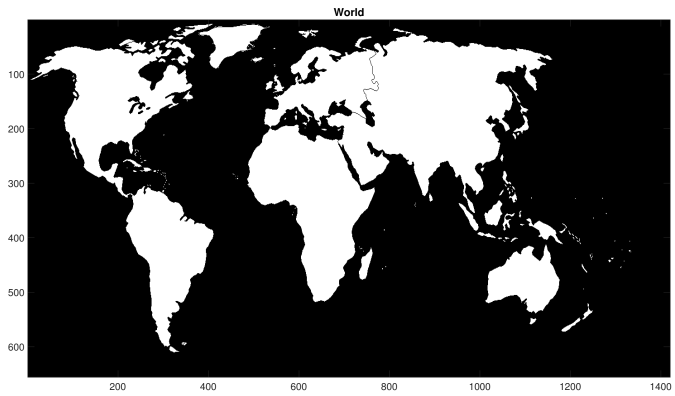

Figure 3: Figure 2 generated by the code in Example 1.

```
%%%% Learning & Game Theory Laboratory (L&G Lab)
%%%% Center on Stability, Instability and Turbulence (SITE)
%%%% NYUAD 2020
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
clear all
close all
clc
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
I = imread('continents.png') ; % The reference image is ...
    imported
resolution_grid = 8 ; % This defines the resolution
[Datos_NorthAmerica,Datos_SouthAmerica,...
Datos_Africa,Datos_Europe,Datos_Asia,...
Datos_Oceania]...
= construct_continents(I,resolution_grid) ; % Data per ...
    continent are generated
%%%% Datos are organized as follows: (1) Number of nodes ...
    in the continent,
%%%% (2) Coordinates vector, and (3) Map of the continent
C = [Datos_NorthAmerica{2};
Datos_SouthAmerica{2};
Datos_Africa{2};
Datos_Europe{2};
Datos_Asia{2};
Datos_Oceania{2}] ; % This is the vector of coordinates ...
    for the entire world
xpos = C(:,1) ; % Position in x axis
ypos = C(:,2) ; % Position in y axis
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
S_NorthAmerica = 1:Datos_NorthAmerica{1} ;
```

```
S_SouthAmerica = Datos_NorthAmerica{1}+...
(1:Datos_SouthAmerica{1}) ;
S_Africa = (Datos_NorthAmerica{1}+...
Datos_SouthAmerica{1})+(1:Datos_Africa{1}) ;
S_Europe = (Datos_NorthAmerica{1}+...
Datos_SouthAmerica{1}+Datos_Africa{1})+...
(1:Datos_Europe{1}) ;
S_Asia = (Datos_NorthAmerica{1}+...
Datos_SouthAmerica{1}+Datos_Africa{1}+...
Datos_Europe{1})+(1:Datos_Asia{1}) ;
S_Oceania = (Datos_NorthAmerica{1}+...
Datos_SouthAmerica{1}+Datos_Africa{1}+...
Datos_Europe{1}+Datos_Asia{1})+...
(1:Datos_Oceania{1}) ;
fz = 20 ; %fontsize
figure(1)
show_map(S_NorthAmerica,[],[],[],C,[1,0,0])
hold on
show_map(S_SouthAmerica,[],[],[],C,[0,1,0])
show_map(S_Africa,[],[],[],C,[0,0,1])
show_map(S_Europe,[],[],[],C,[0,1,1])
show_map(S_Asia,[],[],[],C,[1,1,0])
show_map(S_Oceania,[],[],[],C,[1,0,1])
set(gca,'color',[0,0,0]);
set(gca,'FontSize',fz)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
agent_i = 350 ;
radius_i = 20 ;
agent_d = 520 ;
radius_d = 10 ;
S_NorthAmerica = 1:Datos_NorthAmerica{1} ;
I_NorthAmerica = neighbor(agent_i,radius_i,xpos,ypos) ;
R_NorthAmerica = 500 ;
D_NorthAmerica = neighbor(agent_d,radius_d,xpos,ypos) ;
color = [0,0,1] ;
figure(2)
show_map(S_NorthAmerica,I_NorthAmerica,...
R_NorthAmerica,D_NorthAmerica,C,color)
set(gca,'color',[0,0,0]);
set(gca,'FontSize',fz)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

2.3. Example 3

The main goal of this example is to show how do we generate age for nodes representing the world. For illustration, we use a set of normal distributions to model the age in each continent. However, one can use any other

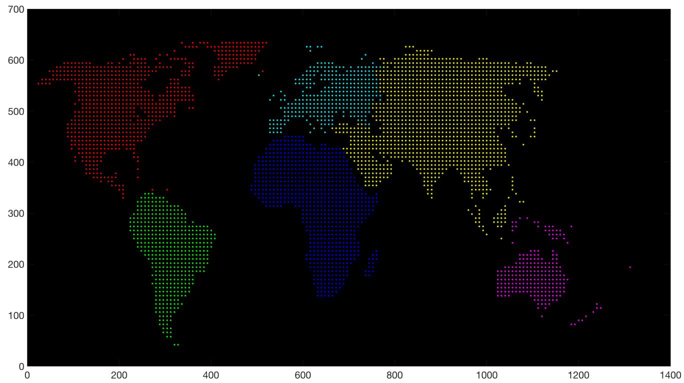

Figure 4: Figure 1 generated by the code in Example 2.

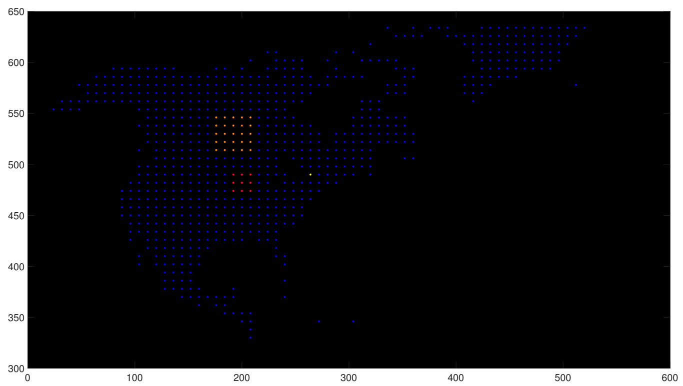

Figure 5: Figure 2 generated by the code in Example 2.

distributions such as the exponential distribution. We made the choice of the distribution as part of the parameter that the user can change.

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%% Authors: J. Barreiro-Gomez, S. E. Choutri, and H. Tembine
%%%% Learning & Game Theory Laboratory (L&G Lab)
%%%% Center on Stability, Instability and Turbulence (SITE)
%%%% NYUAD 2020
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
clear all
close all
clc
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
I = imread('continents.png') ; % The reference image is ...
    imported
resolution_grid = 8 ; % This defines the resolution
[Datos_NorthAmerica,Datos_SouthAmerica,...
Datos_Africa,Datos_Europe,Datos_Asia,...
Datos_Oceania]...
= construct_continents(I,resolution_grid) ; % Data per ...
    continent are generated
%%%% Datos are organized as follows: (1) Number of nodes ...
    in the continent,
%%%% (2) Coordinates vector, and (3) Map of the continent
C = [Datos_NorthAmerica{2};
Datos_SouthAmerica{2};
Datos_Africa{2};
Datos_Europe{2};
Datos_Asia{2};
Datos_Oceania{2}] ; % This is the vector of coordinates ...
    for the entire world
xpos = C(:,1) ; % Position in x axis
ypos = C(:,2) ; % Position in y axis
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
S_NorthAmerica = 1:Datos_NorthAmerica{1} ;
S_SouthAmerica = Datos_NorthAmerica{1}+...
(1:Datos_SouthAmerica{1}) ;
S_Africa = (Datos_NorthAmerica{1}+...
Datos_SouthAmerica{1})+(1:Datos_Africa{1}) ;
S_Europe = (Datos_NorthAmerica{1}+...
Datos_SouthAmerica{1}+Datos_Africa{1})+...
(1:Datos_Europe{1}) ;
S_Asia = (Datos_NorthAmerica{1}+...
Datos_SouthAmerica{1}+Datos_Africa{1}+...
Datos_Europe{1})+(1:Datos_Asia{1}) ;
```

```
S_Oceania = (Datos_NorthAmerica{1}+...
Datos_SouthAmerica{1}+Datos_Africa{1}+...
Datos_Europe{1}+Datos_Asia{1})+...
(1:Datos_Oceania{1}) ;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%% Here we use an arbitrary numerical example to explain ... the use of the
%%%% tool. Nevertheless, real mean and variance age for ... the different
%%%% regions should be introduced
age_mu_NA = 40; % mean age in NorthAmerica
age_sigma_NA = 20; % variance age in NorthAmerica
age_NA = max(0,normrnd(age_mu_NA,...
age_sigma_NA,1,Datos_NorthAmerica{1})) ; % vector age in ... NorthAmerica
%
age_mu_SA = 38; % mean age in SouthAmerica
age_sigma_SA = 30; % variance age in SouthAmerica
age_SA = max(0,normrnd(age_mu_SA,...
age_sigma_SA,1,Datos_SouthAmerica{1})) ; % vector age in ... SouthAmerica
%
age_mu_A = 35; % mean age in Africa
age_sigma_A = 22; % variance age in Africa
age_A = max(0,normrnd(age_mu_A,...
age_sigma_A,1,Datos_Africa{1})) ; % vector age in Africa
%
age_mu_E = 29; % mean age in Europe
age_sigma_E = 30; % variance age in Europe
age_E = max(0,normrnd(age_mu_E,...
age_sigma_E,1,Datos_Europe{1})) ; % vector age in Europe
%
age_mu_As = 42; % mean age in Asia
age_sigma_As = 30; % variance age in Asia
age_As = max(0,normrnd(age_mu_As,...
age_sigma_As,1,Datos_Asia{1})) ; % vector age in Asia
%
age_mu_O = 33; % mean age in Australia
age_sigma_O = 23; % variance age in Australia
age_O = max(0,normrnd(age_mu_O,...
age_sigma_O,1,Datos_Oceania{1})) ; % vector age in Australia
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%% Histrograms corresponding to the ages per region
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
fz = 20 ; %fontsize
figure(1)
subplot(2,3,1)
histogram(age_NA)
title('North America','fontsize',fz)
```

```
set(gca,'FontSize',fz)
subplot(2,3,2)
histogram(age_SA)
title('South America','fontsize',fz)
set(gca,'FontSize',fz)
subplot(2,3,3)
histogram(age_A)
title('Africa','fontsize',fz)
set(gca,'FontSize',fz)
subplot(2,3,4)
histogram(age_E)
title('Europe','fontsize',fz)
set(gca,'FontSize',fz)
subplot(2,3,5)
histogram(age_As)
title('Asia','fontsize',fz)
set(gca,'FontSize',fz)
subplot(2,3,6)
histogram(age_O)
title('Australia','fontsize',fz)
set(gca,'FontSize',fz)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%% Ages in the map
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
Age = [age_NA,age_SA,age_A,age_E,age_As,age_O] ;
age1 = find(Age<40) ;
age2 = find((Age≥40)&(Age<70)) ;
age3 = find(Age≥70) ;
figure(2)
show_age(C,age1,age2,age3)
title('Ages : 0-39 = Red, 40-69 = Blue, 70+ = ...
    Green','fontsize',fz)
set(gca,'color',[0,0,0]);
set(gca,'FontSize',fz)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

2.4. Example 4

First, we introduce the function named *parameters* in charge of the simulation settings.

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%% Authors: J. Barreiro-Gomez, S. E. Choutri, and H. Tembine
%%%% Learning & Game Theory Laboratory (L&G Lab)
%%%% Center on Stability, Instability and Turbulence (SITE)
%%%% NYUAD 2020
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

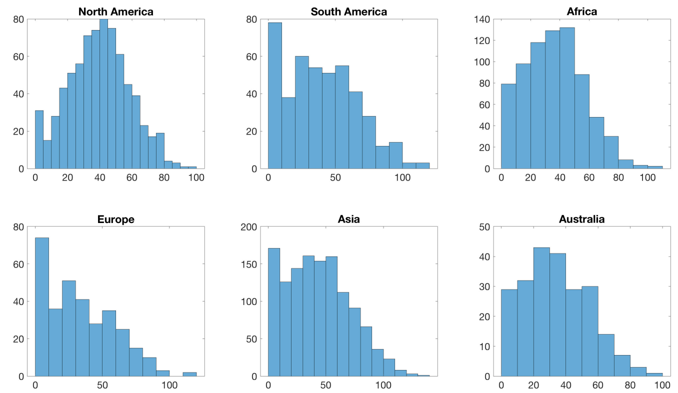

Figure 6: Figure 1 generated by the code in Example 3.

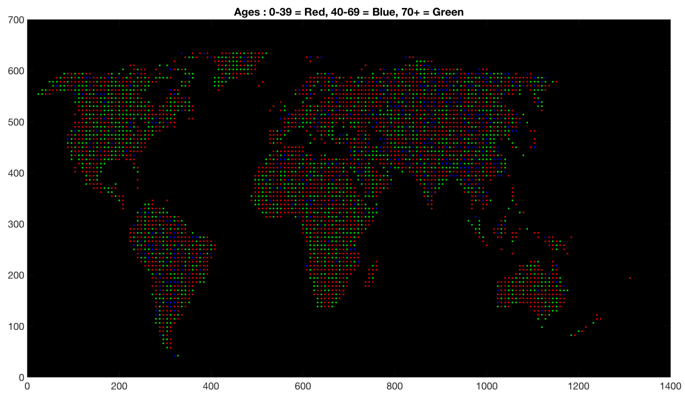

Figure 7: Figure 2 generated by the code in Example 3.

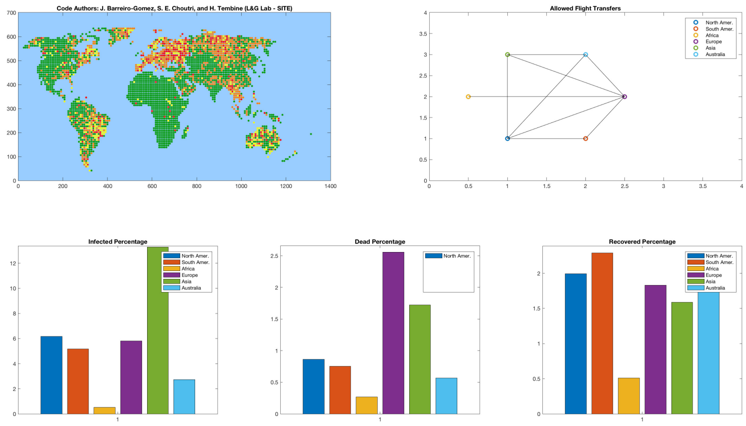

Figure 8: Figure 1 generated by the code in Example 4.

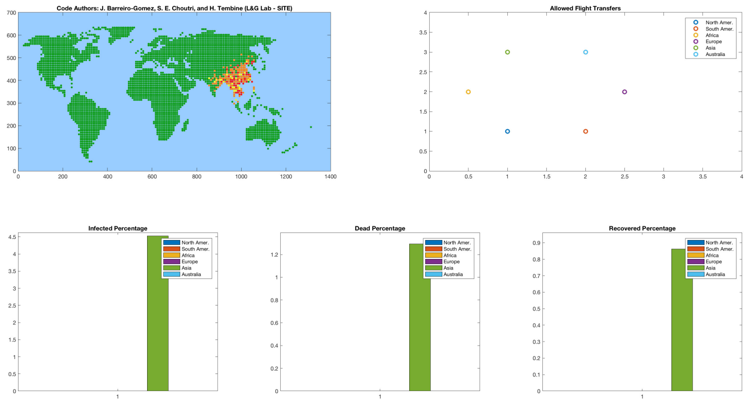

Figure 9: Figure 2 generated by the code in Example 4.

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%% In this code the parameters are introduced for the ...
    whole simulation
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
I = imread('continents.png') ; % The reference image is ...
    imported
resolution_grid = 8 ; % This defines the resolution
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
[Datos_NorthAmerica,Datos_SouthAmerica,...
Datos_Africa,Datos_Europe,Datos_Asia,...
Datos_Oceania]...
= construct_continents(I,resolution_grid) ; % Data per ...
    continent are generated
%%%% Datos are organized as follows: (1) Number of nodes ...
    in the continent,
%%%% (2) Coordinates vector, and (3) Map of the continent
C = [Datos_NorthAmerica{2};
Datos_SouthAmerica{2};
Datos_Africa{2};
Datos_Europe{2};
Datos_Asia{2};
Datos_Oceania{2}] ; % This is the vector of coordinates ...
    for the entire world
xpos = C(:,1) ; % Position in x axis
ypos = C(:,2) ; % Position in y axis
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
S_NorthAmerica = 1:Datos_NorthAmerica{1} ;
S_SouthAmerica = Datos_NorthAmerica{1}+...
(1:Datos_SouthAmerica{1}) ;
S_Africa = (Datos_NorthAmerica{1}+...
Datos_SouthAmerica{1})+(1:Datos_Africa{1}) ;
S_Europe = (Datos_NorthAmerica{1}+...
Datos_SouthAmerica{1}+Datos_Africa{1})+...
(1:Datos_Europe{1}) ;
S_Asia = (Datos_NorthAmerica{1}+...
Datos_SouthAmerica{1}+Datos_Africa{1}+...
Datos_Europe{1})+(1:Datos_Asia{1}) ;
S_Oceania = (Datos_NorthAmerica{1}+...
Datos_SouthAmerica{1}+Datos_Africa{1}+...
Datos_Europe{1}+...
Datos_Asia{1})+(1:Datos_Oceania{1}) ;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%% Here we use an arbitrary numerical example to explain ...
    the use of the
%%%% tool. Nevertheless, real mean and variance age for ...
    the different
%%%% regions should be introduced
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
age_mu_NA = 40; % mean age in NorthAmerica
age_sigma_NA = 20; % variance age in NorthAmerica
age_NA = max(0,normrnd(age_mu_NA,...
age_sigma_NA,1,Datos_NorthAmerica{1})) ; % vector age in ...
    NorthAmerica
%
age_mu_SA = 38; % mean age in SouthAmerica
age_sigma_SA = 30; % variance age in SouthAmerica
age_SA = max(0,normrnd(age_mu_SA,...
age_sigma_SA,1,Datos_SouthAmerica{1})) ; % vector age in ...
    SouthAmerica
%
age_mu_A = 35; % mean age in Africa
age_sigma_A = 22; % variance age in Africa
age_A = max(0,normrnd(age_mu_A,...
age_sigma_A,1,Datos_Africa{1})) ; % vector age in Africa
%
age_mu_E = 29; % mean age in Europe
age_sigma_E = 30; % variance age in Europe
age_E = max(0,normrnd(age_mu_E,...
age_sigma_E,1,Datos_Europe{1})) ; % vector age in Europe
%
age_mu_As = 42; % mean age in Asia
age_sigma_As = 30; % variance age in Asia
age_As = max(0,normrnd(age_mu_As,...
age_sigma_As,1,Datos_Asia{1})) ; % vector age in Asia
%
age_mu_O = 33; % mean age in Australia
age_sigma_O = 23; % variance age in Australia
age_O = max(0,normrnd(age_mu_O,...
age_sigma_O,1,Datos_Oceania{1})) ; % vector age in Australia
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
Age = [age_NA,age_SA,age_A,age_E,...
age_As,age_O] ;
age1 = find(Age<40) ;
age2 = find((Age≥40)&(Age<70)) ;
age3 = find(Age≥70) ;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Parameters to tune
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%p = [pv1,pv2,pm31,pm32,pm33,pr4,p5]
%pv1 (Virality) probability of getting infected if ...
    neighbor of a sick agent
%pv2 (Virality) probability that the virus covers long ...
    distances (airplane)
%pm31 (Mortality) probability of death (age−dependent) [0−39]
%pm32 (Mortality) probability of death (age−dependent) [40−69]
%pm33 (Mortality) probability of death (age−dependent) [70+]
%pr4 (Recovery) probability of being recovered
```

```
%ps5 (Again Sick) probability of being sick after having ...
    been recovered
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
S_NorthAmerica = 1:Datos_NorthAmerica{1} ;
I_NorthAmerica = [] ;
R_NorthAmerica = [] ;
D_NorthAmerica = [] ;
p_NorthAmerica = [0.3,0.02,0.01,0.08,0.08,0.1,0.01] ;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
S_SouthAmerica = Datos_NorthAmerica{1}+...
(1:Datos_SouthAmerica{1}) ;
I_SouthAmerica = [] ;
R_SouthAmerica = [] ;
D_SouthAmerica = [] ;
p_SouthAmerica = [0.3,0.02,0.01,0.08,0.08,0.1,0.01] ;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
S_Africa = (Datos_NorthAmerica{1}+...
Datos_SouthAmerica{1})+(1:Datos_Africa{1}) ;
I_Africa = [] ;
R_Africa = [] ;
D_Africa = [] ;
p_Africa = [0.05,0.02,0.01,0.08,0.08,0.1,0.01] ;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
S_Europe = (Datos_NorthAmerica{1}+...
Datos_SouthAmerica{1}+Datos_Africa{1})+...
(1:Datos_Europe{1}) ;
I_Europe = [] ;
R_Europe = [] ;
D_Europe = [] ;
p_Europe = [1,0.2,0.15,0.02,0.02,0.1,0.03] ;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
S_Asia = (Datos_NorthAmerica{1}+...
Datos_SouthAmerica{1}+...
Datos_Africa{1}+Datos_Europe{1})+...
(1:Datos_Asia{1}) ;
I_Asia = 3442 ;
R_Asia = [] ;
D_Asia = [] ;
p_Asia = [1,0.2,0.15,0.02,0.02,0.1,0.03] ;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
S_Oceania = (Datos_NorthAmerica{1}+...
Datos_SouthAmerica{1}+Datos_Africa{1}+...
Datos_Europe{1}+Datos_Asia{1})+...
(1:Datos_Oceania{1}) ;
I_Oceania = [] ;
R_Oceania = [] ;
D_Oceania = [] ;
p_Oceania = [0.1,0.02,0.005,0.08,0.08,0.1,0.01] ;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
ite = 1000 ; % number of iterations
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%% Here user decides the posible flight connecvtions%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%% North_America / South America / Africa / Europe / ...
    Asia / Oceania
A = [0 1 0 1 1 1 ;
1 0 0 1 0 0 ;
0 0 0 1 0 0 ;
1 1 1 0 1 1 ;
1 0 0 1 0 1 ;
1 0 0 1 1 0] ; % adjacency matrix defining possible flight ...
    connections
[fc_NorthAmerica,fc_SouthAmerica,...
fc_Africa,fc_Europe,fc_Asia,fc_Oceania] = ...
flight_connections(A,S_NorthAmerica,...
S_SouthAmerica,S_Africa,S_Europe,...
S_Asia,S_Oceania) ;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

The following is the main code generates the whole simulation of the virus propagation.

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%% Authors: J. Barreiro-Gomez, S. E. Choutri, and H. Tembine
%%%% Learning & Game Theory Laboratory (L&G Lab)
%%%% Center on Stability, Instability and Turbulence (SITE)
%%%% NYUAD 2020
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
clear all
close all
clc
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%% Parameters are uploaded
parameters
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
figure(1)
subplot(2,2,1)
set(gcf,'Position',[0,1000,1400,800])
set(gca,'color',[153,204,255]/255);
green = [0,153,0]/255 ;
show_map(S_NorthAmerica,...
I_NorthAmerica,R_NorthAmerica,...
D_NorthAmerica,C,green)
hold on
```

```
show_map(S_SouthAmerica,...
I_SouthAmerica,R_SouthAmerica,...
D_SouthAmerica,C,green)
show_map(S_Africa,...
I_Africa,R_Africa,...
D_Africa,C,green)
show_map(S_Europe,...
I_Europe,R_Europe,...
D_Europe,C,green)
show_map(S_Asia,...
I_Asia,R_Asia,...
D_Asia,C,green)
show_map(S_Oceania,...
I_Oceania,R_Oceania,...
D_Oceania,C,green)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
for t = 1:ite
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
figure(1)
subplot(2,2,2)
graph_plot(A)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
S = S_NorthAmerica ;
I = I_NorthAmerica ;
R = R_NorthAmerica ;
D = D_NorthAmerica ;
p = p_NorthAmerica ;
flight_connections = fc_NorthAmerica ;
[S_dif_NA,I_dif_NA,R_dif_NA,...
D_dif_NA,S_NorthAmerica,...
I_NorthAmerica,R_NorthAmerica,...
D_NorthAmerica,FSA_NA,Flag_NA] =...
 update(S,I,R,D,xpos,ypos,...
 flight_connections,p,Age) ;
%
S = S_SouthAmerica ;
I = I_SouthAmerica ;
R = R_SouthAmerica ;
D = D_SouthAmerica ;
p = p_SouthAmerica ;
flight_connections = fc_SouthAmerica ;
[S_dif_SA,I_dif_SA,R_dif_SA,...
D_dif_SA,S_SouthAmerica,...
I_SouthAmerica,R_SouthAmerica,...
D_SouthAmerica,...
FSA_SA,Flag_SA] =...
 update(S,I,R,D,xpos,ypos,...
 flight_connections,p,Age) ;
%
```

```
S = S_Africa ;
I = I_Africa ;
R = R_Africa ;
D = D_Africa ;
p = p_Africa ;
flight_connections = fc_Africa ;
[S_dif_A,I_dif_A,R_dif_A,...
D_dif_A,S_Africa,I_Africa,...
R_Africa,D_Africa,FSA_A,Flag_A] =...
 update(S,I,R,D,xpos,ypos,...
 flight_connections,p,Age) ;
%
S = S_Europe ;
I = I_Europe ;
R = R_Europe ;
D = D_Europe ;
p = p_Europe ;
flight_connections = fc_Europe ;
[S_dif_E,I_dif_E,R_dif_E,...
D_dif_E,S_Europe,I_Europe,...
R_Europe,D_Europe,FSA_E,Flag_E] =...
 update(S,I,R,D,xpos,ypos,...
 flight_connections,p,Age) ;
%
S = S_Asia ;
I = I_Asia ;
R = R_Asia ;
D = D_Asia ;
p = p_Asia ;
flight_connections = fc_Asia ;
[S_dif_As,I_dif_As,R_dif_As,...
D_dif_As,S_Asia,I_Asia,R_Asia,...
D_Asia,FSA_As,Flag_As] =...
 update(S,I,R,D,xpos,ypos,...
 flight_connections,p,Age) ;
%
S = S_Oceania ;
I = I_Oceania ;
R = R_Oceania ;
D = D_Oceania ;
p = p_Oceania ;
flight_connections = fc_Oceania ;
[S_dif_O,I_dif_O,R_dif_O,D_dif_O,...
S_Oceania,I_Oceania,R_Oceania,...
D_Oceania,FSA_O,Flag_O] = ...
update(S,I,R,D,xpos,ypos,...
flight_connections,p,Age) ;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
FSA = [FSA_NA,FSA_SA,...
```

```
FSA_A,FSA_E,FSA_As,FSA_O] ;
%
I_dif = I_dif_NA ;
S = S_NorthAmerica ;
I = I_NorthAmerica ;
[I_dif_NA,I_NorthAmerica] =...
 long_dist_update(FSA,I_dif,S,I) ;
%
I_dif = I_dif_SA ;
S = S_SouthAmerica ;
I = I_SouthAmerica ;
[I_dif_SA,I_SouthAmerica] =...
 long_dist_update(FSA,I_dif,S,I) ;
%
I_dif = I_dif_A ;
S = S_Africa ;
I = I_Africa ;
[I_dif_A,I_Africa] =...
 long_dist_update(FSA,I_dif,S,I) ;
%
I_dif = I_dif_E ;
S = S_Europe ;
I = I_Europe ;
[I_dif_E,I_Europe] =...
 long_dist_update(FSA,I_dif,S,I) ;
%
I_dif = I_dif_As ;
S = S_Asia ;
I = I_Asia ;
[I_dif_As,I_Asia] =...
 long_dist_update(FSA,I_dif,S,I) ;
%
I_dif = I_dif_O ;
S = S_Oceania ;
I = I_Oceania ;
[I_dif_O,I_Oceania] =...
 long_dist_update(FSA,I_dif,S,I) ;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
figure(1)
subplot(2,2,1)
set(gcf,'Position',[0,1000,1400,800])
set(gca,'color',[153,204,255]/255);
show_map([],I_dif_NA,R_dif_NA,...
D_dif_NA,C,green)
hold on
show_map([],I_dif_SA,R_dif_SA,...
D_dif_SA,C,green)
show_map([],I_dif_A,R_dif_A,...
```

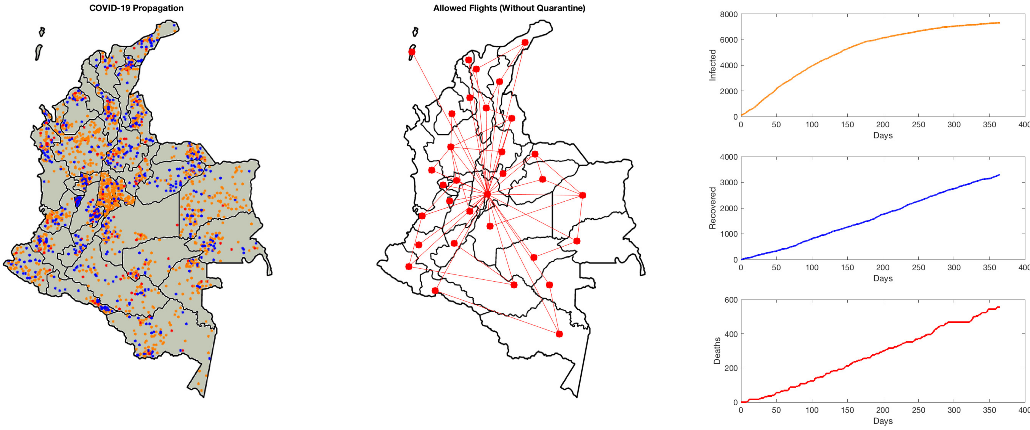

Figure 10: Propagation of the virus in Colombia with migration constraints among the 32 departments (without quarantine).

```
D_dif_A,C,green)
show_map([],I_dif_E,R_dif_E,...
D_dif_E,C,green)
show_map([],I_dif_As,R_dif_As,...
D_dif_As,C,green)
show_map([],I_dif_O,R_dif_O,...
D_dif_O,C,green)
title('Code Authors: J. Barreiro-Gomez,...
 S. E. Choutri, and H. Tembine (L&G Lab - SITE)')
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
bar_plots
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%for slow = 1 : 3
%     F = getframe(gcf) ;
%     writeVideo(v,F) ;
%end
end
%close(v) ;
```

The last called function *bar plots* is as follows and it is in charged of the publication of results:

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%% Authors: J. Barreiro-Gomez, S. E. Choutri, and H. Tembine
%%%% Learning & Game Theory Laboratory (L&G Lab)
%%%% Center on Stability, Instability and Turbulence (SITE)
```

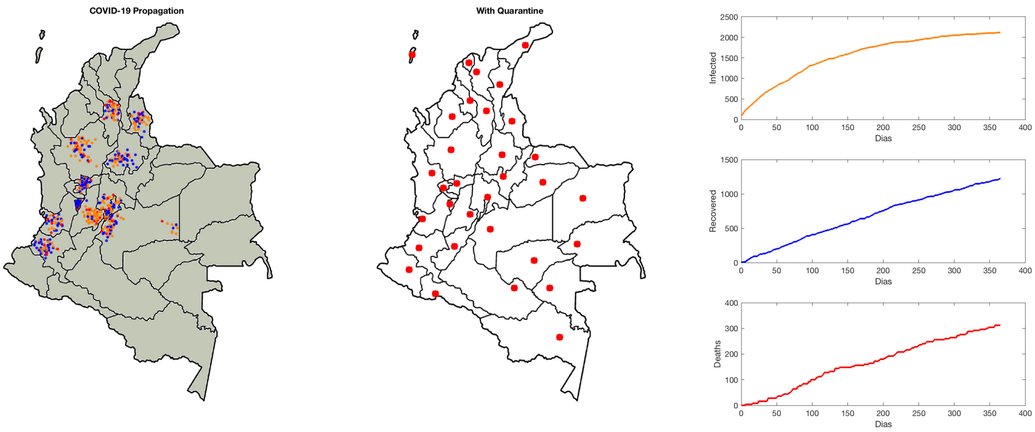

Figure 11: Propagation of the virus in Colombia with migration constraints among the 32 departments (with quarantine).

```
%%%% NYUAD 2020
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
figure(1)
Total = Datos_NorthAmerica{1}+...
Datos_SouthAmerica{1}+Datos_Africa{1}+...
Datos_Europe{1}+Datos_Asia{1}+...
Datos_Oceania{1} ;
subplot(2,3,4)
y1 = 100*[(length(I_NorthAmerica)+...
length(R_NorthAmerica))/Total,...
 (length(I_SouthAmerica)+...
length(R_SouthAmerica))/Total,...
 (length(I_Africa)+length(R_Africa))/Total,...
 (length(I_Europe)+length(R_Europe))/Total,...
 (length(I_Asia)+length(R_Asia))/Total,...
 (length(I_Oceania)+length(R_Oceania))/Total;
0,0,0,0,0,0];
bar(y1)
title('Infected Percentage')
legend('North Amer.', 'South Amer.',...
 'Africa' , 'Europe', 'Asia', 'Australia')
axis([0.5 1.5 0 max(y1(1,:))+0.1])
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
subplot(2,3,5)
y1 = 100*[(length(D_NorthAmerica))/Total,...
 (length(D_SouthAmerica))/Total,...
 (length(D_Africa))/Total,...
 (length(D_Europe))/Total,...
 (length(D_Asia))/Total,...
```

```
(length(D_Oceania))/Total;
0,0,0,0,0,0];
bar(y1)
title('Dead Percentage')
legend('North Amer.', 'South Amer.', ...
'Africa' , 'Europe', 'Asia', 'Australia')
axis([0.5 1.5 0 max(y1(1,:))+0.1])
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
subplot(2,3,6)
y1 = 100*[(length(R_NorthAmerica))/Total,...
(length(R_SouthAmerica))/Total,...
(length(R_Africa))/Total,...
(length(R_Europe))/Total,...
(length(R_Asia))/Total,...
(length(R_Oceania))/Total;
0,0,0,0,0,0];
bar(y1)
title('Recovered Percentage')
legend('North Amer.', 'South Amer.',...
 'Africa' , 'Europe', 'Asia', 'Australia')
axis([0.5 1.5 0 max(y1(1,:))+0.1])
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

2.5. Example 5

The presented toolbox can be easily adapted in order to consider any set of regions. To illustrate this, let us consider a unique country divided into several regions. In particular, let us consider Colombia, which is divided into 32 departments. Therefore, we consider not only the different transition rates for the departments, but also the migration constraints among them. Figure 10 shows the propagation of the virus during a year without quarantine whereas Figure 11 shows the propagation of the virus during a year with quarantine and no allowed domestic flights.