UNIVERSIDAD DE INVESTIGACIÓN DE TECNOLOGÍA EXPERIMENTAL YACHAY

Escuela de Ciencias Físicas y Nanotecnología

TÍTULO: Coexisting dynamical processes in complex networks

Trabajo de integración curricular presentado como requisito para la obtención del título de Físico

Autor:

Capa Sandoval Victor Alexander

Tutor:

Cosenza Miceli Mario Giuseppe, PhD.

Urcuquí, mayo de 2024.

Autoría

Yo, **CAPA SANDOVAL VICTOR ALEXANDER**, con cédula de identidad **172421671-6**, declaro que las ideas, juicios, valoraciones, interpretaciones, consultas bibliográficas, definiciones y conceptualizaciones expuestas en el presente trabajo; así cómo, los procedimientos y herramientas utilizadas en la investigación, son de absoluta responsabilidad del autor del trabajo de integración curricular. Asimismo, me acojo a los reglamentos internos de la Universidad de Investigación de Tecnología Experimental Yachay.

Urcuquí, mayo de 2024.

Αλεξανδερ Καππα

Victor Alexander Capa Sandoval

CI:1724216716

Autorización de publicación

Yo, **CAPA SANDOVAL VICTOR ALEXANDER**, con cédula de identidad **172421671-6**, cedo a la Universidad de Investigación de Tecnología Experimental Yachay, los derechos de publicación de la presente obra, sin que deba haber un reconocimiento económico por este concepto. Declaro además que el texto del presente trabajo de titulación no podrá ser cedido a ninguna empresa editorial para su publicación u otros fines, sin contar previamente con la autorización escrita de la Universidad.

Asimismo, autorizo a la Universidad que realice la digitalización y publicación de este trabajo de integración curricular en el repositorio virtual, de conformidad a lo dispuesto en el Art. 144 de la Ley Orgánica de Educación Superior.

Urcuquí, mayo de 2024.

Αλεξανδερ Καππα

Victor Alexander Capa Sandoval

CI:1724216716

Dedication

For all the people who take care of me, accompany me and worry about me, whom I can consider my family. Each of you has built a little bit of who I am.

Acknowledgment

Resumen

La reciente pandemia de COVID-19 ha generado mucho interés en la investigación de la relación entre el proceso de propagación de enfermedades en redes y los cambios de comportamiento de los individuos de una población. En esta Tesis, investigamos un modelo de enfermedad infecciosa junto con un juego evolutivo de formación de opinión en agentes sociales que coexisten y coevolucionan en varias redes complejas, y cómo la información global y local sobre una enfermedad afecta el surgimiento de la cooperación colectiva en las redes sociales. Además, estudiamos el papel de las propiedades topológicas de los grafos para permitir que emerja la cooperación. Finalmente, analizamos otras dinámicas de propagación de epidemias y juegos evolutivos sobre los comportamientos colectivos emergentes en estas redes complejas.

Palabras clave: Sistemas complejos, sociofísica, redes complejas, coevolución, modelos compartimentales en epidemiología, teoría de juegos, comportamiento colectivo.

Abstract

The recent COVID-19 pandemic has generated much interest in the research of the relationship between the process of disease spreading on networks and the behavioral changes of individuals in a population. In this Thesis, we investigate an infectious disease model coupled with a evolutionary game of opinion formation in social agents coexisting and coevolving on several complex networks, and how global and local information about a disease affect the emergence of collective cooperation in social networks. Additionally, we study the role of the topological properties of the graphs in allowing cooperation to occur. Finally, we look into other epidemic propagation dynamics and evolutionary games on the emergent collective behaviors in these complex networks.

Keywords: Complex systems, complex networks, sociophysics, coevolution, compartmental models in epidemiology, game theory, collective behavior.

Contents

List of Figures

Chapter 1

Introduction

1.1 Dialectics and complexity

Result of the political transformation of the entire world in the 15th century, the fall of the feudal nobility, the founding of the great national kingdoms, the basis of the states that today hold economic power, and the *renaissance* of the arts, inspired by Greco-Roman classical antiquity, modern scientific research of nature was a revolution against the ancient philosophy of nature. Like any revolution, it sought to conquer its *right to life*, especially with the mortal persecution of its first martyrs, such as Giordano Bruno.

With Copernicus, less transgressive than Bruno and before him, the *independence* of modern research was declared in the face of religious authority over nature, and over time, it would gain strength by leaps and bounds, with the task of dominating all the matter. For this reason, the greatest development of the time concerned the mechanics of bodies and the improvement of the mathematical methods that it requires.

However, the search for this perfection in *motion* concludes with the conception of the *absolute immutability of nature*. The planets, for example, in their motion with a mysterious origin, would remain there forever, in perfect ellipses according to *universal gravitation*. Fauna and flora were established at birth, since *like begets like*, with exceptions due to human engineering. The continents, mountains, glaciers, and rivers have always been there, except for modifications by man.

So, while humanity had a history, in time, the history of nature was restricted to its development in space. In the words of Engels [3]: "Copernicus, at the beginning of the period, writes a letter renouncing theology; Newton closes the period with the postulate of a divine first impulse." Thus, the science of nature, once revolutionary, became conservative, until today, where it is a fundamental doctrine of uncritical use.

To deal a blow to the *mechanistic* or static conception of nature, decisive progress was needed

1

in the different branches of knowledge: in chemistry, with the categorization of the processes of transformation of inorganic matter, as *alive* as of organic matter; in geology, with the discovery of remote strata and fossils of species that no longer exist; in biology, with the development of the theory of evolution and the discovery of genetic mutations; and in physics, with thermodynamics, phase transitions, and quantum mechanics, while physical forces are reduced to the motion of matter or, in other words, the flow of energy. Nature has changed over time, where "everything that exists deserves to perish." This is the manifestation of **Dialectics**, as a study of motion and interconnections. It means that the history of nature is studied as the motion, change, or process that matter undergoes, not in the solitude of its existence, but as part of a real system of which it is a part, dependent on it.

Complexity arises from the fundamental premise that development is not linearly continuous and that various elements acting together produce something more than the sum of their parts. This science arises as a result of the technological advance in computational science, which allows working with a greater amount of information and therefore allows access to knowledge about patterns and processes that only occur when all this new information is taken into consideration, be it the appearance of chaos due to very small fractions of change in the parameters of a system or due to the addition of tens, hundreds, thousands, or millions of elements that interact with each other.

The science of complexity proposes the development of nature, according to the laws of dialectic, as a change from quantity to quality, unity and contradiction of opposites, and negation of the negation: the emergence of new phenomena from old and simpler relationships or interactions; the interconnection of phenomena; the transition to chaos; the emergence of structures and global order from chaos or random activity; and the existence of system boundaries known as attractors [4].

This recent science is part of the revolution in the methods of obtaining knowledge and surely finds its place in this time of the historical development of humanity, following the premise of replacing immutability with emergence and fixed characteristics with motion.

1.2   Complex Systems

In recent years, Physics and other sciences have have created the concept of complex system to describe a diversity of natural and artificial systems within a unified framework [5] [6]. A complex system comprises a set of interacting dynamical elements whose collective behavior cannot be derived from the knowledge of the properties of the isolated elements. The collective behavior is said to *emerge* from the interactions between the components. The field of complex systems investigates how a large set of components – locally interacting with each other at small scales –

can spontaneously self-organize to exhibit non-trivial structures, functionalities, and behaviors at larger scales, without external intervention, specific design, central control, or leaders. The global properties of a complex system may not be inferred or predicted from the full knowledge of its constituents alone.

The components form a network of interactions. The main challenge of complexity science is not only to see the parts and their connections but also to understand how these connections give rise to the whole. In simple systems, the properties of the whole can be understood or predicted from the addition or aggregation of its components. In other words, macroscopic properties of a simple system can be deduced from the microscopic properties of its parts. In complex systems, however, the properties of the whole often cannot be understood or predicted from the knowledge of its components because of the phenomenon of *emergence*. This phenomenon involves diverse mechanisms causing the interaction between components of a system to generate novel information and exhibit non-trivial collective behaviors at larger scales. This fact is usually summarized with the popular phrase "the whole is more than the sum of its parts".

A paradigmatic example of a complex system is the brain [5]. It is well known how a single neuron functions. A single neuron cannot think nor have consciousness by itself, but a network of billions of them forming the brain can give rise to thought, conscience, and emotions. The concept of complex system takes on another interesting manifestation in social systems, where human interactions can lead to the emergence of diverse social and political structures. The application of the concepts and methods of complex systems to study social systems has been called *Sociophysics* [7].

1.3   Complex Networks

There are many systems composed of components interacting and linked together in some way. Examples include human societies, which are collections of people linked by acquaintance or social interaction, and the Internet, a collection of computers linked by data exchanges. The pattern of connections in a given system can be represented as a network. A network is a collection of points or nodes joined together in pairs by lines called links or edges.

In complex systems, the interactions between components can be characterized as links forming a complex network, where the topology of the connectivity is not uniform nor trivial. The research on natural and technological complex networks has exponentially increased in recent years thanks to the access to large databases and the availability of high speed computer power that allows the management and analysis of these large amounts of data in a very short time. Complex networks constitute the backbone of complex systems. Many systems of interest in the physical, biological, and social sciences can be represented as complex networks.

There can be more than one type of node in the network, and the links can be different to represent more complex relations among the nodes. In this manner, the links can have a direction, in the sense that a node i has a relation with node j but the node j has no relation with node i. There can be also relations that are not equal among all nodes but are stronger or weaker depending on the nodes involved. This can be represented by a weighted link, where the strength of the link is quantified. The state of the nodes and their links can also change over time. Sophistication can be added to include these and others variations observed in real systems. The study of complex networks has evolved into a contemporary, interdisciplinary science.

1.4   Coevolutionary Networks

Many networks observed in nature and society are not static. Actually, many natural and technological complex systems can be represented as dynamical networks of interacting elements, or nodes, where the connections between the elements and their state variables evolve simultaneously [8]. The links describing the interactions between nodes can vary their strengths or appear and disappear as the system evolves on diverse timescales. In multiple cases, these variations in the topology of the network arise from a feedback effect of the dynamics of the states of the nodes: the network changes in response to the evolution of those states which in turn determines the modification of the network. Systems that possess this coupling between the topology and states have been denoted as coevolutionary dynamical systems or adaptive networks [8]. The collective behaviors emerging in coevolutionary systems depend on the competition between the time scales of these two coexisting processes: the dynamics of states of the nodes and the dynamics of the network connections. Coevolution has been investigated in the context of spatiotemporal dynamical systems, such as neural networks, coupled map lattices, motile elements, game theory, models of social dynamics, and epidemic propagation [8].

1.5   Research Problem

A problem of great interest and importance, where the phenomenon of coevolution often appears, is the propagation of epidemics. The recent COVID 19 pandemic has made humanity more aware of the importance of the research into the dynamics of disease spreading in networks.

Infectious diseases are transmitted through social contacts between individuals. The modeling of epidemic spreading among human beings has been extensively studied in mathematical epidemiology and network science. The developments of transportation system have enabled people to travel more globally. Consequently, epidemics starting from a local market can spread to the entire network in a very short time.

Human beings often react to the presence of an infectious disease by changing their behavior. The perception of the risk associated with the infection and countermeasures are usually accompanied by changes in behavior and opinions of the social agents, such as cutting the connection with infectious contacts, accepting vaccination, wearing face-masks, reducing travel range, along with others [9, 10, 11]. Agents acting as information carriers may pass the message of the epidemic situation to uninformed individuals, which may potentially alter their future mobility patterns, thus, affecting the epidemic spread. For example, COVID-19 has revealed the importance of social distancing to reduce infection risk within the population so the capacity of health systems is not saturated [9]. As epidemics evolve, individuals obtain information provided by health institutions concerning the status of an epidemic or are strongly influenced by beliefs of the disease in their population.

Motivated by the relevance of these issues in today's world, in this Thesis we investigate a general model for epidemic propagation where both the epidemics dynamics and the opinion formation coexist in a network of social agents and can coevolve with the topology of the underlying network. We address the questions of how a cooperation behavior can arise and what properties of the network affect the propagation of epidemics in a social network.

1.6 Objectives

1.6.1 General objective

To understand the emergence of collective cooperation by a replicator rule in several epidemic compartmental models, through the properties of complex networks.

1.6.2 Specific objectives

1. To consider different types of networks with the same number of nodes, but with different properties.

2. To study the origin of the emergence of cooperation in a network with an evolutionary game dynamics linked to a epidemic dynamics.

3. To investigate the occurrence of the cooperation for different evolutionary game dynamics and sundry epidemic models.

4. To interpret the results according to the properties of the networks and the dynamical processes.

1.7 Overview

Chapter 2 describes the theoretical framework on which this Thesis is founded. We present a review of the main concepts employed in this Thesis, comprising complex networks, compartmental epidemic models, and evolutionary game theory. Chapter 3 contains our implementation of a model with coexisting and coevolving dynamics of epidemic propagation and cooperation game on different networks of social agents. We delineate the mechanism of coupling between the evolutionary game and the epidemic susceptible-infected-susceptible (SIS) compartmental model and show the algorithms that we have elaborated for the computer simulations of the model. We investigate the role of the topology of connectivity on the emergence of collective cooperation by exploring several different complex networks. In Chapter 4, we propose a generalization of the model by considering other epidemic compartmental models and game dynamics for social behavior. The algorithms for these new models are shown. Chapter 5 presents the conclusions of our work, exposing significant findings and providing ideas for future research. The Appendix includes the computer code for the simulations in Python 3.9 elaborated by the author of this Thesis.

Chapter 2

Theoretical Framework

2.1   Graphs and complex networks

Networks are structures that combine a set of elements and the relationships between them. In mathematics they are called graphs, where the elements are named nodes, points or vertices, while the relationships, weighted or not, between a pair of vertices are named edges, links or lines [12]. They are the object of study of graph theory as part of discrete mathematics that formally defines networks like this [13]:

Given a finite set $V = \{v_1, v_2, \ldots, v_n\}$ of non-specific elements, and the set $V \otimes V$, of all ordered pairs $[v_i, v_j]$ of the elements of V, a network is the triplet $G = (V, W, f)$, where V is the finite set of nodes, $W = \{w_1, w_2, \ldots, w_n | w_i \in \mathbb{R}\}$ the set of the weights of each connection and f, a mapping that associates some elements of W to an element of $V \otimes V$, $f : w_p \rightarrow [v_i, v_j]$.

These structures are abstractions of a variety of natural and artificial systems. For example, the World Wide Web (WWW) (Figure 2.1), ecological or social relationships [14], economic equilibrium models of commodity flows in hierarchical societies [15], or artificial neural networks that control the content that is shown to users on social networks such as YouTube [16].

Complex networks, like any complex system, are a set of elements that when interacting with each other, behaviors emerge that are not necessarily explained by the properties of the individual elements. For the physical study of networks, the nodes can represent any biological, socioeconomic, technological entity, among others, and we can classify the network according to their nature. In the same way, we can classify the network according to what its links represent, which can be: physical or tangible links, such as a road; physical interactions, such as forces or energy transferred; intangible connections, such as data or information; social interactions, such as friendships or family ties, or conceptual, such as references or dictionaries [13].

We can distinguish some important types of networks according to their topological characteristics, such as regular grids, small-world networks and scale-free networks.

Figure 2.1: Wikipedia as part of the World Wide Web [1]. The labeled nodes represent several Internet web pages, and in the center, the main page of Wikipedia. The links correspond to hyperlinks between these pages.

2.1.1 Grid graph or Lattice

A grid graph is a regular homogeneous network. Mathematically, it is an n-dimensional graph, which forms a mosaic of regular geometric figures. The most common grid network is the square lattice, as the one in Figure 2.2, where the coordinates of the first dimension are in the range $1, \ldots, n$, those of the second dimension are $1, \ldots, m$, and so on. Any two vertices are connected by a link if its corresponding distance is 1 [17].

2.1.2 Small-world network

In a small-world network, although each node has a small number of neighbors relative to the size of the network, most nodes can reach any other node in a short number of "hops" or links. They were devised as a conceptualization of Stanley Milgram's six degrees of separation experiment [18]. The number of links between nodes, which is called degree, follows a Poisson distribution, as in Figure 2.3b. These networks are commonly constructed as random networks. A random network consists of nodes with random connections between them, and most nodes have approximately the same number of links [19]. Specifically, small-world networks that follow the Watts-Strogatz model have highly clustered nodes, or high clustering coefficient, as in regular networks, but have

Figure 2.2: 4×4-grid network. Colors represents the three types of nodes that this graph has according to their degree.

small characteristic path lengths, such as random graphs [20].
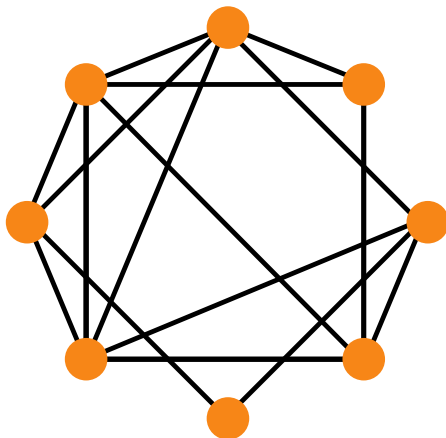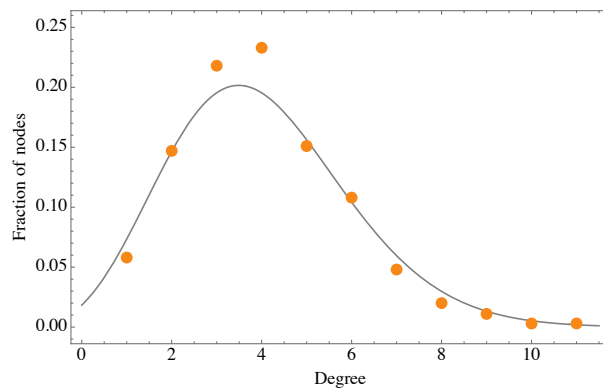
| (a) | (b) |

Figure 2.3: (a) 8-nodes small-world network representation. (b) Typical degree distribution of a small-world network compared with its Poisson distribution.

2.1.3 Scale-free network

Scale-free networks were an improvement over what was understood as a complex network (which was almost always related to the random and chaotic ones). They are defined as networks that have a few highly connected nodes, while others have few links, as seen in representation of Figure 2.4. The most connected nodes, called hubs, have a number of links several orders of magnitude

different from the rest. In this sense, the degree distribution follows a power law, $P_D \sim \kappa^\gamma$, where κ is the degree and γ is the power-law parameter, whose value is in range $2 < \gamma < 3$.
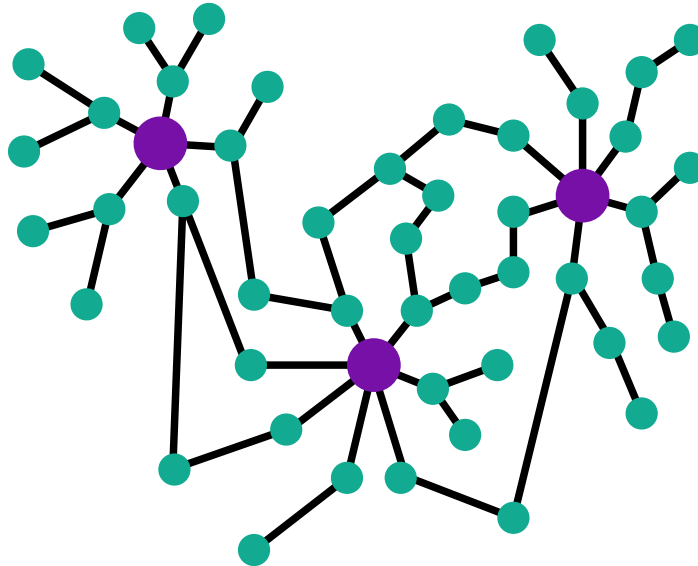
Figure 2.4: Scale-free network representation. Purple points represent the hubs.

In order to build a scale-free network, we use two basic mechanisms: one is growth, that is, you start with few connected nodes, and new vertices are added; the other is preferential coupling, meaning that new nodes have a preference or greater probability of connecting to nodes that already have more numbers of edges [19]. Examples of scale-free networks are Internet networks, the WWW, interpersonal relationships, or cellular metabolism [21].

2.2 Compartmental epidemic models

Mathematically studying the spread of contagious diseases can be very useful for its control. From knowing the speed of reproduction of the disease or the probabilities of contagion, to defining the time in which a population reaches the maximum or minimum numbers of infected or recovered, they help epidemiologists to be able to define strategies to face a possible epidemic or pandemic, such as identifying trends and formulating general forecasts [2], and avoiding the greatest number of serious cases that could collapse health systems, or fatalities that the disease could cause, beyond the work that biologists, chemists, pharmacists, among others do with the same purpose.

The best known and most used epidemiological models are compartmental models, which divide the population into compartments or classes, as seen in Figure 2.5, for each of the phases of
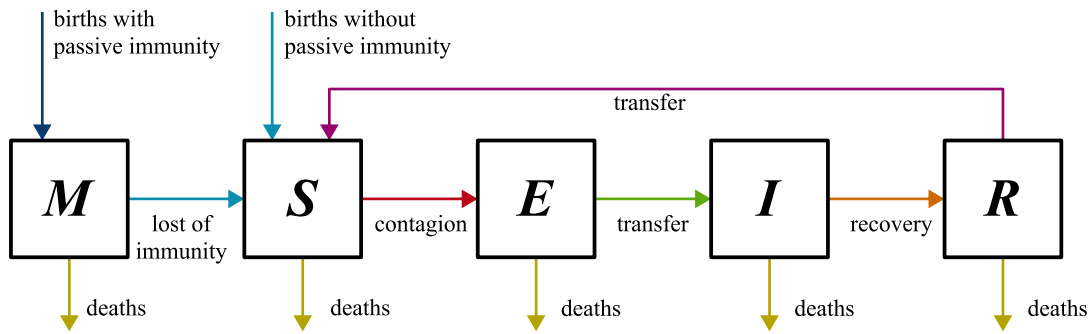
Figure 2.5: The transfer scheme for the general MSEIR model with the passively immune class *M*, the susceptible class *S*, the exposed class *E*, the infected class *I*, and the recovered class *R* [2].

the spread of the disease, and establish mechanisms for transfer between different groups, either through differential equations or probabilities.

The compartments that each model has will depend on the characteristics of the disease and the population, as well as the objectives or purposes of the model. In any case, they are still models that idealize the real dynamics of the disease. Thus, the best known models for fixed networks, without births or deaths, are:

1. Susceptible - Infected - Recovered (SIR), for long-term or permanent active immunity after illness.

2. Susceptible - Infected - Susceptible (SIS), for very short or no immunity after illness.

3. Susceptible - Exposed - Infected - Recovered (SEIR), when contagion leads to a stage before infection, but later immunity is permanent.

4. Susceptible - Exposed - Infected - Susceptible (SEIS), when contagion leads to a stage before infection, but later immunity is short or non-existent.

The factor on which epidemiological models depend is the *basic reproduction number*, R_0, which is defined as the average number of secondary infections produced by a single infected individual that is introduced into a population completely made up of susceptible individuals. For many of these models, infection can only begin when $R_0 > 1$ [2]. To calculate its value, we must focus on new infections that appear in the population. The transmission of the disease consists of a system of differential equations, with non-negative initial conditions x_0:

$$\dot{x}_i = f_i(x) = \mathscr{F}_i(x) - \mathscr{V}_i(x), \qquad i = 1, \ldots, n, \tag{2.1}$$

where $x = (x_1, x_2, \ldots, x_n)$ is the number or fraction of individuals in each compartment, $\mathscr{F}$ is the rate of appearance of new individuals in compartment *i*, and $\mathscr{V}$ is the rate of transition to other

compartments. We define the matrices

$$F = \left[\frac{\partial \mathscr{F}_i(x_0)}{\partial x_j} \right] \text{ and } V = \left[\frac{\partial \mathscr{V}_i(x_0)}{\partial x_j} \right] \qquad \text{for } 1 \leq i,\, j \leq m < n, \qquad (2.2)$$

where F is a non-negative matrix, while V is non-singular, so its inverse V^{-1} is non-negative. m is the total number of infection compartments (no susceptible or recovered). Let I_0 be the number of initially infected individuals.

Then $FV^{-1}I_0$ is a vector of the expected number of new infections. Thus, the multiplication FV^{-1} is the matrix of the next generation of the disease, and has input (i, j) equal to the expected value of secondary infections. Therefore, we can define the basic reproduction number as the spectral radius of the matrix FV^{-1} [22]:

$$\mathscr{R}_0 = \rho \left(FV^{-1} \right) \qquad (2.3)$$

2.3 Evolutionary game theory

The population dynamics of competition or cooperation between individuals is studied in the evolutionary game theory, as a framework of strategies and decisions that members of the population can make rationally or irrationally, depending on maximizing individual or collective benefit according to certain reward and penalty parameters for the entire population. These models represent biological dynamics, such as competition between bacterial strains [23] or cooperation between viruses [24], or social dynamics, such as cultural evolution, moral dilemmas or economic behaviors. In finite populations, evolutionary games function as stochastic processes, specifying the reward mechanism that governs the transmission of strategy between one individual and another [25].

2.3.1 Prisoner's dilemma

The best-known evolutionary game model arises from a traditional dilemma in which cooperation is analyzed as a social problem, where the reward for cooperating or not cooperating benefits or harms, respectively, the other individual, in such a way that between two individuals particularly the strategy of defecting is more beneficial than that of cooperating [26]. However, if both individuals do not cooperate, the reward is zero, bringing the problem to a paradox. This system is defined by a reward matrix between cooperators (C) and defectors (D),

$$\begin{array}{cc} & \begin{array}{cc} C & D \end{array} \\ \begin{array}{c} C \\ D \end{array} & \left[\begin{array}{cc} b - h & -h \\ b & 0 \end{array} \right] \end{array}, \qquad (2.4)$$

such that $b > b - h$ and $0 > -h$, therefore strategy D is dominant over strategy C.

12

2.3.2 Voluntary public goods games

The generalization of the prisoner's dilemma allows individuals to take other strategies that mean new rewards or penalties, depending on the strategy taken by the other [27]. The most common thing is that each strategy is dominant over another, meaning that there is no balanced or favorable state, as in the *Rock, Paper, Scissors* game, which follows the rules of the reward matrix

$$
\begin{array}{c c c c}
 & R & P & S \\
\begin{array}{c} R \\ P \\ S \end{array} & \left[\begin{array}{c c c} 0 & -s & 1 \\ 1 & 0 & -s \\ -s & 1 & 0 \end{array}\right], & &
\end{array}
\tag{2.5}
$$

In social models, these strategies can be cooperation (*C*), defection (*D*), or loneliness (*L*) or equidistance, and in general, there can be strategies that are totally dominant or dominant depending on certain conditions being met. Other models include many other strategies, such as punishment [28].

2.3.3 Strategy change probability

The probability of changing from one strategy to another is usually a function of the total reward received by the individual and by the rest of the individuals who have a connection or link with him. An example is a linear function of rewards, such as

$$
p = \frac{1}{1} + w\left(\frac{\pi_f - \pi_r}{\Delta\pi}\right).
\tag{2.6}
$$

Here, $0 \leq w \leq 1$ and determines the intensity of strategy selection; π_f and π_r are the payoffs of the focal and the model individuals, respectively, while $\Delta\pi$ is the maximum payoff difference. The focal is the node that compares its total payoff with the model's payoff to know if its strategy is better and has to be kept, or is worse and has to be changed.

Another commonly used probability function is the Fermi function from statistical mechanics,

$$
p = \frac{1}{1 + e^{w(\pi_f - \pi_r)}},
\tag{2.7}
$$

where if we replace w by k^{-1}, this new factor k can represent the irrationality of the focal individual or the population in general. $w \ll 1$ means a weak selection, and $w \to \infty$ leads to an imitation system [25].

2.4 Emergence of Cooperation

Cooperation in human beings, the Aristotle's *zoon politikon*, is a phenomenon that has served the species not only for survival, but also for developing and transmitting to future generations more

complex structures, such as religion, law, language, that have transformed nature and our own nature. Being a way to contribute to the well-being of other individuals, even for personal benefit, results solely from moral values and collective social norms. In the evolutionary game theory, the origin of these norms is set aside, leaving this study to sociology and anthropology, and it focuses on the emergence of this phenomenon in populations, despite being a strategy that is not the more favorable for the individual. The study of the emergence of cooperation despite the assumption of rational egoism, although human beings do not necessarily follow this logic, allows us to focus attention on why cooperation is a problem of interest. Likewise, it facilitates analysis between interdependent entities. On the other hand, with the more advanced development of model theory, it is possible to relax this assumption in favor of more realistic ones. Establishing social dilemmas as dynamic reward games where individual and collective rationalities come into conflict leads us to define cooperation as the rational strategy only when the other individuals in the community also cooperate [29].

Chapter 3

Network properties for the emergence of cooperation

3.1   Coupled evolutionary game-epidemic mechanism

In this Chapter, we present our implementation of a model with coexisting and interdependent dynamics of epidemic propagation and cooperation game. Compartmental models allow the biology of a contagious disease to be parameterized. However, many factors, such as the population's own actions, influence transmission and could be included in the model. In the same way, the behavior of a population is susceptible to changes due to the existence of a global concern, such as an epidemic. This creates a scenario of coexisting dynamics that evolve interdependently. The epidemiological model that we use in this work is the susceptible-infected-susceptible compartmental model (SIS), as a modification of the Kermack and McKendrick system [30] where the disease has no immunity or has one of very short duration. The population is divided into two classes, where the susceptible (S) can be affected by infected individuals (I) according to the contagion strength, β. The latter, upon overcoming the disease depending on the recovery rate, become susceptible again. This model is expressed in the form of differential equations:

$$\frac{\mathrm{d}S}{\mathrm{d}t} = -\beta \frac{SI}{N} + \gamma I, \tag{3.1}$$

$$\frac{\mathrm{d}I}{\mathrm{d}t} = \beta \frac{SI}{N} - \gamma I. \tag{3.2}$$

If we divide both equations by N, the size of the population, and substitute $\mathsf{s} = S/N$ and $\mathsf{i} = I/N$ for the fractions of susceptible and infected, respectively, the system of equations is equal to

$$\frac{\mathrm{d}\mathsf{s}}{\mathrm{d}t} = -\beta \, \mathsf{s}\mathsf{i} + \gamma \mathsf{i}, \tag{3.3}$$

$$\frac{\mathrm{d}\mathsf{i}}{\mathrm{d}t} = \beta \, \mathsf{s}\mathsf{i} - \gamma \mathsf{i}. \tag{3.4}$$

From this ODEs, we can define the basic reproduction factor, $\mathscr{R}_0 = \beta/\gamma$.

To apply the system in the network, the agents have one of the two possible states, and we use a stochastic model, where the probability of infection of a susceptible individual in the presence of one or more infected neighbors is a specific contagion strength $\hat{\beta}$, while recovery depends on its rate γ and the units of time elapsed since the beginning of the infection Δt [31]. Then,

$$p(S \to I) = \hat{\beta}, \tag{3.5}$$

$$p(I \to S) = \Delta t\, \gamma. \tag{3.6}$$

To couple social behavior with the epidemic, the contagion strength will be an exponential function of the fraction of cooperators, with the original β as the constant factor. Here we distinguish between a contagion with global influence, where we take into account the infected cooperators throughout the network, C_I, or with local influence, with just the cooperating neighbors, $\mathscr{C}_i$:

$$\text{with global influence: } \hat{\beta}(c) = \beta \exp\left(\frac{C_I}{N}\right) = \beta \exp\left(-c_I\right); \tag{3.7}$$

$$\text{with local influence: } \hat{\beta}(c) = \beta \exp\left(\frac{\mathscr{C}_i}{\mathscr{N}_i}\right) = \beta \exp\left(-c_i\right). \tag{3.8}$$

Meanwhile, dynamics between cooperators (C) and defectors (D) are determined by the traditional prisoner's dilemma (2.4), but modified to make it dependent on the development of the disease. Therefore, the reward matrix is:

$$A = \begin{array}{c} \\ C \\ D \end{array} \begin{array}{cc} C & D \\ \begin{bmatrix} b-h & -h \\ b-g(\sigma,I) & -g(\sigma,I) \end{bmatrix} \end{array}, \tag{3.9}$$

where, for simulations, the benefit $b = 1.5$, and the harm $h = 0.5$. Here we introduce $g(\sigma,I)$, defined as the individual's awareness function about the disease according to the current state of the epidemic and directly proportional to the fraction of infected nodes. Here we can distinguish two types of awareness, local or global. Knowledge about the disease and the individual's reaction may correspond to the number of infected neighbors around them, that is, they have local information about the disease. On the other hand, if the individual has knowledge about the total number of infected individuals in the population, we speak of global information about the disease. Thus,

$$\text{with global information: } g_g = \sigma \frac{I}{N} = \sigma\, i; \tag{3.10}$$

$$\text{with local information: } g_l = \sigma \frac{\mathscr{I}_i}{\mathscr{N}_i}, \tag{3.11}$$

where $\mathscr{I}_i$ is the number of infected among the neighbors $\mathscr{N}_i$ of node i, and $\sigma \in (0,1]$ is the infection awareness factor.

The probability of changing strategy is defined by the Fermi function. Each node in the network is taken, and their respective payoffs, π_i are calculated, according to the equation

$$\pi_i = \sum_{j \in \mathscr{N}_i} s_i A s_j, \tag{3.12}$$

where s_i is the unit vectors of the strategy of node i, and s_j the one of the j neighbor. For each focal player, a model neighbor is chosen, and their rewards are compared following the function

$$p_{i \to j} = \frac{1}{1 + e^{(\pi_i - \pi_j)/k}}, \tag{3.13}$$

where k is the player's irrationality, set for this case at 0.5. For simplicity, in our simulations, the change to infected state will occur when the probability $p_{i \to j}$ is grater than a threshold, which is set to 0.5.

3.1.1 Algorithms

Figure 3.1 shows our iterative algorithm of a replicator that defines the coexisting and co-evolving dynamics between the evolutionary game and the disease, depending on the parameters β and σ, and occurring during T units of time (days). This replicator has three subprocesses.

The first subprocess, in Figure 3.2, is the recovery of infected nodes depending on the days that have passed since the onset of the illness. The recovery is reached when the infection time elapsed is equal to $1/\gamma = 7$. The second subprocess, in Figure 3.3, defines the evolutionary game, in this case the Prisoner's dilemma iterative for each node, with local or global information about the disease. Finally, the algorithm in Figure 3.4 shows the disease propagation with local or global influence of cooperation.

We run the model one hundred times for each pair of parameters σ and $\mathscr{R}_0$ and each network studied. The code that we used for the simulations of this algorithms can be reviewed in the Appendix A.
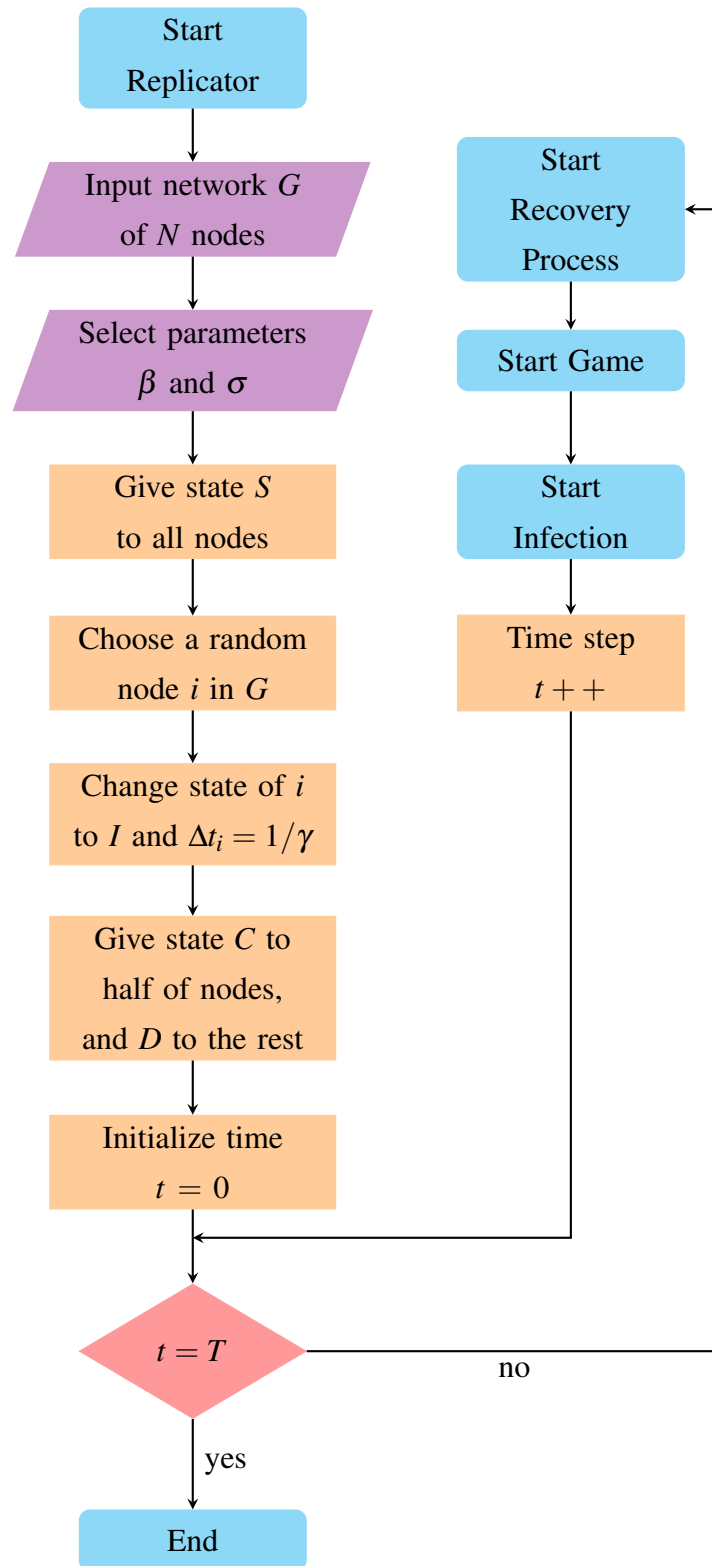
Figure 3.1: Flowchart of the replicator dynamics.
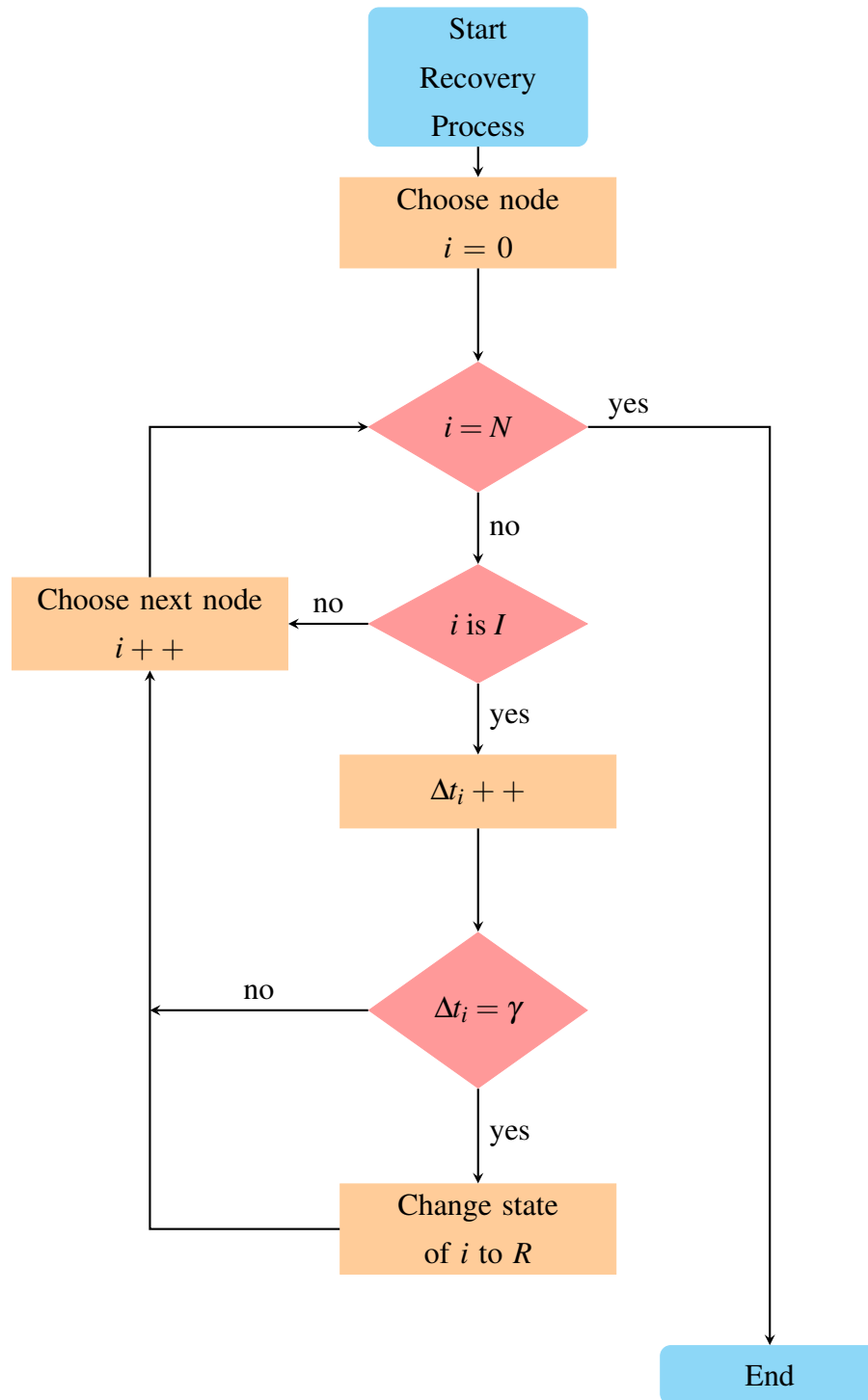
Figure 3.2: Flowchart of the recovery process.
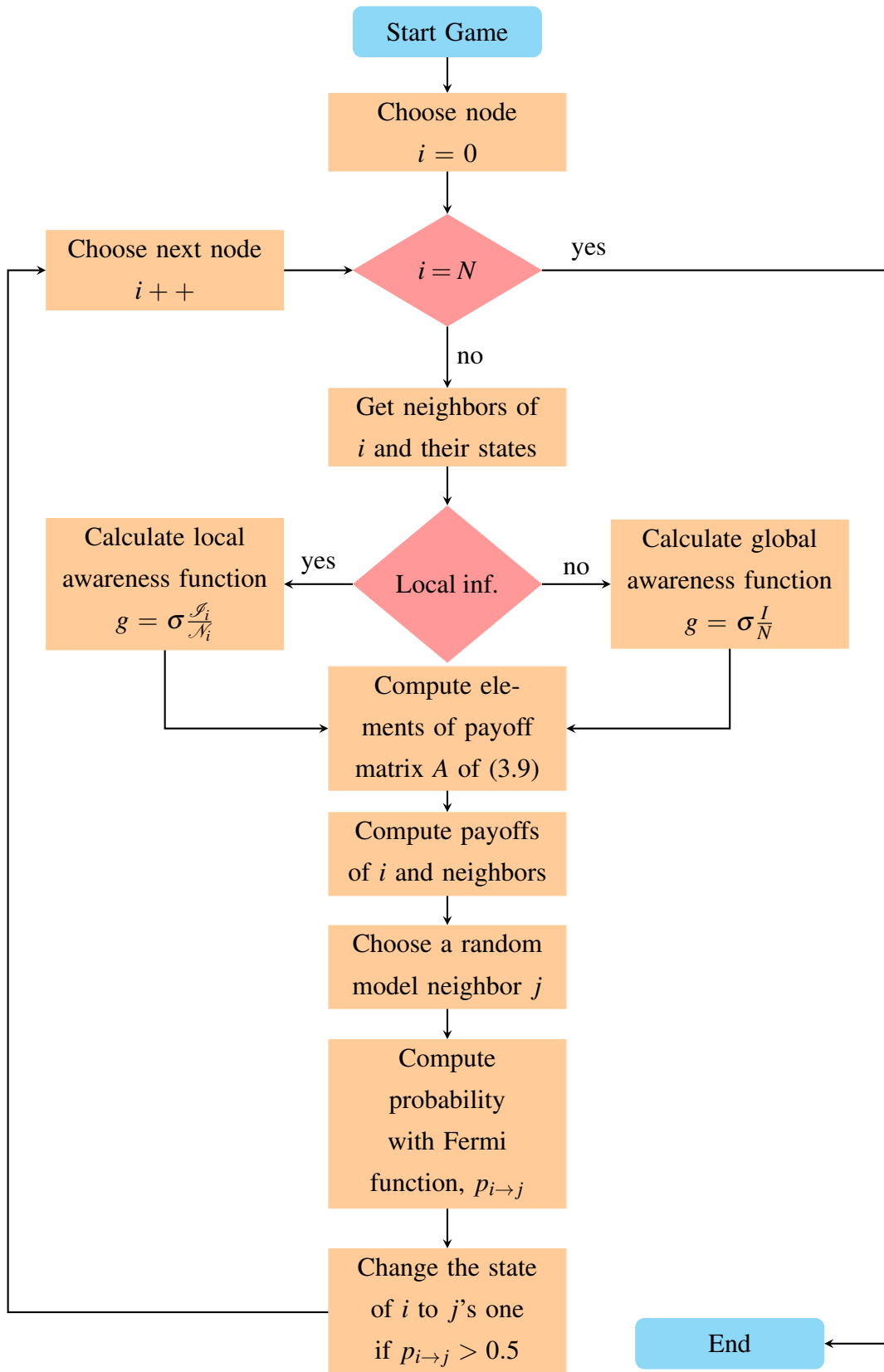
Figure 3.3: Flowchart of the evolutionary game.
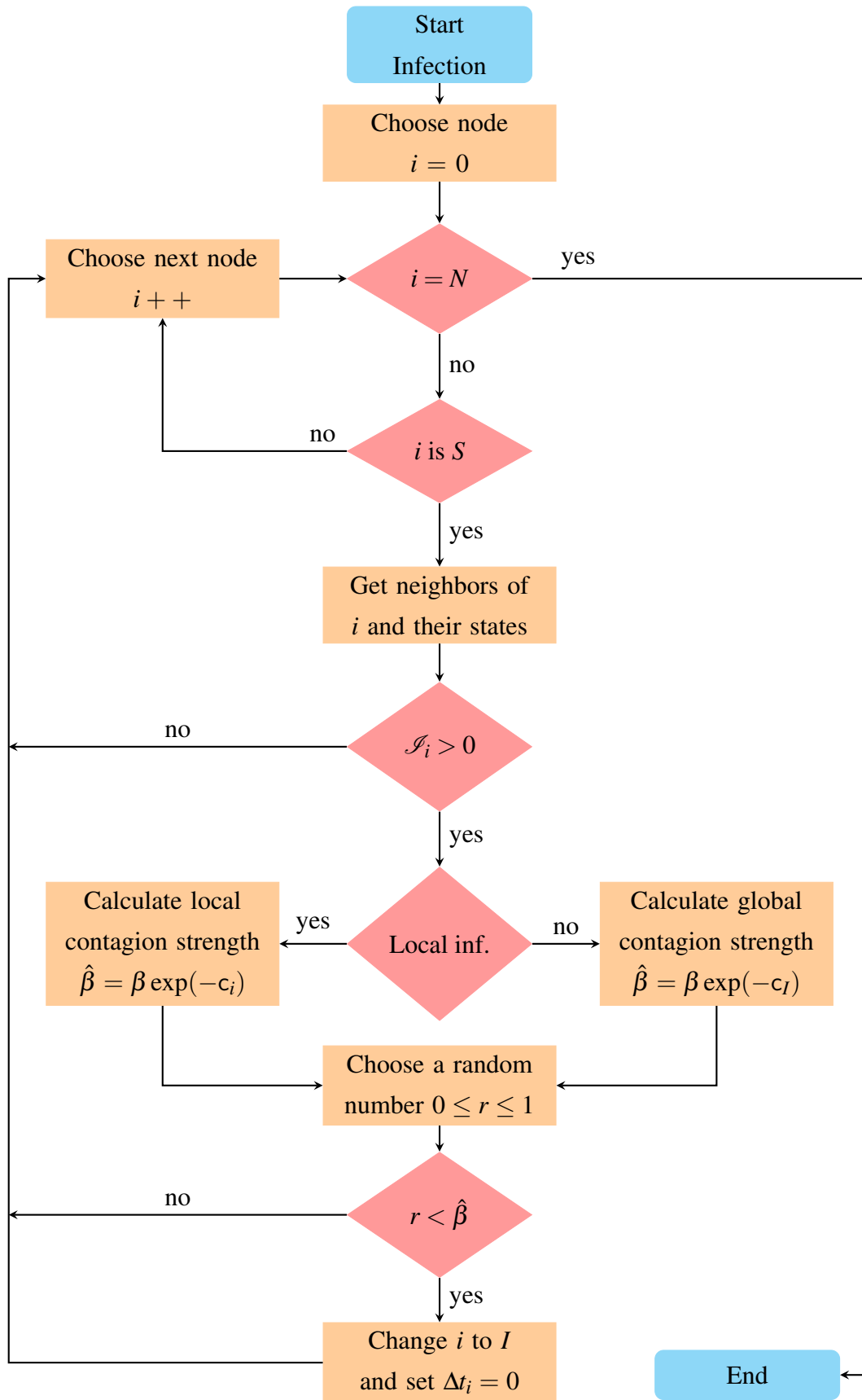
Figure 3.4: Flowchart of the disease propagation.

3.2 Cooperation in scale-free, small-world and grid networks: Characterization and comparison

3.2.1 Creation of the networks

The first network for our simulations is a scale-free, as in Figure 3.5a, built with the mechanisms of growth and preferential coupling. For each iteration, the probability of adding a node is 41%; the probability of adding an edge between two existing nodes is 54%; meanwhile, the cost of adding a new node connected to a random existing node is 5%. This scale-free network has a degree distribution that follows a power law, as seen in Figure 3.5b.

<div align="center">(a)   (b)</div>
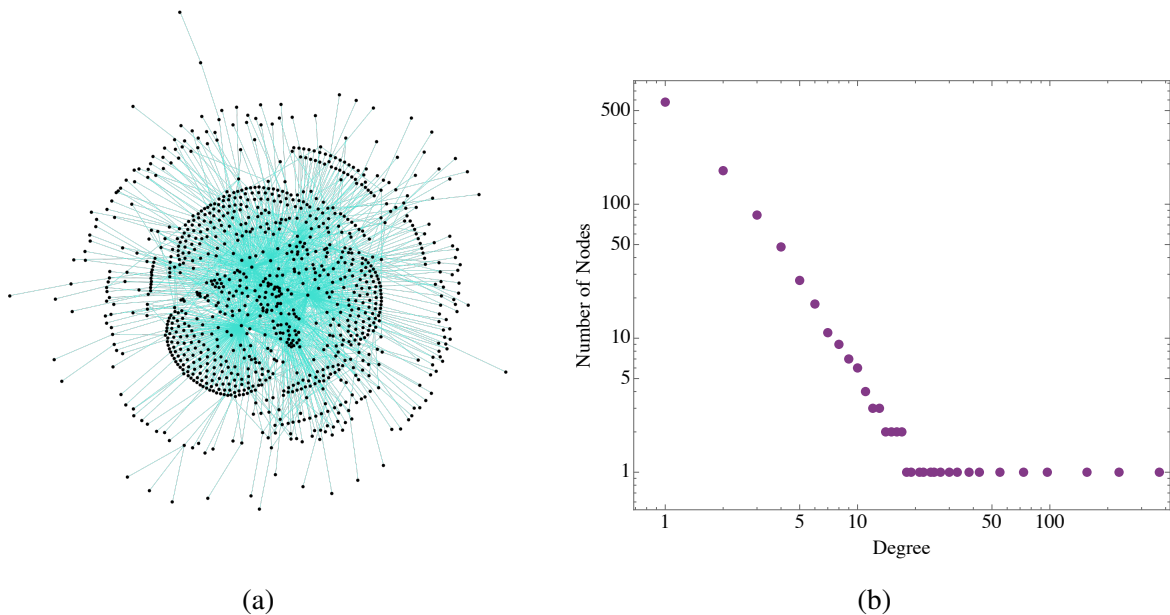
Figure 3.5: Illustration (a) and degree distribution (b) of the scale-free network.

The second type of graph that we employed is a random small world network, as in Figure 3.6a, following the parameters of the Watts-Strogatz model, from a ring where each node is joined with the 4 nearest neighbors, and 50% chance of randomly rewire each edge. Figure 3.6b shows the degree distribution of the small-world network.

The third network that we consider is a two-dimensional grid network of 32×32 nodes, shown in Figure 3.7, with the degree distribution, which is not complex as we can divide the nodes if their are inside (the mayority), in a border or in a corner (only four).
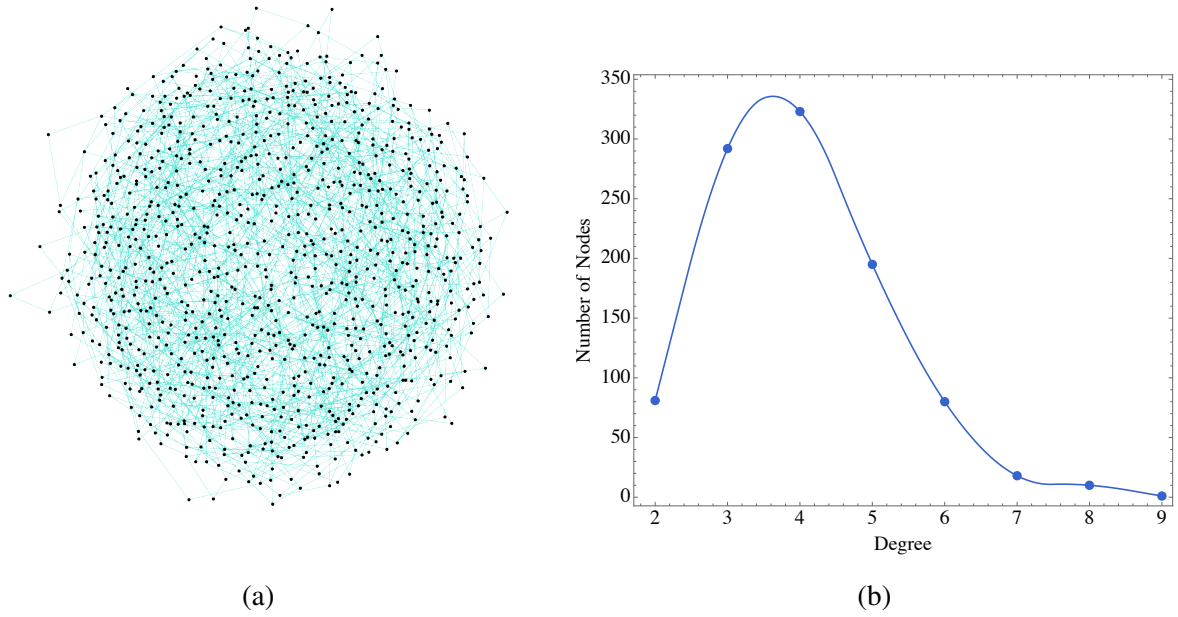
Figure 3.6: Illustration (a) and degree distribution (b) of the random small-world network.
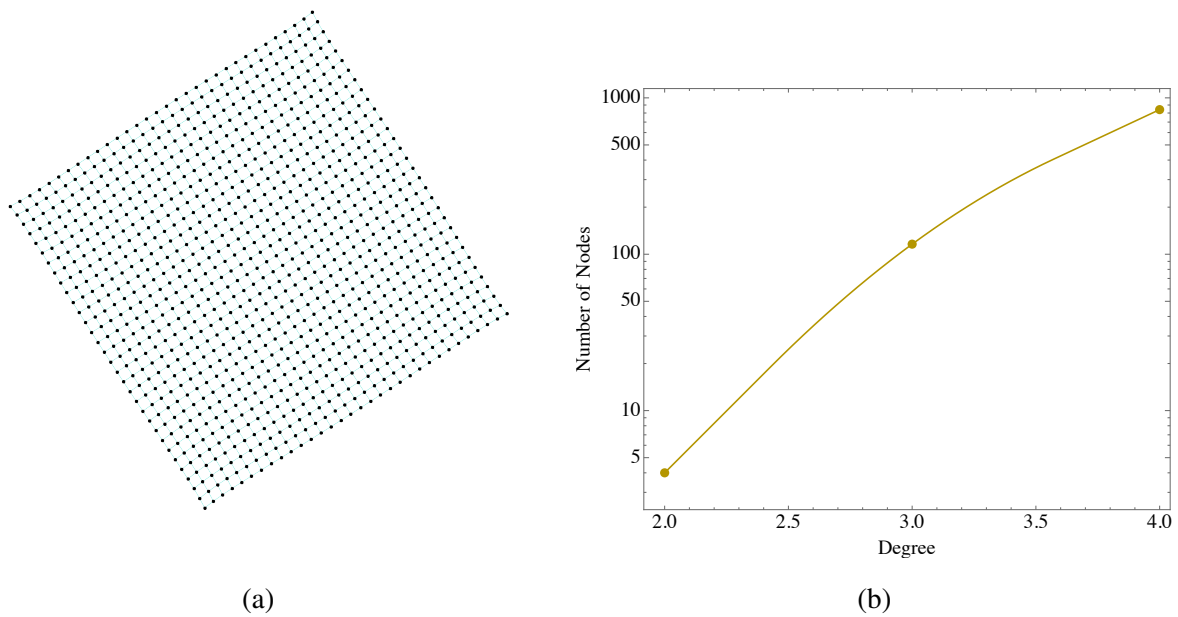
Figure 3.7: Illustration (a) and degree distribution (b) of the regular 32×32-grid network.

3.2.2 Infected and cooperators fractions with local and global information

As initial conditions of the network, both for the state of the disease and the strategies of the game, we establish that 50% of the population is cooperator. These initial conditions do not represent any distortion of the model, because the non-cooperative strategy is dominant. On the other hand, a

single recently infected individual is included, which represents 0.1% of the population. Everyone else is susceptible.

After the examination period, $T = 150$ days, we can say with complete certainty that the disease has reached the steady state, and we can count the number of infected and cooperators out of the total population. Depending on whether the information the nodes have is global or local, and for σ between 0 and 1 and R_0 between 0.7 an 7.0, we survey the fractions of cooperators and infected nodes for each type of network, as shown in Figure 3.8.

The cooperators fraction is almost the same for any value of σ and $\mathscr{R}_0$ in small-world and grid networks, with around 1 in 10 and 1 in 5, respectively. Meanwhile, the cooperation fraction in scale-free networks is high for a region where $\sigma > 0.5$ and $\mathscr{R}_0 > 2.1$.

On the other hand, the infected fraction in small-world and grid networks changes as expected without the evolutionary game dynamics, with high transmission for high reproduction $\mathscr{R}_0$ and vice versa. On the contrary, the infection fraction in scale-free networks is slightly affected by the cooperators, and in the *cooperation region*, the infected fraction is lower than awaited, $\mathrm{i} < 0.8$, when the mean cooperation fraction is over 0.2.

3.2.3   Disease and behavioral dynamics by time units

Although in the steady state we can examine the result of the coevolution of the two coupled dynamics, if we analyze the temporal progress of the cooperators and infected we also find important results.

The infection in the grid network, as seen in Figure 3.9c, reaches its stable state very quickly, in about a third of the time studied. Cooperation also reaches its final value a moment after the illness begins.

The disease in the small-world random network evolves very slowly, as shown in Figure 3.9b, and although it approaches the stable state, even at the end of the period, the peaks and valleys of the epidemic are noticeable. Cooperation, as in the grid network, also reaches its final value a moment after the disease begins.

In the scale-free network, the evolution of the fraction of infected, in the figure, is similar to that found in the small world network, but when there is a considerable percentage of cooperators in the network, proportionally and progressively, reduces the steady state value. Even more interesting is that the larger the cooperators fraction, the faster the stable point is reached, and the peaks and valleys of the infected fraction are not as prominent, as seen in Figure 3.9a (left).

In all cases, the cooperation decays to a minimum around 0.1 on the first days, because the infected fraction is too low and the payoff of cooperation is always unfavourable.

From these results, we can ensure that the topological properties of the scale-free network are producing the emergence of cooperation between subject-nodes for certain parameters. These
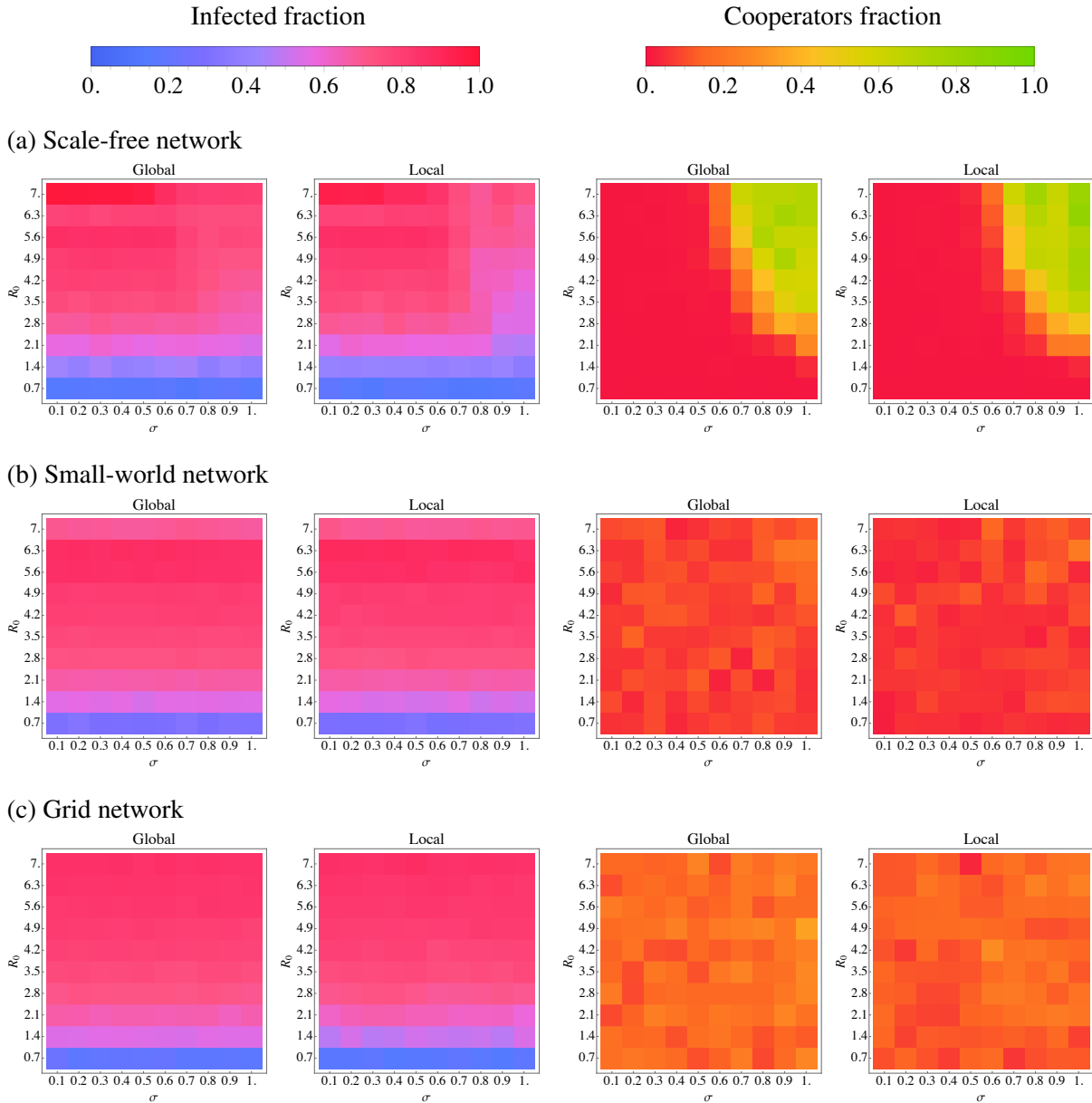
Figure 3.8: State at $T = 150$ days of infected fraction, and cooperators fraction in population of the three networks, by the basic reproduction number $\mathscr{R}_0$ (*y*-axis), and awareness σ (*x*-axis), for global (left) and local (right) information and influence.

properties should not be present in small-world random networks or grid networks.

3.2.4 Analysis of the dynamical evolution in communities of scale-free networks

We investigate the behavior of communities in the scale-free network to find how the coupled dynamics evolve between nodes that are more connected to each other than to the rest of the network.

To do this, we use the Louvain method to detect communities [32]. This community partitioning method seeks to optimize modularity, $Q \in [-0.5, 1]$, which is a value that compares the density of edges inside a community and outside it. To do this, two steps are followed:

On the first step, each node starts in its own community, and then moves each node, i towards the community C, of each of the neighbors, and we calculate the modularity gain,

$$\Delta Q = \frac{k_{i,\text{in}}}{2N} - \phi \frac{\Sigma \cdot k_i}{2N^2} \tag{3.14}$$

where N is the number of nodes; $k_{i,\text{in}}$ is the number of links from i to the nodes in C; k_i is the degree of node i; Σ is the total number of edges incident on nodes of C and ϕ is the resolution. In the case of weighted networks, instead of the number of edges, it is treated with the sum of their respective weights.

The largest positive ΔQ is chosen, and node i joins the community that corresponds to this value. If, on the other hand, it is negative, the node remains in the community where it was before.

This process is iterative and ends when no individual move improves modularity (or is less than a defined threshold), i.e., it converges.

On the second phase, a weighted network is built where each node corresponds to a community of the original network according to the results of the first phase. The edges between two nodes have the weight of the total sum of links between the two communities they represent. Then the phase one procedure is applied again and thus find larger communities, increasing modularity.

In one scale-free network, we use this method and take the three biggest communities to investigate the temporal evolution of social behavior and disease, as seen in Figure 3.10.

(a) Scale-free network
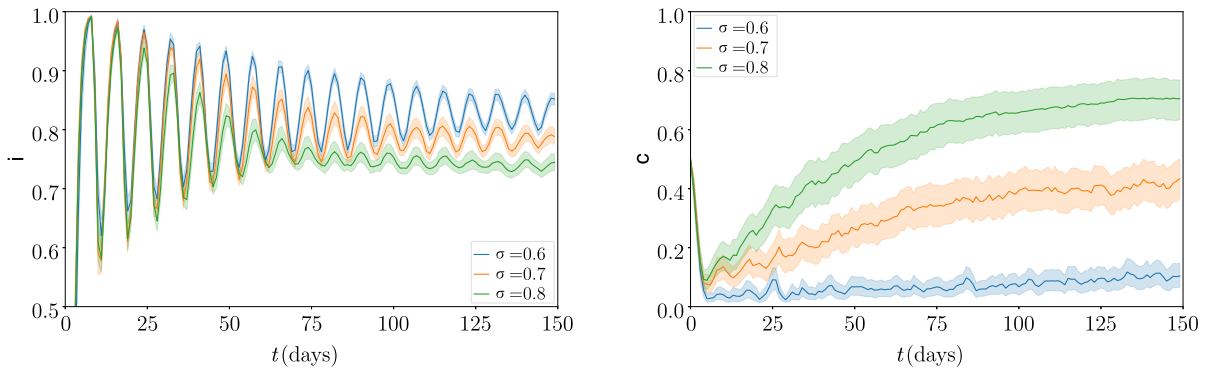
(b) Small-world network
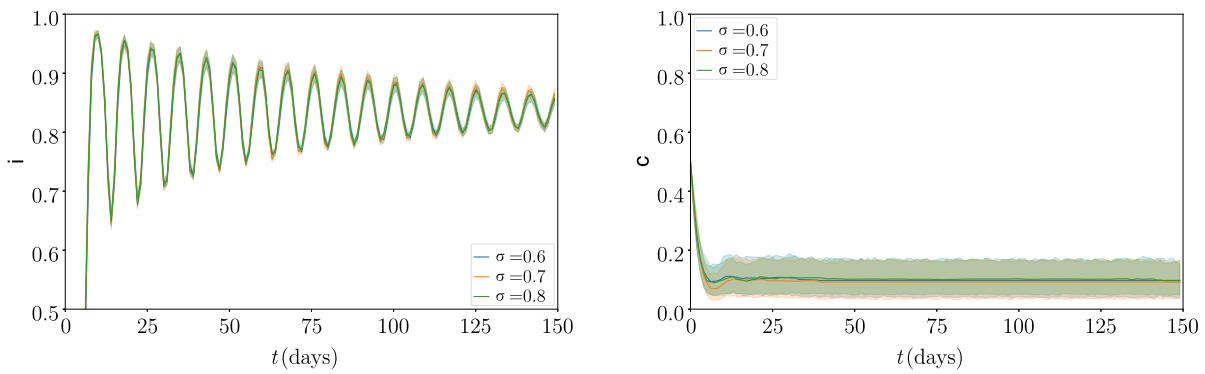
(c) Grid network
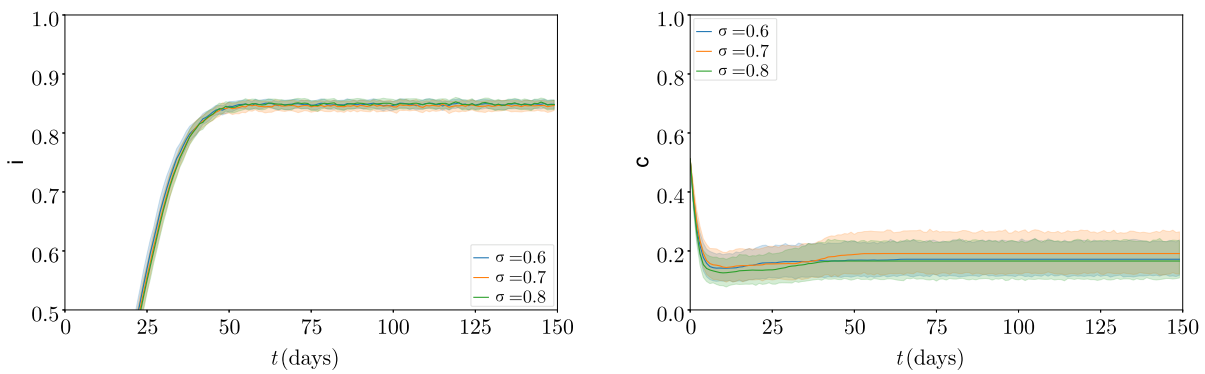
Figure 3.9: Temporal evolution for different awareness $\sigma \in \{0.6, 0.7, 0.8\}$, and a basic reproduction number $\mathscr{R}_0 = 5.6$ of each network. The lines represents the mean of 100 simulations with 95% confidence interval. Left column shows the evolution of the infected fraction, while right corresponds to cooperators fraction. The interactions between nodes respond to the global influence and information of cooperation and disease respectively.

(a) Global information.
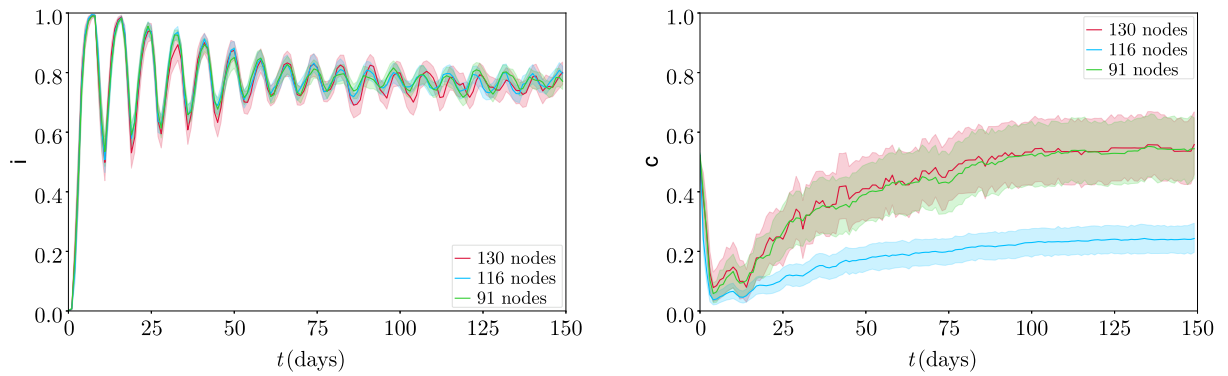
(b) Local information.
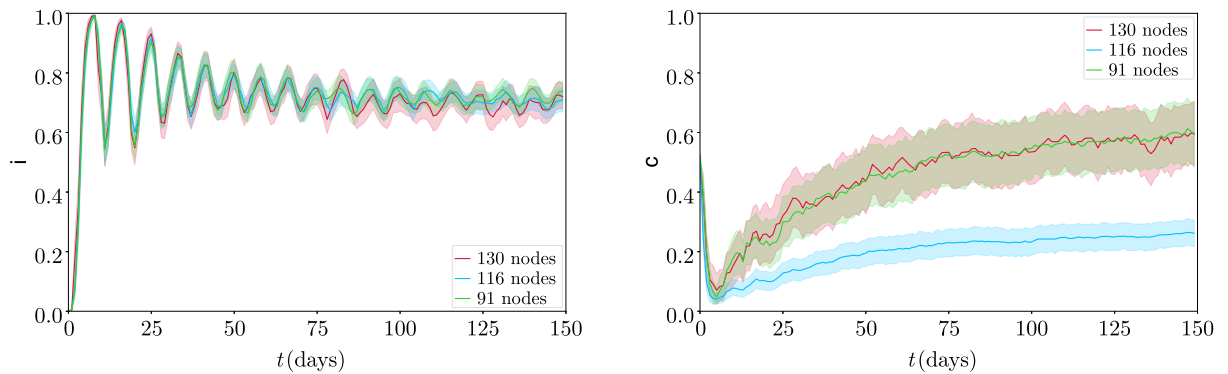
Figure 3.10: Temporal evolution for awareness $\sigma = 0.8$ and basic reproduction $\mathscr{R}_0 = 5.6$ of the three biggest communities in a scale-free network, for global (a) and local (b) influence/information. Left column shows the evolution of the infected fraction, while right corresponds to cooperators fraction. The lines represents the mean of 80 simulations with different initial conditions, with 95% confidence interval.

In these three communities studied, the spread of the disease is essentially equivalent. However, in scale-free networks, there are cases of communities where the fraction of cooperators is especially low relative to average, which means that, at least in part, the cooperation strategy emerges in its own way depending on the state conditions of the respective network region, in relation to the heterogeneous distribution of nodes.

In small-world and the grid graph the the study of communities doesn't have special results as their distribution is homogeneous and the evolution is the same for all regions in the graph.

3.2.5   Characterization of networks

As the structure of the complex network seems to be responsible for the emergence of cooperation, we investigate some properties of each of these networks for comparison. We create one lattice network, 50 scale-free networks and 50 small-world networks, following the parameters of section 3.2.1, as samples to compute some features that can explain the behavior dynamics: average shortest path length, efficiency, clustering coefficient, assortativity and average degree. A summary of the mean values of the studied properties for each network is in Table 3.1.

Average shortest path length

Figure 3.11: Box-and-whisker plot of average shortest path lengths of scale-free networks and small-world graphs. Gray line represents the median, the dot shows the mean.

We compute shortest path lengths between all pair of nodes in the graph. As the graphs' edges are not weighted or directed, we just need to find the path $\mathfrak{p}_{i \to n} = (i, j, \ldots, n)$ with **fewest** elements of adjacent vertices between the nodes i and n, and consider one less than the length of this path, $d_{in} = \dim(\mathfrak{p}_{i \to n}) - 1$, corresponding to the number of edges between them. We calculate the average of these shortest lengths,

$$\hat{d} = \frac{1}{N(N-1)} \sum_{\substack{i,j \in V \\ i \neq j}} d_{ij}, \tag{3.15}$$

where N is the total number of nodes in the graph [33]. Figure 3.11 shows quartiles and mean

of shortest path lengths between scale-free and small-world networks, while for the lattice graph, $\hat{d} = 20.6$.

We can notice a substantial difference between the networks. The scale-free network has a very low characteristic length, promoted by highly connected nodes. Short paths through the network facilitate the transfer of information at higher speeds, and this can be noted in the time it takes to reach the peak of the disease. Then, the high fraction of infected nodes can favour the payoff of cooperation strategy on the first stages of the disease. However, it is not enough to explain the phenomenon of cooperation, as it is a behavior that depends on the general strategy of all neighbors, not a contagion or simple transfer of state.

Global and local efficiency

(a)   (b)
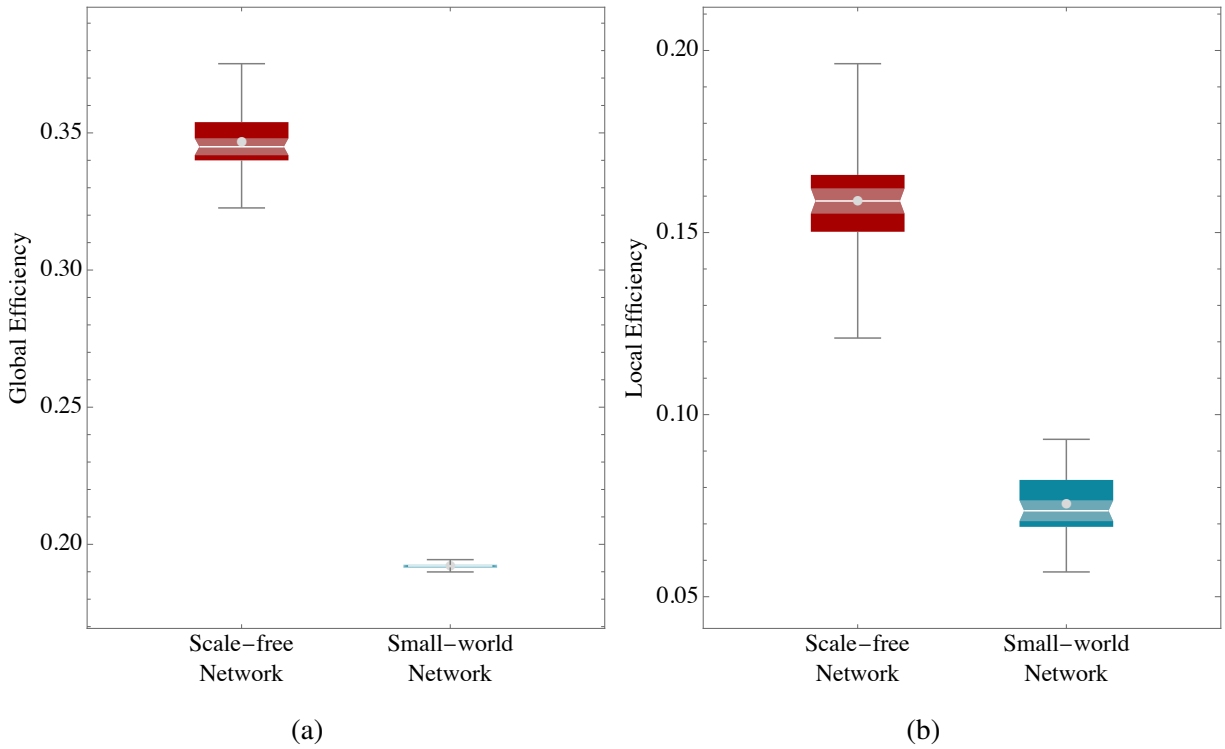
Figure 3.12: Box-and-whisker plot of global (a) and local (b) efficiency of scale-free and small-world networks. Gray line represents the median, the dot shows the mean.

The efficiency of a pair of nodes in a graph is the multiplicative inverse of the shortest path distance between the nodes [34]. The global efficiency is the average efficiency of all pairs of nodes,

$$\mathscr{E}_{\mathrm{g}} = \frac{1}{N(N-1)} \sum_{\substack{i,j \in V \\ i \neq j}} \frac{1}{d_{ij}}. \tag{3.16}$$

The local efficiency of a node i is the average efficiency induced by the local sub-graph of

the neighbors of this node [34], $\mathscr{E}_{ig}$. The local efficiency of the graph is the average of the local efficiency of all nodes,

$$\mathscr{E}_1 = \frac{1}{N} \sum_{i \in V} \mathscr{E}_{ig}. \tag{3.17}$$

We calculate these for each network for quartiles and mean comparison of Figure 3.12. Moreover, the global efficiency of grid network is $\mathscr{E}_g = 0.07$, while the local efficiency is null. The efficiency is a more directed measure of the speed of transfer of information. Scale-free networks are almost twice more efficient than small-world networks, and 4.6 times more than the grid graph.

Clustering coefficient
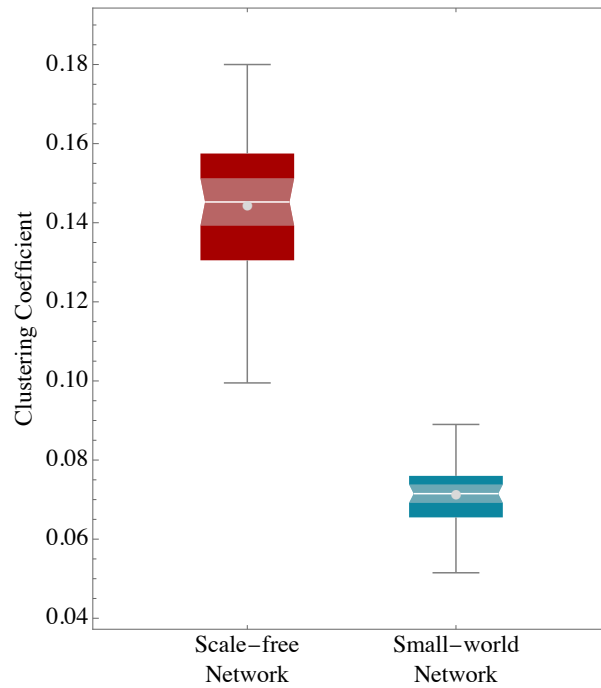
Figure 3.13: Box-and-whisker plot of clustering coefficients of scale-free networks and small-world graphs. Gray line represents the median, the dot shows the mean.

The clustering coefficient $\mathscr{X}$ of a node i is the number of triangles (subgraph of three connected nodes) through this node, $\mathscr{T}_i$, normalized by the total possible number of triangles,

$$\mathscr{X}_i = \frac{2 \mathscr{T}_i}{\deg(i)\left[\deg(i) - 1\right]}, \tag{3.18}$$

where $\deg(i)$ is the degree of the node i [35]. The average clustering is computed over the total number of nodes, N, so

$$\hat{\mathscr{X}} = \frac{1}{N} \sum_{i \in V} \mathscr{X}_i \tag{3.19}$$

In real-world social networks, nodes tend to build groups with high density of ties. The clustering coefficient is a measure of how clustered on average the nodes are in the network. The

scale-free network, with $\mathscr{X} \approx 0.14$, as seen in Figure 3.13, is closer to a real social network than random small-world networks, and thus has more groups in its structure. This is important for the emergence of cooperation, since the strategy is taken according to the nodes with which the two evaluated nodes, focal and model, have contact. This can also explain that in different communities on the network the cooperation fraction may vary. Nevertheless, it cannot explain why small-world network has a little less cooperation than grid networks.

Assortativity coefficient

Assortative mixing measures the similarity of connections in the graph with respect to the node degree. It can be characterized by the fraction of edges connecting vertices from of different types or communities in the network. We define the assortativity coefficient, to the level of assortative mixing in a network, on undirected networks thus:

$$\mathscr{A} = \frac{\sum_q e_q - \sum_q a_q^2}{1 - \sum_q a_q^2}, \qquad \mathscr{A} \in [-1, 1]; \tag{3.20}$$

where e_q is the fraction of edges that connects two vertices of type q, and a_q is the fraction of edges that ends in a vertex of type q [36]. Sums go for each type. We calculate this coefficient for each network, and the comparison is in Figure 3.14. In addition, grid networks has $\mathscr{A} = 0.64$.

The negative value of the coefficient in scale-free networks is due to the preference of highly connected nodes to link with slightly connected nodes, which are the two types of nodes that exist in this class of network. Grid networks have a coefficient close to zero because almost all nodes are connected to nodes of the same type, edge nodes or interior nodes. The small world network, due to its random construction, has an assortativity close to zero.

Let us remember that the strategy is dependent on the total balance of the rewards given by the strategy between a node and each of its neighbors. In general, in scale-free networks, loosely connected nodes will only be compared to the highly connected node with which they share an edge. This can benefit the cooperating nodes of the "periphery", allowing them to change the strategy of highly connected nodes that are defectors, or to maintain their strategy over time, especially for higher awareness. And on the other hand, they would always be at a disadvantage when awareness is low. This would make the nodes very willing not to cooperate in these circumstances.
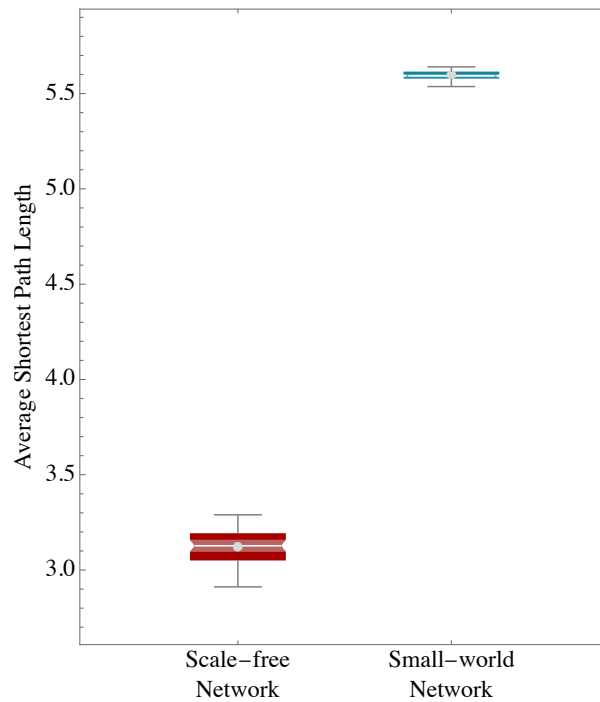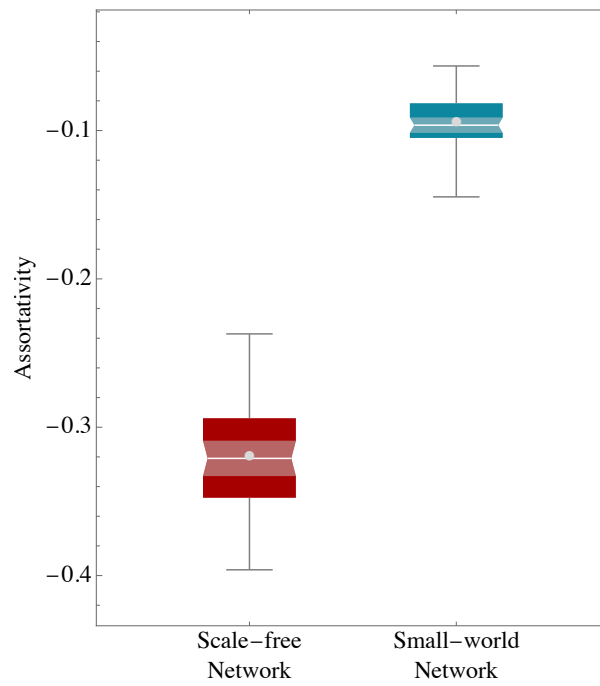
Figure 3.14: Box-and-whisker plot of assortativity of scale-free networks and small-world graphs. Gray line represents the median, the dot shows the mean.
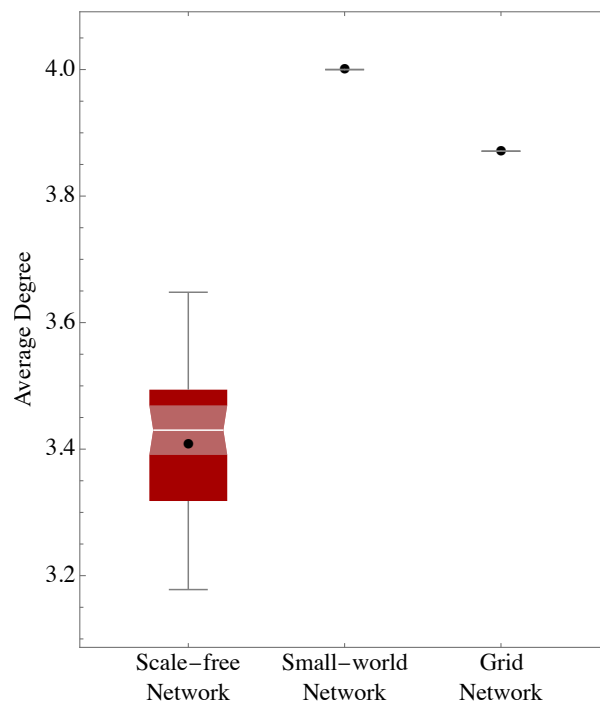

Figure 3.15: Box-and-whisker plot of average degree of scale-free networks, small-world graphs and grid. Gray line represents the median, the dot shows the mean.

Average Degree

Finally, it was necessary to compare the average degree between the networks, taking into account, of course, that the degree distribution is very heterogeneous in the case of scale-free networks. On average, as shown in Figure 3.15, $\hat{\kappa}$ of small-world networks are higher, for the given rewiring characteristics, than those of grid networks. The low average degree of scale-free networks is due to the predominance of loosely connected nodes.

Summary

In general, scale-free networks have all the properties that allow them to develop a significant fraction of cooperation between their nodes in cases of high awareness: high efficiency and information transmission, grouping and negative assortativity.

The other two networks have some advantage in certain properties, but not in all, and at the same time with less strength relative to scale-free networks.

Network	Av. Shortest Path Length $\hat{d}$	Clustering Coefficient $\hat{\mathscr{X}}$	Assortativity $\mathscr{A}$	Average Degree $\hat{\kappa}$	Global Efficiency $\mathscr{E}_{\mathrm{g}}$	Local Efficiency $\mathscr{E}_{\mathrm{l}}$
Scale-free	3.119	0.144	-0.3197	3.407	0.346	0.1585
Small-world	5.597	0.071	-0.0945	4.0	0.192	0.0753
Grid	20.667	0.0	0.6399	3.871	0.074	0.0

Table 3.1: Mean values of evaluated properties of the networks.

3.3   Rewiring scale-free networks to disappear the cooperation region

The goal now is to find the properties that cause the appearance of the cooperation region in the scale-free network for certain β and σ by converting one network in another and calculate the evolution of the cooperation region and the properties by percentage of rewiring.

We start with a scale-free network, and then we randomly remove edges between two nodes arbitrarily chosen. Then, we create a new edge between one of the two separated nodes to another, again, random node in the network. Finally, in case that the network becomes disconnected, we also create edges to join the isolated groups to the entire network. Visually, we can notice the destruction of the network clustering as a greater number of links between nodes are modified, as seen in the Figure 3.16.

| Scale-free network | 50% of rewiring | 100% of rewiring |
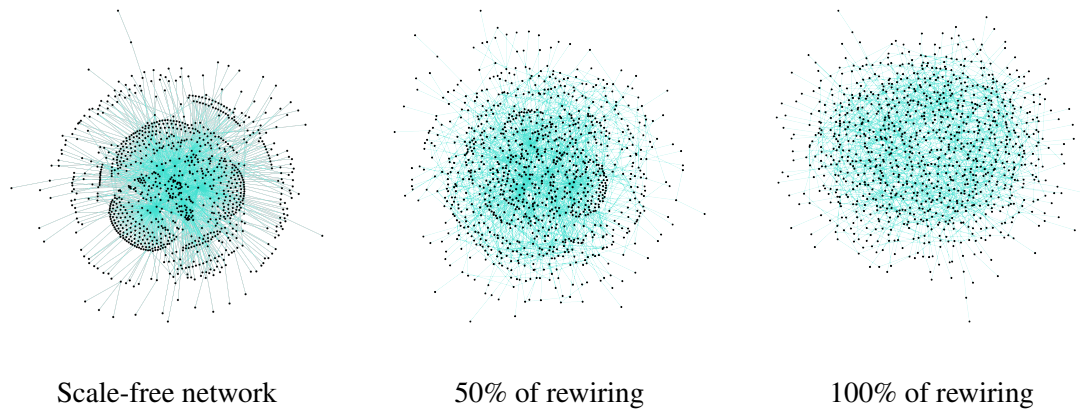
Figure 3.16: Illustration of the change made in scale-free network structure.

Without fear of losing important information and for visualization reasons, we decided to choose a specific awareness value, $\sigma = 0.8$, for the entire range of basic reproduction numbers of the disease, and see the resulting fraction of cooperators after 150 days for each network with a certain percentage of random rewiring. We do the same, but for a specific value of basic reproduction, $\mathscr{R}_0 = 0.8$, and the entire awareness range. The resulting cooperation fraction by percentage of rewiring for local-information simulations is presented in Figure 3.17. Global-information ones has similar results.

It is notable that when faced with random rewiring in the scale-free network, the fraction of cooperation declines for high values of awareness or the strength of the disease, remaining very close to 0 for low values. However, it is only possible to observe this effect from a rewiring greater than 50%, with a peak of no-cooperation in 95% of rewiring. Furthermore, when all edges are removed by new ones (100% rewiring) we see a relevant increase in the total number of cooperators and for all values. Finally, there is a peak in the cooperation fraction when random rewiring reaches between 15% and 40% of the total links.

From the degree distribution of the scale-free network, we know that most connections between nodes begin or end in the very loosely connected nodes. Therefore, it is expected that with a low percentage of rewiring, the network architecture will change decisively. And that is why the region of cooperation only decreases when more than half of the edges have been replaced by new ones.

Likewise, when the network reaches full rewiring, the expected results should be similar to those of the random network. However, in these results the fraction of cooperation is higher, comparing the cooperation plots of Figure 3.8b, since in this new case, the cooperation achieved is around half.

Therefore, we compute the properties' values of the rewired networks to compare them with the original scale-free network and the random small-world network, in Figure 3.18.
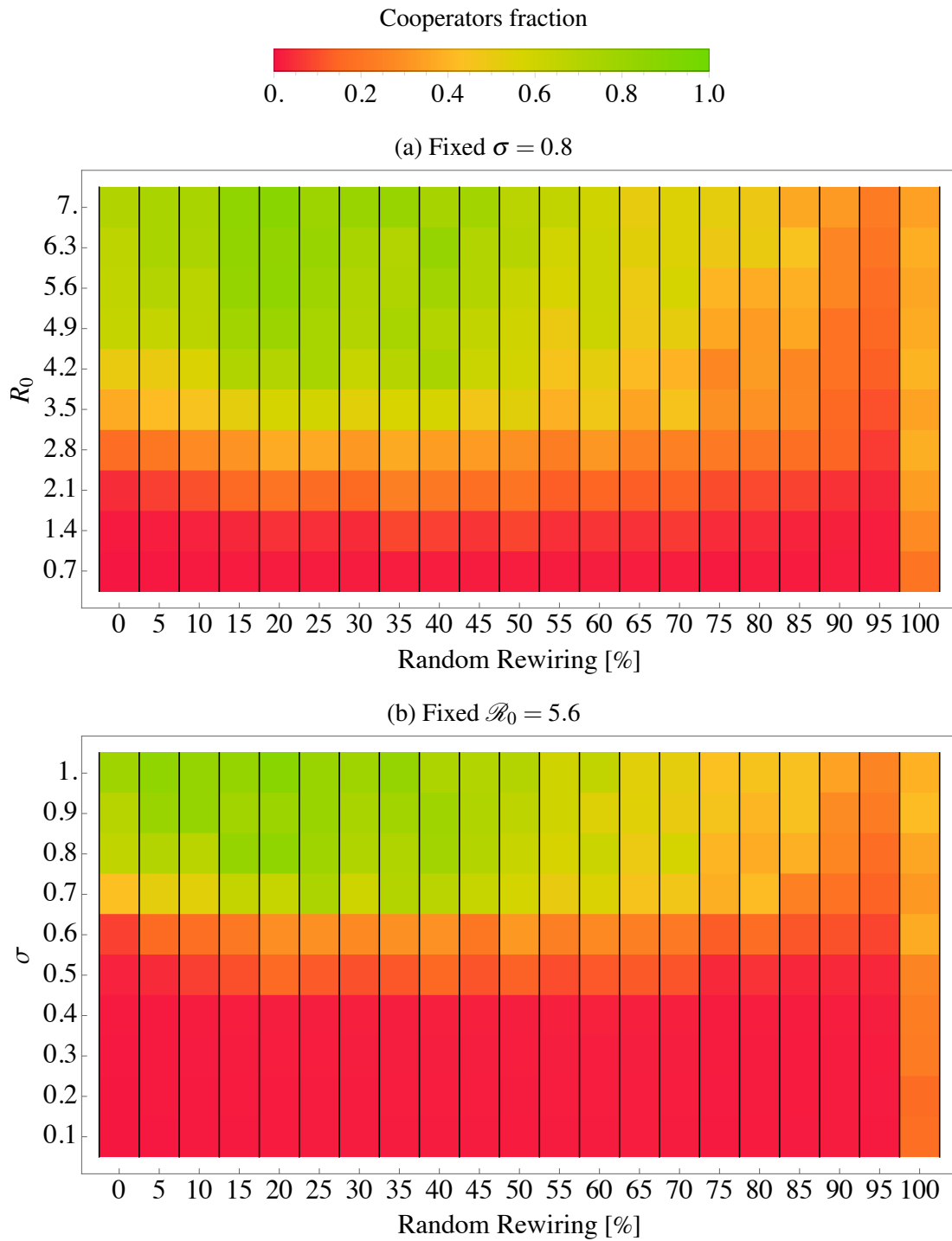
Figure 3.17: Cooperation fraction for a complete range of awareness (a) or basic reproduction (b) values, by percentage of random rewiring. Results with local influence and information of 100 simulations with 3 networks.
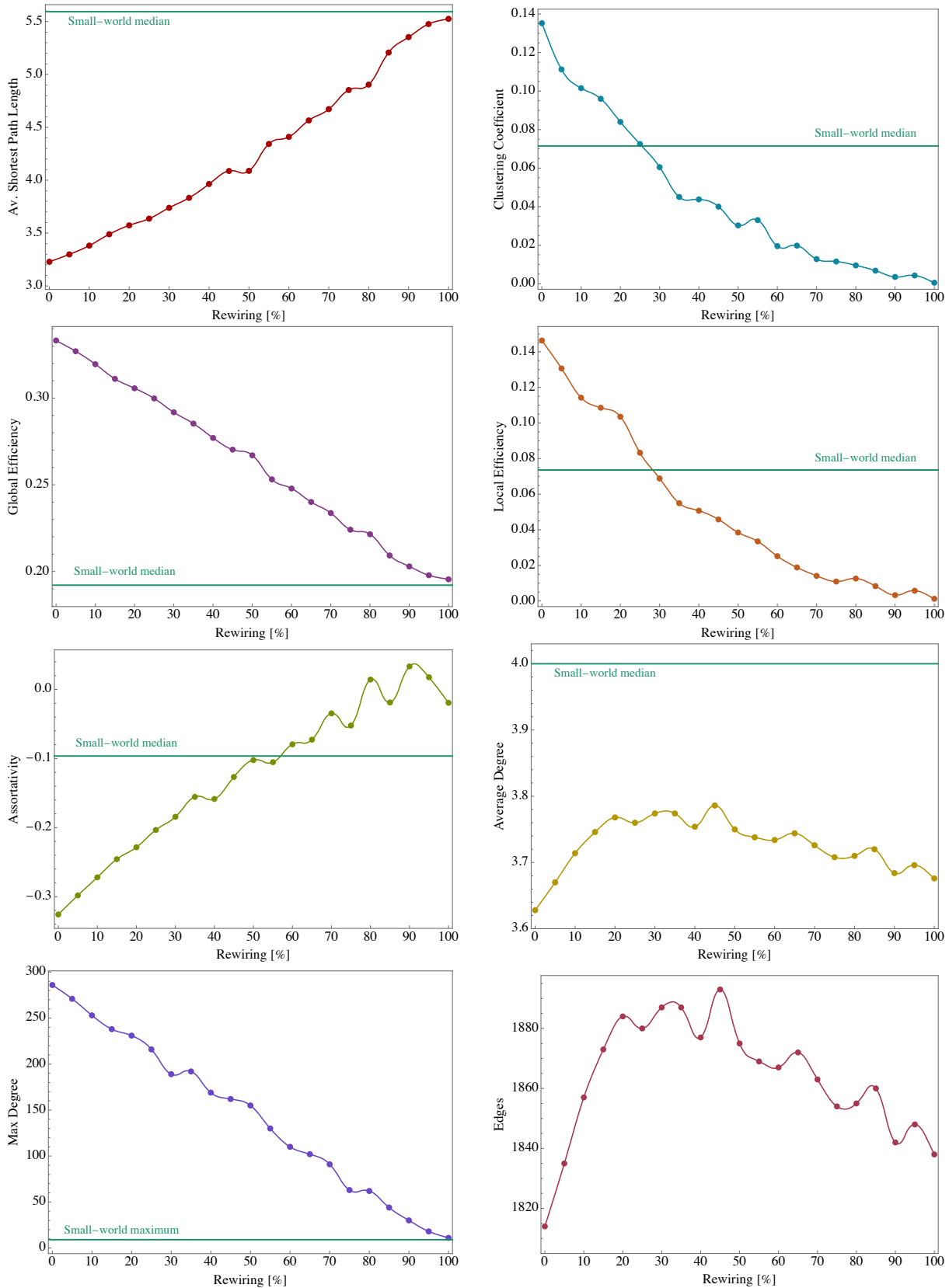
Figure 3.18: Properties of 1000-node rewired scale-free networks by rewiring percentage. Turquoise line represents the small-world median value. Bottom right plot shows the number of edges.

For the average shortest path length, the value of the small world network is approximately reached when all edges have been reconnected. The same is observed for the overall efficiency of the network and the maximum degree of a node.

However, other properties reach the value of the random network in partial rewiring. For example, the clustering coefficient and local efficiency reaches the median value of the small world network with 30% rewiring. Above all, the assortativity reaches the value of that of a small world, between 50% and 60% of rewiring. These values resemble the point at which the cooperation fraction begins to disappear. Negative assortativity would, therefore, be a sine qua non condition and the fraction of cooperation is high. This means that cooperation results from connections between very popular nodes with their very little connected nodes. It is also necessary to add that the maximum peaks of cooperation are reached when the rewiring over the total edges is between 15% and 20%. The number of total edges, as well as the average degree of the network, peaks at these rewiring values at the same time that the clustering coefficient is greater than in small-world networks. That is, cooperation increases the more connected the network is, as long as the other properties for cooperation are met.

3.4 Creating the cooperation region from a ring network to a globally coupled network
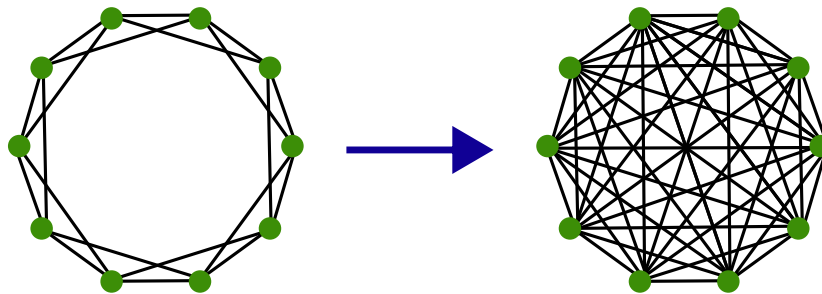
Figure 3.19: From ring-topology network to a globally coupled network.

We also investigated how to achieve high cooperation fractions for certain ranges of awareness and strength of the disease. We assume that the more connected the network, the more likely the emergence of cooperation, or the higher the total fraction of cooperators. For that, we start with a ring network, where each node is connected to two neighbors. Then we add new links with the next closest neighbors. This continues until all nodes are connected to all the others, that is, a globally coupled network, as seen in Figure 3.19. At this limit, dynamics always develop with the global influence of cooperation or global information about the disease. Since we want to see

how cooperation changes, we will add links and calculate the fraction of cooperation in each case keeping $\mathscr{R}_0 = 0.8$ constant.
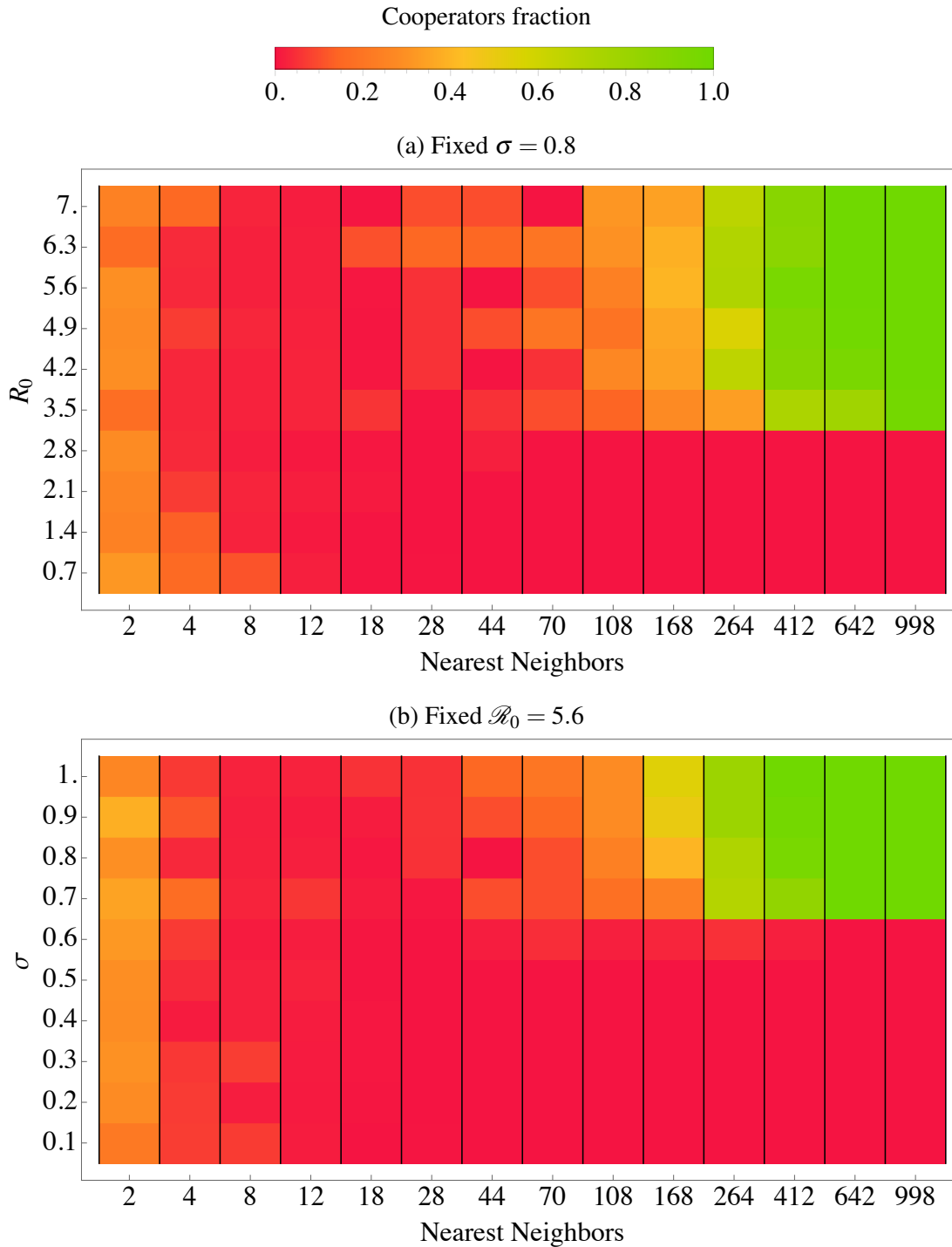
Figure 3.20: Cooperation fraction from a ring to a globally coupled networks with fixed (a) $\sigma = 0.8$ or (b) $\mathscr{R}_0 = 5.7$. Exponential adding of edges to nearest neighbors and 100 simulations for each parameter and network.

When there are only two neighbors, the resulting cooperation fraction is around 0.30 for any value. Adding a few more neighbors brings cooperation to a minimum, around 0.1. Then, as more neighbors are added, the cooperation region gradually appears. When global coupling has been achieved between more than half of the nodes, cooperation reaches its maximum (all nodes) for high values of awareness or strength of the disease, where all nodes cooperate. This shows that cooperation is an emergent phenomenon when for a certain number of nodes in the network there are many edges connected between them and the others. If these nodes are not cooperating, for high awareness or reproduction number values, the payoff they have would be very negative, according to the reward matrix used, and the cooperating nodes that have a higher payoff would prevail. That is, cooperation appears when there are a large number of nodes with which to calculate the rewards of the prisoner's dilemma.

This also explains the emergence of cooperation in the scale-free network. When it reaches the minimum in the first days, before the first peak of the disease, the slightly connected nodes that cooperate, even if there are very few left, will tend to maintain that state, because the highly connected nodes that do not cooperate would have a very high payoff for high values of the awareness function, $g(\sigma, I)$. So, it would be a matter of time so that, when compared, the cooperators prevail and make the strategy of a good part of the defectors change. Cooperation does not arrive completely because slightly connected nodes that do not cooperate could also have an advantage and maintain the defection state. On the other hand, in small-world networks the low number of neighbors does not allow the reward of cooperators to be truly higher than that of defectors. Therefore it is not possible for cooperation to emerge. When the network reaches the minimum number of cooperators, the cooperators find no advantage, but no disadvantage either. That is why the resulting small fraction of nodes is stable over time.

Chapter 4

Emergence of cooperation with other compartmental models and evolutionary game dynamics

Knowing the topological characteristics that allow a high fraction of cooperation, our next objective in this Chapter is to ensure that cooperation emerges in a scale-free network, but using other compartmental or social behavior models.

4.1 Cooperation emergence in SIR model

For this case, the population is divided into three classes, we follow the model from Kermack and McKendrick [30], where the susceptible (S) can be affected by infected individuals (I) according to the contagion strength, β. The latter, upon overcoming the disease depending on the recovery rate, become recovered (R). This model is expressed in the form of differential equations:

$$\frac{\mathrm{d}S}{\mathrm{d}t} = -\beta \frac{SI}{N}, \tag{4.1}$$

$$\frac{\mathrm{d}I}{\mathrm{d}t} = \beta \frac{SI}{N} - \gamma I. \tag{4.2}$$

$$\frac{\mathrm{d}R}{\mathrm{d}t} = \gamma I \tag{4.3}$$

From this ODEs, we can define the basic reproduction factor, $\mathscr{R}_0 = \beta/\gamma$, which is identical to that of the SIS case.

To apply the system in the network, the agents have one of the three possible states, and we use a stochastic model, where the probability of infection of a susceptible individual in the presence of one or more infected neighbors is a specific contagion strength $\hat{\beta}$, while recovery depends on its rate γ and the units of time elapsed since the beginning of the infection Δ_t. Susceptible and

41

recovered states are disconnected, and there is no possibility to change between this two. Then,

$$p(S \to I) = \hat{\beta}, \tag{4.4}$$

$$p(I \to R) = \Delta t \, \gamma. \tag{4.5}$$

The prisoner's dilemma continues to act in the same way, just as the values of the reward matrix are maintained between each pair of strategies,

$$A = \begin{array}{c} \\ C \\ D \end{array} \begin{array}{cc} C & D \\ \begin{bmatrix} 1.0 & -0.5 \\ 1.5 - g(\sigma, I) & -g(\sigma, I) \end{bmatrix} \end{array}. \tag{4.6}$$

In the same way, the awareness function, $g(\sigma, I)$ of (3.10) and (3.11), and therefore the influence of behavior on the disease, and the information about the infected, are kept in evolutionary game, both global and local. The change in strategy follows the Fermi function of (3.13).

4.1.1 Algorithms

The iterative replicator is the same for the disease propagation and the evolutionary game. However, there is a change in the recovery dynamics, as the infected nodes fall into the R state after the infection time. This change is shown in flowchart of Figure 4.1, and the code used is in Appendix A. We also run this model one hundred times for each pair of parameters σ and $\mathscr{R}_0$.

4.1.2 Disease and behavioral dynamics by time units

As initial conditions of the network, both for the state of the disease and the strategies of the game, we establish that 50% of the population is cooperator. On the other hand, a single recently infected individual is included, which represents 0.1% of the population. Everyone else is initially susceptible. For obvious reasons, no one has the status of recovered, as the disease is recently introduced.

Firstly, we use an estimated recovery time of 7 days, the same as for the SIS model. It is expected that the steady state will be reached very soon, where all or almost all nodes are recovered from the disease. Just a couple of weeks after the spread of the disease began, especially when the strength of the disease is high. In this studied range, $\mathscr{R}_0 \in [0.7, 7.0]$, as seen in the Figure 4.2, for both global and local information about the infected nodes in the network, cooperation only has a peak a couple of days after the number of infected reaches its maximum, for any level of awareness. Immediately the cooperators begin to decrease until their number is very close to zero.
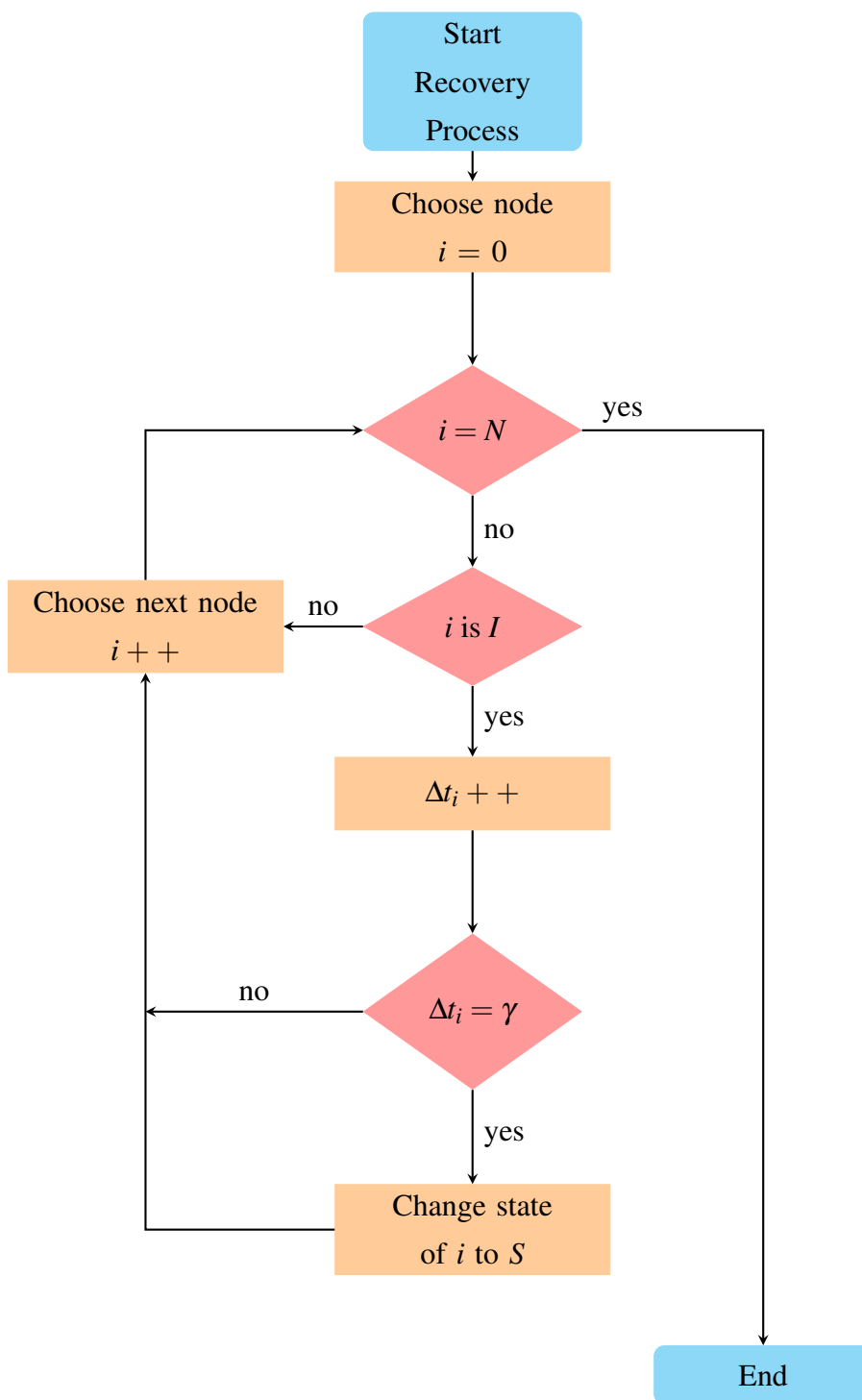
Figure 4.1: Flowchart of the recovery process in SIR model.

To obtain higher cooperation, it will be necessary to increase the time expected for the recovery of individuals, and for the infected quality to have more time to take effect in the payoff that governs the change in strategy or behavior. If we divide the reproduction factor by two, that is, $\gamma = 1/14$, we can notice an increase in the fraction of cooperators after the peak of the disease, as well as the persistence of a large part of these cooperators once the disease is over, as we can see in Figure

4.3. That is to say, there is a part of the population that keeps cooperating even though there is no longer an epidemic, and only the dynamics of the evolutionary game continue, although this fraction is not very high, between 0.3 and 0.6 (more for local than global information) with a very high level of awareness, $\sigma = 1.0$.

(a) Global information
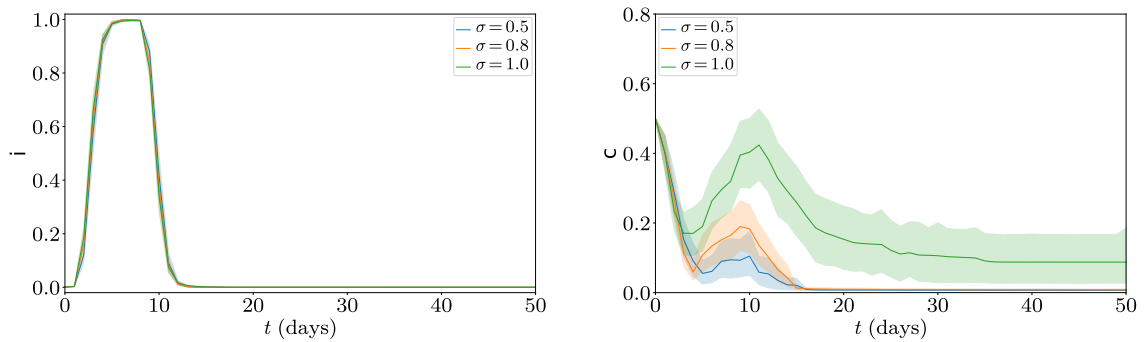
(b) Local information
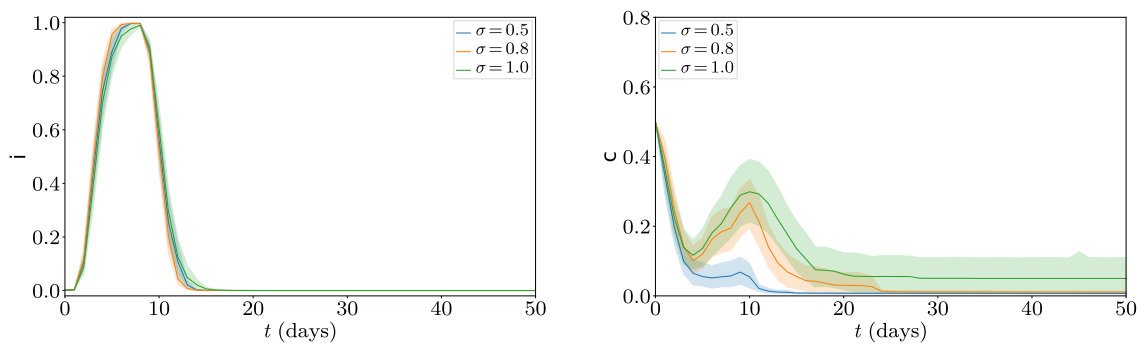
Figure 4.2: Temporal evolution (50 days) for different awareness $\sigma \in \{0.5, 0.8, 1.0\}$, and a high basic reproduction number $\mathscr{R}_0 = 7.0$, when the recovery factor $\gamma = 1/7$. The lines represents the mean of 100 simulations with 95% confidence interval. Left column shows the evolution of the infected fraction, while right corresponds to cooperators fraction. The interactions between nodes respond to the global (a) or local (b) influence and information of cooperation and disease respectively.

44

(a) Global information
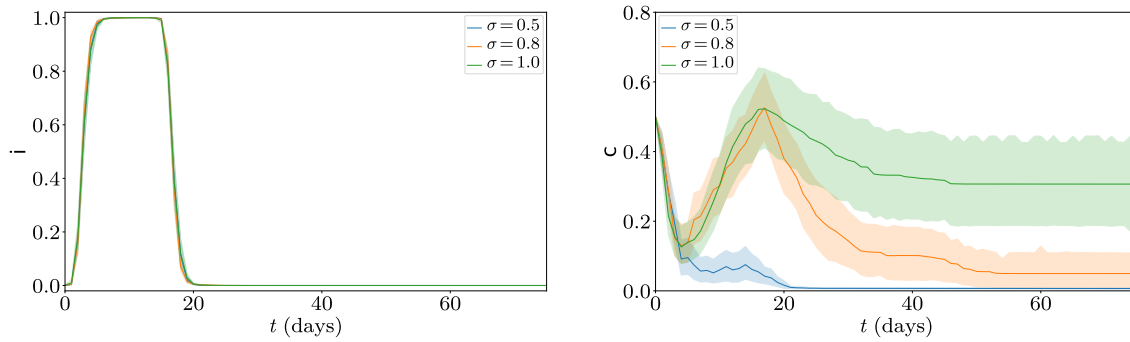
(b) Local information
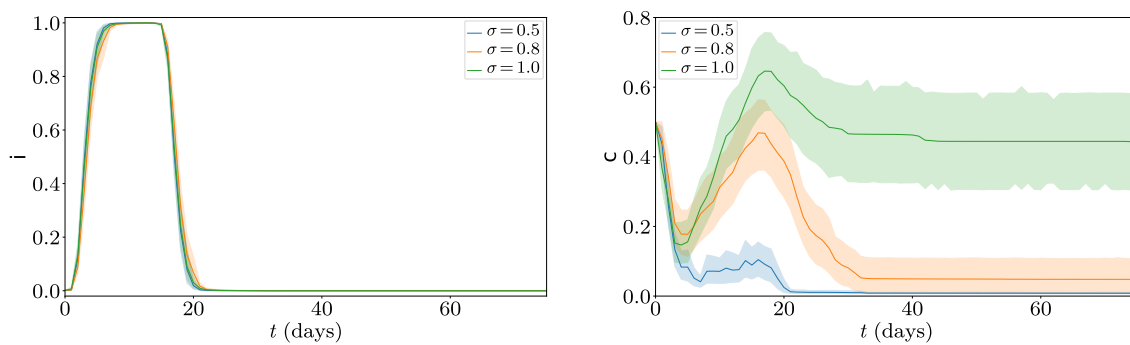
Figure 4.3: Temporal evolution (75 days) for different awareness $\sigma \in \{0.5, 0.8, 1.0\}$, and a high basic reproduction number $\mathscr{R}_0 = 14.0$, when the recovery factor $\gamma = 1/14$. The lines represents the mean of 100 simulations with 95% confidence interval. Left column shows the evolution of the infected fraction, while right corresponds to cooperators fraction. The interactions between nodes respond to the global (a) or local (b) influence and information of cooperation and disease respectively.

4.1.3 Infected and cooperators fraction at their peaks, with local and global information

The peak of cooperation in the scale-free network is reached a few days after the peak of the disease. If we compare the cooperation achieved when $\gamma = 1/7$ and when $\gamma = 1/14$, as shown in Figure 4.4, we can say that cooperation also depends on the parameters of recovery from the disease, and not only its propagation strength. Thus, in the SIS model, cooperation was achieved for basic reproduction numbers greater than 2.1, comparable to that of diseases such as SARS

(2-4) [37], COVID-19 (2.9-9.5) [38], or smallpox (3.5-6.0) [39]. Meanwhile, the cooperation fractions in the SIR model require a high $\mathscr{R}_0$, such as mumps (10-12) [40], or measles (12-18) [41]. This comparison is made ignoring the compartmental transmission model that is more accurate to describe each of them, but is relevant because inside those parameters, high cooperation will be found. However, cooperation has no real effect on the development of the disease.
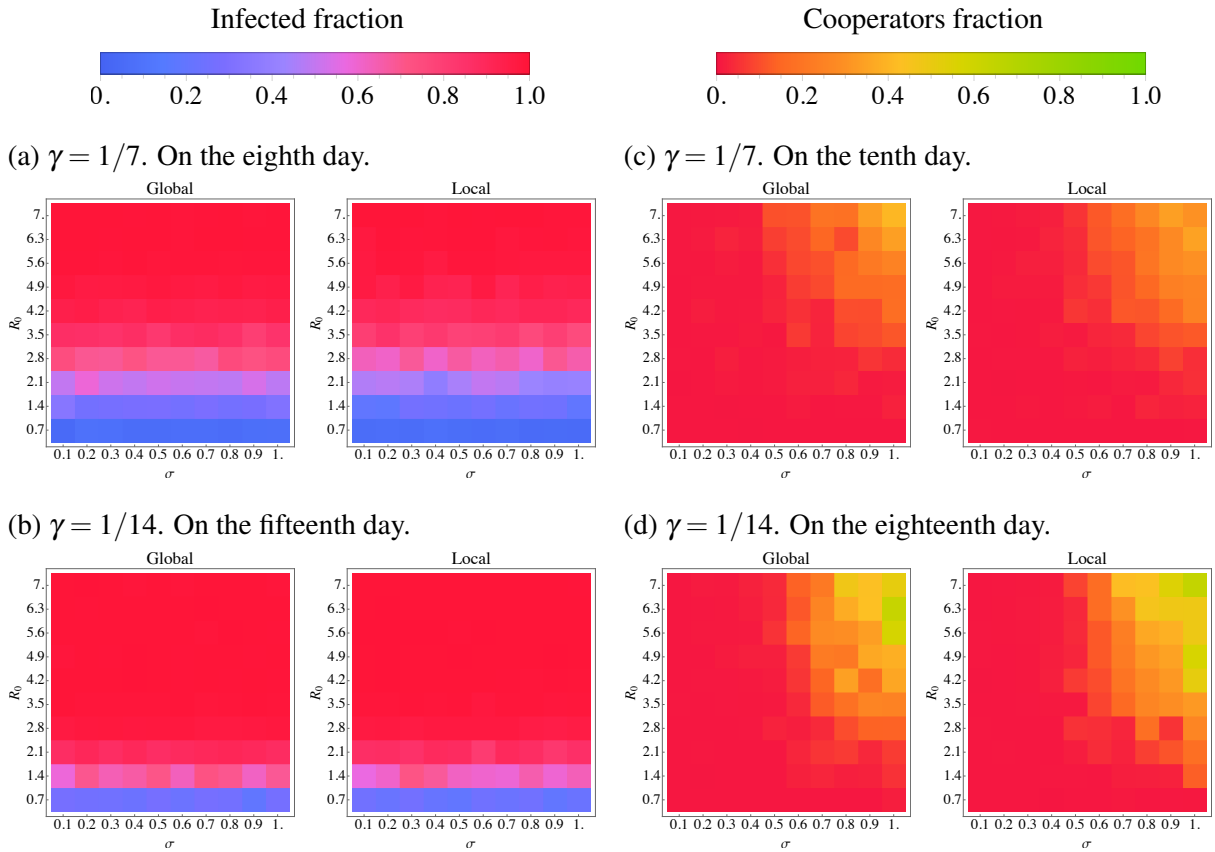
Figure 4.4: State of infected fraction, and cooperators fraction in population of scale-free network, by the basic reproduction number $\mathscr{R}_0$ (*y*-axis), and awareness σ (*x*-axis), for global (left) and local (right) information and influence. Plots represents the infected or cooperators fraction at their peak day, (a) and (c) for $\gamma = 1/7$, while (c) and (d) for $\gamma = 1/14$

4.2 Cooperation emergence in SEIS model

The incidence of the disease due to the correlation between susceptible individuals with infected individuals plays an important role in the dynamics of the disease. In most models, the incidence follows a bilinear relation, that is, as the product of susceptible elements with infected elements, βSI. However, this incidence function is not the most appropriate to describe diseases that spread mainly in humans.

For this SEIS model, including the exposed state, we replace the bilinear incidence with a nonlinear function of saturated incidence rate, following the study of Capasso and Serio on the cholera epidemic in Bari of 1973 [42], which includes an inhibiting factor due to the crowding caused by high numbers of cases. We will consider this epidemic model with general incidence rates that the diseases, such as COVID-19, can be infected in the latent/exposed period and the infected period:

$$\frac{dS}{dt} = -(f_I + f_E)S + \gamma I, \tag{4.7}$$

$$\frac{dE}{dt} = (f_I + f_E)S - \lambda E, \tag{4.8}$$

$$\frac{dI}{dt} = \lambda E - \gamma I. \tag{4.9}$$

Then, we define the nonlinear functions of saturated incidence respect to the number of exposed individuals or [43],

$$f_E = \frac{\beta_E E}{1 + \alpha_1 E}, \tag{4.10}$$

$$f_I = \frac{\beta_I I}{1 + \alpha_2 I}, \tag{4.11}$$

where β_I and β_E are the strengths of infection according to respective state the transmitter agent. For the examinations, we define $\beta_I = 2\beta_E = \beta$, supposing a greater strength when the individual is infected than exposed. Also, we establish $\alpha_1 = 0.2$ and $\alpha_1 = 0.1$ as the inhibitor factors for exposed and infected population. In our simulations, E and I of f_E and f_I correspond to the fraction of neighbors that have these states

From this ODEs, we can define the basic reproduction factor as

$$\mathscr{R}_0 = \frac{\lambda \beta + \frac{\gamma \beta}{2}}{\lambda \gamma} = \frac{\beta}{\gamma} + \frac{\beta}{2\lambda} = \beta \left(\frac{2\lambda + \gamma}{2\gamma \lambda} \right). \tag{4.12}$$

To apply the system in the network, the agents have one of the three possible states, and we use a stochastic model, where the probability of infection of a susceptible individual in the presence of one or more infected neighbors is the sum of the values given by the functions f_E and f_I, the exposition-to-infection change depends on the time of exposition Δt_E and the latency factor λ, while recovery depends on its rate γ and the units of time elapsed since the beginning of the infection Δt_I. Susceptible and recovered states are disconnected, and there is no possibility to

change between this two. Then,

$$p(S \to E) = \frac{\hat{\beta}I}{1 + \alpha_1 I} + \frac{\hat{\beta}E}{2(1 + \alpha_2 E)}, \tag{4.13}$$

$$p(E \to I) = \Delta t_E \, \lambda, \tag{4.14}$$

$$p(I \to S) = \Delta t_I \, \gamma. \tag{4.15}$$

As in the SIS model, we consider the influence, local or global, of collective behavior on the disease, making the strength of contagion $\hat{\beta}$ dependent on the fraction of cooperators, in an exponential function:

$$\hat{\beta}(\mathsf{c}) = \beta \exp(-\mathsf{c}) \tag{4.16}$$

As this SEIS model has its own natural inhibitor, we must adjust the reward matrix of the evolutionary game, so that the difference between the reward for cooperating and that for defect is smaller and the threshold that allows the appearance of cooperation in the network is reached. This is achieved by changing the defector reward in the presence of a cooperator from 1.5 to, say, 1.1, equally affected by the awareness function.

$$B = \begin{matrix} & \begin{matrix} C & \quad D \end{matrix} \\ \begin{matrix} C \\ D \end{matrix} & \begin{bmatrix} 1.0 & -0.5 \\ 1.1 - g(\sigma, I) & -g(\sigma, I) \end{bmatrix} \end{matrix}. \tag{4.17}$$

we use the awareness function $g(\sigma, I)$ of (3.10) and (3.11), and therefore, the information about the infected are kept modifying the evolutionary game, both global and local. The change in strategy follows the Fermi function of (3.13).

4.2.1 Algorithms

The iterative replicator is almost the same for the evolutionary game except because the reward for the defector versus a cooperator has changed, that is, we introduce (4.17) replacing (3.9). However, there is a big change in the disease propagation dynamics, as the susceptible nodes fall into the E state after contagion. Also, after latent period, nodes in E state begin infection. These changes are shown in flowcharts of Figure 4.5 and 4.6. We run this model one hundred times for each pair of parameters σ and $\mathcal{R}_0$. The code used is in Appendix A.
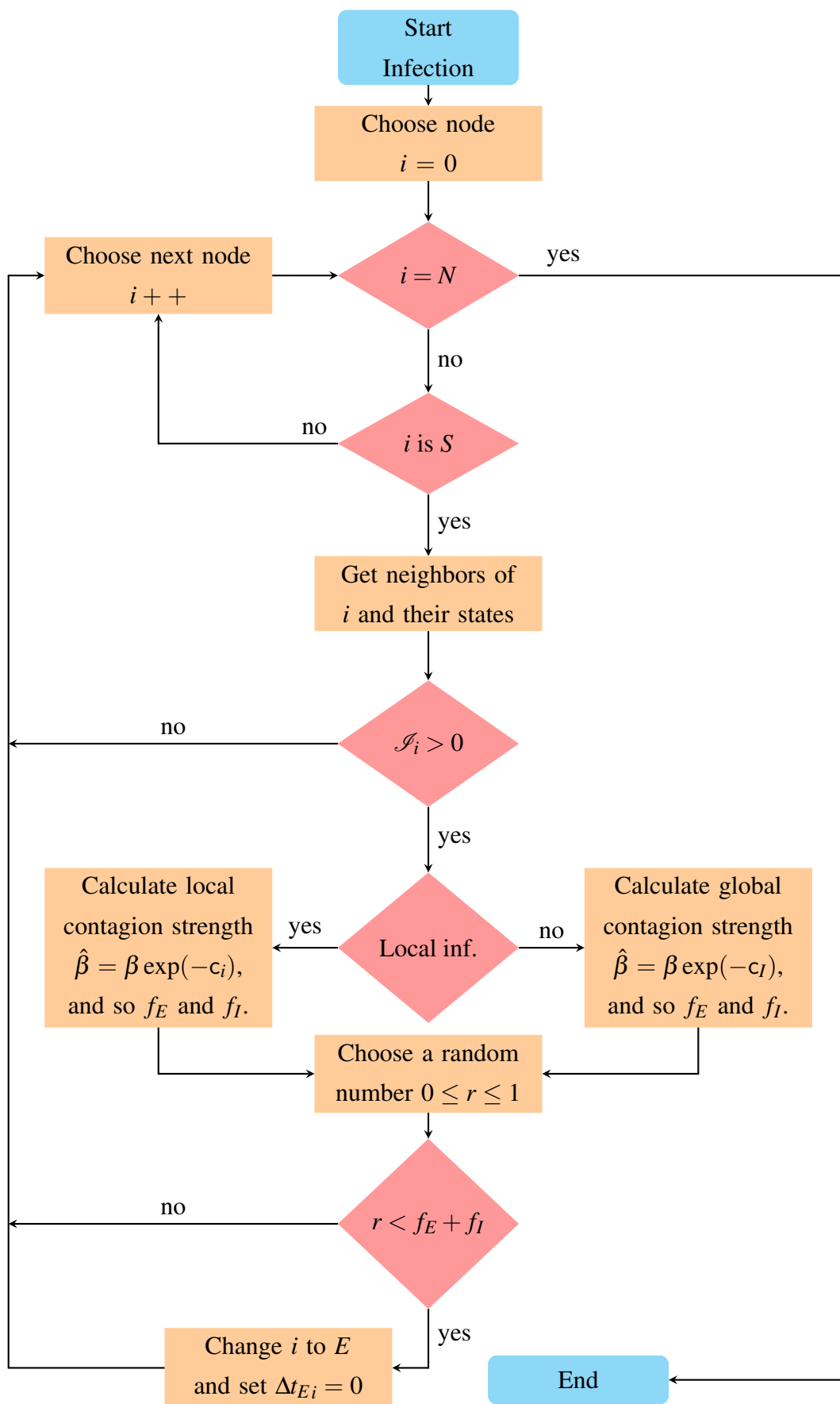
Figure 4.5: Flowchart of the contagion process in SEIS model.
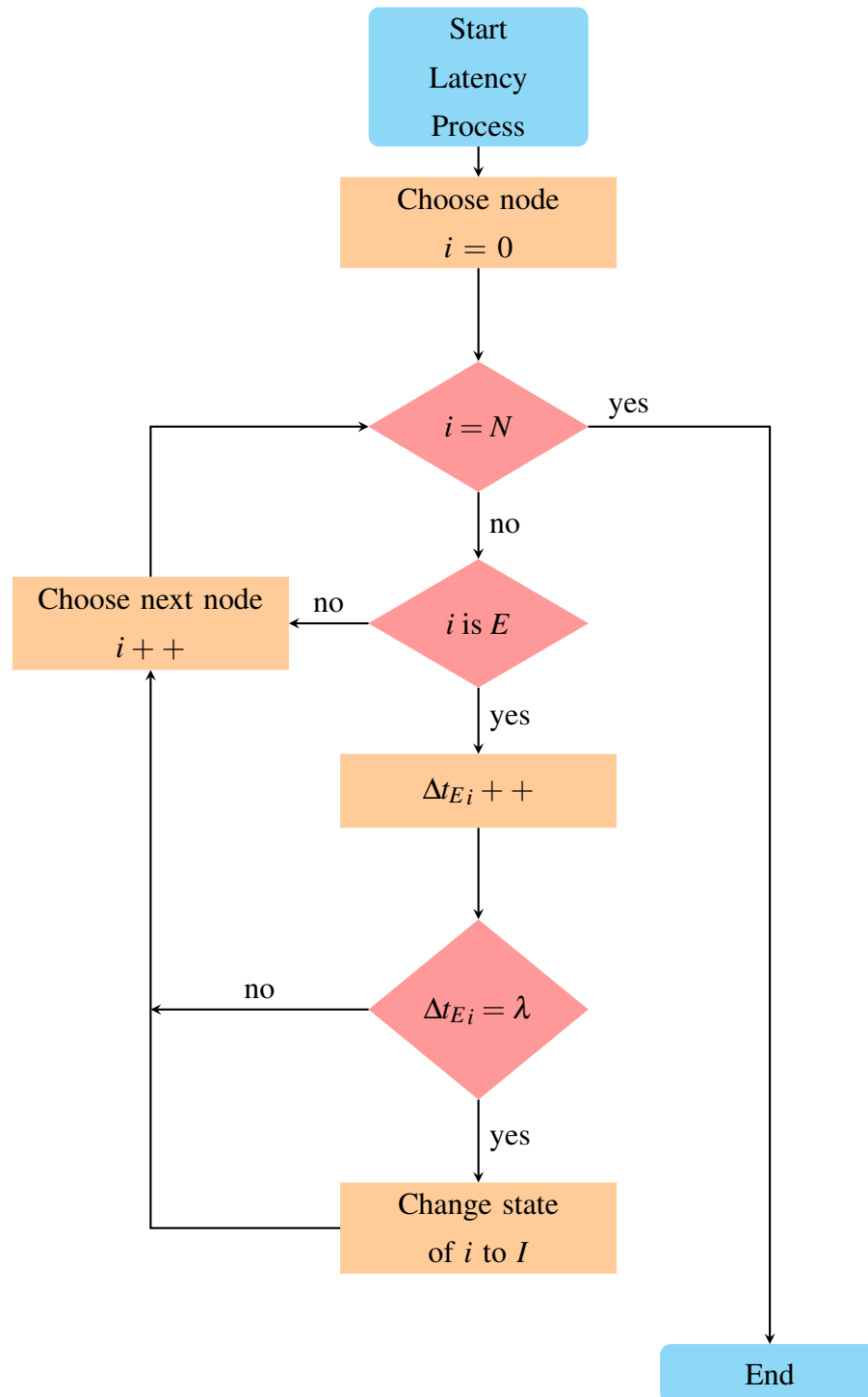
Figure 4.6: Flowchart of the latency process in SEIS model.

4.2.2   Infected and cooperators fractions with local and global information

As initial conditions of the network, both for the state of the disease and the strategies of the game, we establish that 50% of the population is cooperator. These initial conditions do not represent any distortion of the model, because the non-cooperative strategy is dominant. On the other hand,

thirty three recently infected individual are included, which represents around 3% of the population. Everyone else is susceptible, so there are not exposed individuals at that moment. After the examination period, T = 150 days, the disease is close to reaching a stable state, and we can count the number of infected and cooperators out of the total population as seen in Figure 4.7. Since β runs from 0.1 to 1.0, the range of the basic reproduction number is $\mathscr{R}_0 \in [1.55, 15.5]$, as calculated using (4.12). Cooperation emerges for $\sigma > 0.7$ and for $\mathscr{R}_0 > 3.1$. It is important to highlight that for $\mathscr{R}_0 > 12.4$, the fraction of cooperation begins to decrease for awareness values where, with lower $\mathscr{R}_0$, cooperation was obtained. We can conclude that the SEIS model used allows cooperation to be achieved for reproduction numbers similar to those allowed by the previous SIS model and SIR model. On this occasion, the cooperation achieved is sufficient to influence the development of the disease, causing the fraction of infected to be lower than if this strategy were not predominant.
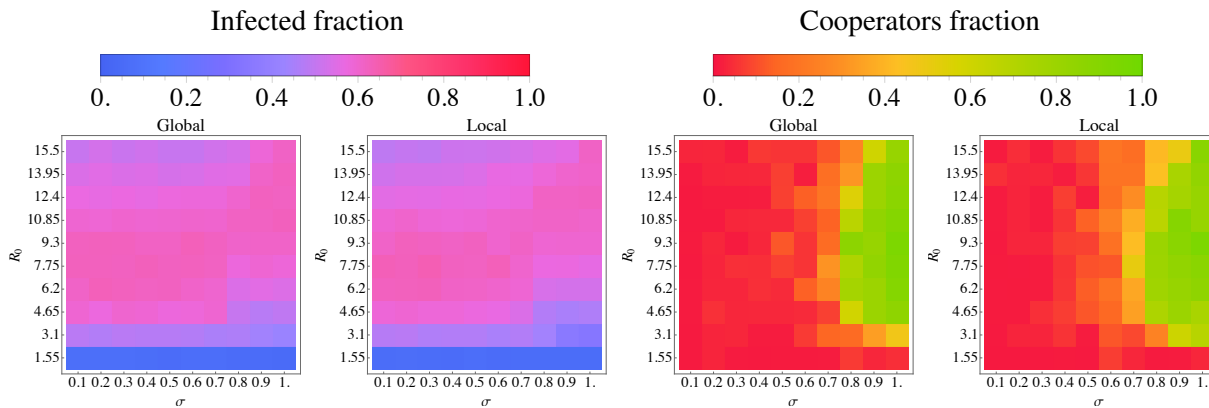
Figure 4.7: State of infected fraction, and cooperators fraction in population of scale-free network, by the basic reproduction number $\mathscr{R}_0$ (*y*-axis), and awareness σ (*x*-axis), for global (left) and local (right) information and influence.

4.2.3   Disease and behavioral dynamics by time units

The fraction of cooperation drops to approximately 0.1 once the disease begins. When the disease reaches its peak (quite attenuated compared to the SIS model, at the same time less oscillating), the cooperation fraction increases rapidly in the network, reaching values greater than 0.6 for awareness σ or basic reproduction number $\mathscr{R}_0 > 3.1$. These results are summarized in Figure 4.8 for global and local information about the disease and influence over the evolutionary game.

(a) Global information
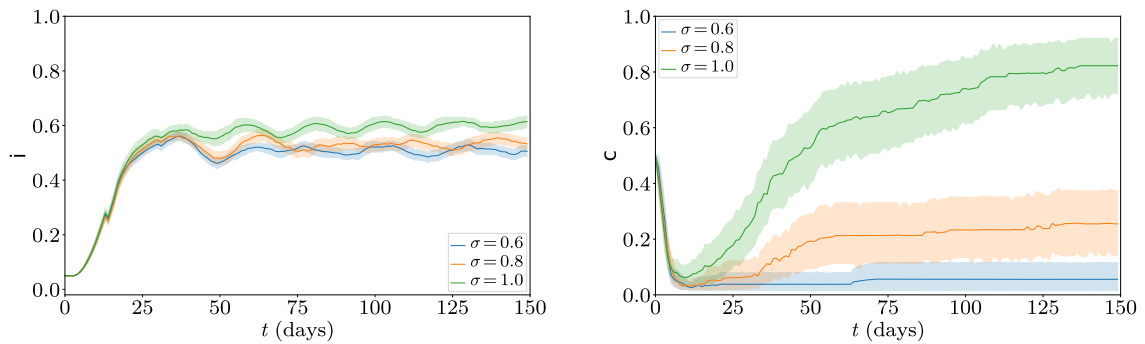
(b) Local information

Figure 4.8: Temporal evolution for different awareness $\sigma \in \{0.6, 0.8, 1.0\}$, and a basic reproduction number $\mathscr{R}_0 = 12.4$. The lines represents the mean of 100 simulations with 95% confidence interval. Left column shows the evolution of the infected fraction, while right corresponds to cooperators fraction. The interactions between nodes respond to the global (a) or local (b) influence and information of cooperation and disease respectively.

4.3 Cooperation emergence with a 3-strategy evolutionary game model

Following the generalization of the prisoner's dilemma in a **voluntary public goods game**, in this case, individuals have the possibility of choosing between three different strategies: to cooperate

(*C*), defect (*D*) or isolate (*L*), with the rules of the reward matrix:

$$
G = \begin{array}{c} \\ C \\ D \\ L \end{array}
\begin{array}{c} C \\ \left[\begin{array}{ccc} 1 & -0.5 & 0.5 \\ 1.5 - g(\sigma, I) & -g(\sigma, I) & -0.5 - g(\sigma, I) \\ 0 & -g(\sigma, I) & 0.75 - g(\sigma, I) \end{array} \right]. \end{array} \tag{4.18}
$$

In this model, we consider that information about the disease affects the defectors (*D*) in any circumstance. Meanwhile, loners (*L*) are only affected by the awareness function $g(\sigma, I)$, when the other individual is a defector (*D*) or is also a loner.

The epidemic will be a SIS model, equivalent to the system of (3.1) and (3.2). Therefore, the stochastic propagation model will be that of (3.5) and (3.6). It follows then that $\mathscr{R}_0 = \beta / \gamma$.

The probability of changing strategy is defined by the Fermi function of (3.13). Each node in the network is taken, and their respective payoffs, π_i are calculated, according to the (3.12).

4.3.1 Algorithms

The iterative replicator is almost the same for the evolutionary game except because the reward for the defector versus a cooperator has changed and now we use the 3×3-matrix *G* of (4.18) and nodes can take one of three states *C*, *D* and *L*. We also run this model one hundred times for each pair of parameters σ and $\mathscr{R}_0$. The code used is in Appendix A.

4.3.2 Infected and cooperators fractions with local and global information

As initial conditions, we now have a third of cooperators, another third of defectors and the remaining third as loners. The examination time is 150 days, when the disease is expected to be around steady state. The results are compiled in Figure 4.9. Cooperation emerges in the high values of awareness and reproduction number. However, for these parameters, the region is much broader. In the rest of the parameter space, it is the defector's strategy that predominates. However, for very low values of both awareness and strength of the disease, the strategy of isolating in part disputes the predominance of that of not cooperating.

Finally, the fraction of infected is reduced for the set of awareness and $\mathscr{R}_0$ values for which cooperation emerges.
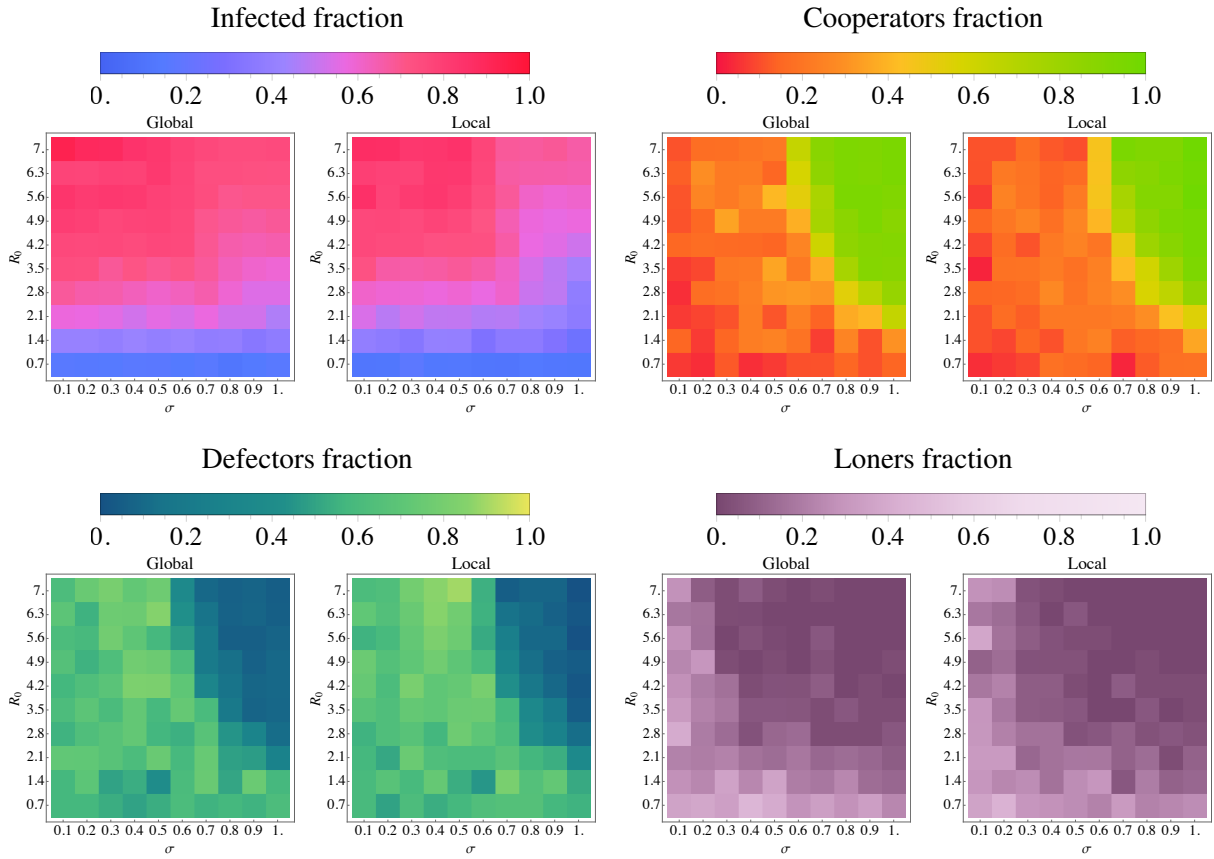
Figure 4.9: State of infected (upper left), cooperators (upper right), defectors (lower left) and loners (lower right) fraction in population of scale-free network, by the basic reproduction number $\mathscr{R}_0$ (*y*-axis), and awareness σ (*x*-axis), for global (left) and local (right) information and influence.

4.3.3 Disease and behavioral dynamics by time units

The behavior of the evolutionary game varies with respect to previous models, as shown in Figure 4.10, mainly due to the change in the reward matrix. There is no longer an initial minimum point for cooperation, but depending on the level of awareness, the fraction of cooperators increases or decreases progressively. Defectors, likewise, rapidly increase in number when the level of awareness is low. But when σ is very low, they stagnate, and instead, the number of loners increases until it reaches its ceiling.

On the other hand, the fraction of infected actually has a notable reduction when high cooperation is manifested. Also, the steady state of infection is reached quickly when cooperators are the majority, and the quantity of infected nodes does not oscillate much around this point.
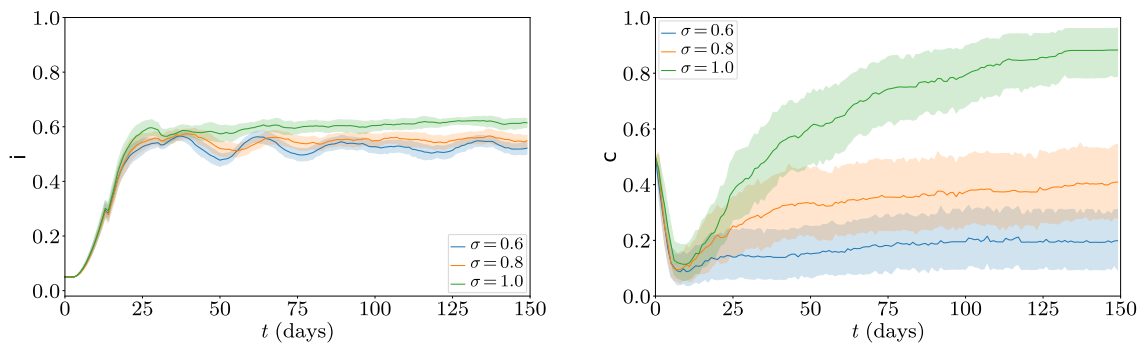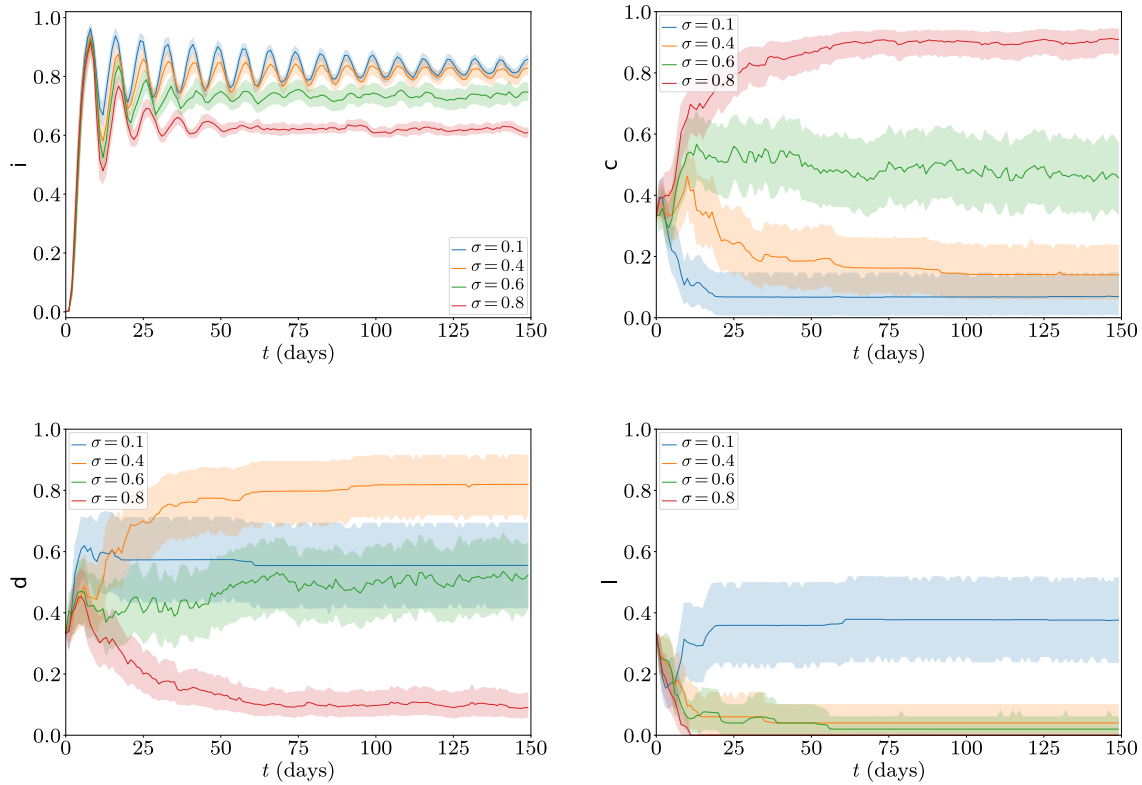
Figure 4.10: Temporal evolution for different awareness $\sigma \in \{0.1, 0.4, 0.6, 0.8\}$, and a basic reproduction number $\mathscr{R}_0 = 5.6$. The lines represents the mean of 100 simulations with 95% confidence interval. Upper left plot shows the evolution of the infected fraction, while upper right corresponds to cooperators fraction. In addition, lower left is the defectors fraction and lower right is the loners fraction on time. The interactions between nodes respond to the local influence and information of cooperation and disease respectively.

Chapter 5

Conclusions

Cooperation as an emergent phenomenon can result from more complex dynamics than copying the most profitable strategy against individuals in contact, when it is the globally beneficial strategy. It can also be obtained when the network or community in question faces a survival or well-being problem, such as an illness. Then, the parameters of the disease, such as the strength of contagion or the recovery rate, affect the strategy that individuals in a social network take.

We have found that cooperation appears for high levels of awareness ($\sigma > 0.6$), in addition to a sufficient strength of the disease ($\beta > 0.2$) or in the case of the SIS model studied, a value $\mathscr{R}_0 > 2.1$; and only after the first peak of the disease has been reached, this is when the maximum number of infected individuals in the network is reached.

Our results show that the fraction of cooperation within the communities of a scale-free network is similar in most cases, with some exceptions due to the network structure. The emergence of cooperation at different scales means that this collective phenomenon is practically independent from the evolutionary game influenced by local or global knowledge about the number of infected, represented in the awareness function g. In addition, the spread of the disease is not conditioned differently by a global influence or by a local influence of cooperation in the network.

We have uncovered that topology is relevant in the dynamics of evolutionary games. Strong cooperation emerge in scale-free networks as well as in globally coupled networks, while it is null in grid networks or small-world random networks. This occurs due to the dependence of the cooperation strategy on a high number of participants in the total sum of the payoff of the focal nodes that are compared for the change of strategy according to the Fermi model. Therefore, cooperation only appears after exceeding a threshold, from which the cooperator strategy is superior to that of being a defector in networks that are highly connected, as in a globally coupled network, or have highly connected nodes, as in a scale-free network. Furthermore, cooperation is strengthened in order of importance when: (i) the number of edges between nodes increases, (ii) the average degree of the network increases, and (iii) the clustering coefficient grows.

We showed that the appearance of cooperation in globally coupled networks allows to infer the behavior of an equivalent system of differential equations for the two coupled dynamics, where cooperation would also be an emergent phenomenon for high levels of awareness.

The SIR model revealed that the emergence cooperation depends on the estimated recovery time after infection. Because the cooperation strategy is adopted node by node, it is necessary that the infection status is maintained on a large portion of the population for at least a certain time. Otherwise, cooperation will not be able to be dominant or exceed the fraction threshold from which this strategy is maintained over time.

We found that the strategy of cooperating is also dominant in systems where the game dynamics is coupled to epidemic propagation models that are more complex than the bilinear relationship, or that are more realistic. This result suggests a possible application of these models to generate collective cooperation strategies in order to mitigate the consequences of infectious diseases. Cooperation also emerges for certain parameter values, where individuals have more strategies than whether to cooperate or not.

A future step of research points to the study of the emergence of collective cooperation behavior in the face of natural or social problems other than the spread of a disease. Also of interest is the investigation of cooperation beyond evolutionary games that are guided by the principle of natural selfishness. These generalizations may contribute to understand, through the Physics of Complex Systems and without anthropological or even biological explanations, cooperation as an emergent phenomenon of humanity under adverse circumstances.

Bibliography

[1] Wikimedia Commons, WorldWideWeb Around Wikipedia – Wikipedia as part of the World Wide Web. 2004; `https://commons.wikimedia.org/wiki/File:WorldWideWebAroundWikipedia.png`, File: `WorldWideWebAroundWikipedia.png`.

[2] Hethcote, H. W. The Mathematics of Infectious Diseases. *SIAM Review* **2000**, *42*, 599–653.

[3] Engels, F. *Dialectics of Nature*; Well Red Publications, 2012; written between 1873 and 1886.

[4] Brand, M. Dialectics, Complexity and the Crisis. N.D.; unpublished.

[5] Jensen, H. J. *Complexity science: the study of emergence*; Cambridge University Press, 2022.

[6] Sayama, H. *Introduction to the modeling and analysis of complex systems*; Open SUNY Textbooks, 2015.

[7] Sen, P.; Chakrabarti, B. K. *Sociophysics: an introduction*; OUP Oxford, 2014.

[8] Gross, T.; Blasius, B. Adaptive coevolutionary networks: a review. *Journal of The Royal Society Interface* **2008**, *5*, 259–271.

[9] West, R.; S., M.; Rubin, G.; Amlot, R. Applying principles of behaviour change to reduce SARS-CoV-2 transmission. *Nature Human Behaviour* **2020**, *4*, 451–459.

[10] N., F. Capturing human behaviour. *Nature* **2007**, *446*, 733–733.

[11] Funk, S.; Salathé, M.; Jansen, V. Modelling the influence of human behaviour on the spread of infectious diseases: a review. *Journal of The Royal Society Interface* **2010**, *7*, 1247–1256.

[12] Gross, J. L.; Yellen, J.; Anderson, M. *Graph Theory and Its Applications*, 3rd ed.; Chapman and Hall/CRC, 2018.

[13] Estrada, E. *The Structure of Complex Networks*; Oxford University Press, 2011.

[14] Freeman, L. *The Development of Social Network Analysis: A Study in the Sociology of Science*; Empirical Press, 2004.

[15] Quesnay, F. Tableau économique. 1758-1759.

[16] Covington, P.; Adams, J.; Sargin, E. Deep Neural Networks for YouTube Recommendations. *Proceedings of the 10th ACM Conference on Recommender Systems* **2016**, 191–198.

[17] Harary, F. *Graph Theory*; CRC Press, 2018.

[18] Milgram, S. The small world problem. *Psychology today* **1967**, *2*, 60–67.

[19] Barabási, A.-L.; Bonabeau, E. Scale-Free Networks. *Scientific American* **2003**, *288*, 60–69.

[20] Watts, D. J.; Strogatz, S. H. Collective dynamics of 'small-world' networks. *Nature* **1998**, *393*, 440–442.

[21] Goh, K.-I.; Oh, E.; Jeong, H.; Kahng, B.; Kim, D. Classification of scale-free networks. *Proceedings of the National Academy of Sciences* **2002**, *99*, 12583–12588.

[22] van den Driessche, P.; Watmough, J. Reproduction numbers and sub-threshold endemic equilibria for compartmental models of disease transmission. *Mathematical Biosciences* **2002**, *180*, 29–48.

[23] Kerr, B.; Riley, M. A.; Feldman, M. W.; Bohannan, B. J. M. Local dispersal promotes biodiversity in a real-life game of rock–paper–scissors. *Nature* **2002**, *418*, 171–174.

[24] Turner, P. E.; Chao, L. Prisoner's dilemma in an RNA virus. *Nature* **1999**, *398*, 441–443.

[25] Schuster, H. G., Ed. *Reviews of Nonlinear Dynamics and Complexity*; Wiley, 2008; Vol. 2.

[26] Axelrod, R.; Hamilton, W. D. The evolution of cooperation. *science* **1981**, *211*, 1390–1396.

[27] Kagel, J. H.; Roth, A. E. *The handbook of experimental economics, volume 2*; Princeton university press, 2020.

[28] Hauert, C.; Traulsen, A.; Brandt, H.; Nowak, M. A.; Sigmund, K. Via freedom to coercion: the emergence of costly punishment. *science* **2007**, *316*, 1905–1907.

[29] Diekmann, A.; Lindenberg, S. *Cooperation: Sociological Aspects*; Elsevier, 2001; pp 2751–2756.

[30] Kermack, W. O.; McKendrick, A. G. A contribution to the mathematical theory of epidemics. *Proceedings of the Royal Society of London. Series A, Containing Papers of a Mathematical and Physical Character* **1927**, *115*, 700–721.

[31] Cascante-Vega, J.; Torres-Florez, S.; Cordovez, J.; Santos-Vega, M. How disease risk aware-ness modulates transmission: coupling infectious disease models with behavioural dynamics. *Royal Society Open Science* **2022**, *9*.

[32] Blondel, V. D.; Guillaume, J.-L.; Lambiotte, R.; Lefebvre, E. Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment* **2008**, *2008*, P10008.

[33] Sayama, H. *Introduction to the modeling and analysis of complex systems*; Open SUNY Textbooks, 2015.

[34] Latora, V.; Marchiori, M. Efficient Behavior of Small-World Networks. *Physical Review Letters* **2001**, *87*, 198701.

[35] Saramäki, J.; Kivelä, M.; Onnela, J.-P.; Kaski, K.; Kertész, J. Generalizations of the clustering coefficient to weighted complex networks. *Physical Review E* **2007**, *75*, 027105.

[36] Newman, M. E. J. Mixing patterns in networks. *Physical Review E* **2003**, *67*.

[37] World Health Organization, Consensus document on the epidemiology of severe acute respiratory syndrome (SARS). 2003.

[38] Liu, Y.; Rocklöv, J. The effective reproductive number of the Omicron variant of SARS-CoV-2 is several times relative to Delta. *Journal of Travel Medicine* **2022**, *29*, taac037.

[39] Gani, R.; Leach, S. Transmission potential of smallpox in contemporary populations. *Nature* **2001**, *414*, 748–751.

[40] Australian Government Department of Health, Mumps (Mumps virus) Laboratory case definition. 2015.

[41] Guerra, F. M.; Bolotin, S.; Lim, G.; Heffernan, J.; Deeks, S. L.; Li, Y.; Crowcroft, N. S. The basic reproduction number (R0) of measles: a systematic review. *The Lancet Infectious Diseases* **2017**, *17*, e420–e428.

[42] Capasso, V.; Serio, G. A generalization of the Kermack-McKendrick deterministic epidemic model. *Mathematical Biosciences* **1978**, *42*, 43–61.

[43] Naim, M.; Lahmidi, F.; Namir, A. Threshold Dynamics of an SEIS Epidemic Model with Nonlinear Incidence Rates. *Differential Equations and Dynamical Systems* **2021**,

Appendix A

Computer Code

We develop the code for the simulations in Python 3.9 with the libraries to date. The iterative processes of the coupled dynamics for contagion, recovery, and the evolutionary game of Chapter 3 and 4 are the functions presented below:

```python
import numpy as np
import pandas as pd
import networkx as nx
import random

### PAYOFF FUNCTION
'''
    Input:
    A - Payoff matrix .
    nn - Nearest neighboor obtained from adjacency matrix.
    node_i - Node to calculate payoff.
    current_state - Current node states.
    Output:
    u - node payoff (float)
'''
def payoff(A, nn, node_i, current_state):

    u = 0;
    for j, j_nn in enumerate(nn):
        u += A[int(actual_state[node_i]), int(actual_state[j_nn])]
    u /= NN.size
    return u

### FERMI DISTRIBUTION FUNCTION
'''
    Input:
    u_i - Payoff of focal node.
```

```
28      u_j - Payoff of model node.
29      Output:
30      Pij - Probability of change i's strategy to j's one.
31  '''
32  def Pij(u_i, u_j):
33      k = 0.5       # Irrationality of players
34      Pij = 1/(1 + np.exp(-(u_j - u_i)/k))
35      return Pij
36
37
38  ### SIS-PD REPLICATOR FUNCTION
39  '''
40      Input:
41      graph - Networkx graph.
42      max_time -  period of examination
43      beta - Contagion strenght.
44      sigma - Awareness level.
45      gamma - Recovery factor
46      local - Boolean for using global (False) or local (True) information.
47  '''
48  def sis_prisoner(graph, max_time, beta, sigma, gamma=1/7, local=True):
49
50      N = len(graph.nodes())
51      G = graph
52
53      id_node = {node:idx for idx,node in enumerate(G.nodes())}
54
55      T = max_time
56      maxRecTime = int(1/gamma)                    # Recovery time
57
58      # Initial strategies
59
60      epid_states = np.zeros([N, T])          # 1: infected, 0: susceptible
61      game_states = np.zeros([N, T])          # 1: defectors, 0: coperator
62
63      recTime = np.zeros(N,)                        # Count of recovery time
        of each node.
64      payoffs = np.zeros([N, T])           # Payoff's of each iteration.
65
66      ## Initial infected nodes
67
68      if initInfected is None:     # If initial conditions are given
69          initInfected = random.sample(range(int(N)), 1)
70          epid_states[initInfected,0] = 1
```

```
71
72      else:
73          epid_states[initInfected,0] = 1
74
75      # Start recovery of infected nodes
76      recTime[initInfected] = maxRecTime
77
78
79      ## Initial defectors nodes
80      if initDefectors is None:    # If initial conditions are given
81          initDefectors = random.sample(range(int(N)), int(N/2))
82          game_states[initDefectors,0] = 1
83      else:
84          game_states[initDefectors,0] = 1
85
86
87      ## Simulation
88      for t in range(1, T):
89
90          curInf_nodes = np.nonzero(epid_states[:,t-1])[0]    # Get current
    infected nodes
91          epid_states[curInf_nodes, t] = 1                             # Update
    infected nodes
92
93          if local is False:
94              FracInf = np.sum(epid_states[:,t])/N          # Get fraction of
     infected nodes
95              g_func = sigma*FracInf                    # Calculate g global
    awareness function
96
97          recTime = np.maximum(recTime-1,0)                    # Update
    infection state depending on recovery time
98          epid_states[ recTime>0, t] = 1
99          epid_states[ recTime==0, t] = 0
100
101
102         ## Run Evolutionary Game
103         for idx, node_i in enumerate(G.nodes()):
104             node_i_nn_ = [neig for neig in G.neighbors(node_i)]          #
    Get neighbors
105             node_i_nn  = np.array([id_node[nn] for nn in node_i_nn_ ])
106
107             if local is True:
```

65

```
108              FracInf = np.sum(epid_states[node_i_nn, t-1])/node_i_nn.
    size          # Calculate fraction of infected neighbors
109              g_func = sigma*FracInf
        # Calculate g local awareness function
110
111          A = np.array([[1,-0.5], [1.5-g_func, 0-g_func]])
    # Game payoff matrix
112
113          payoffs[node_i, t-1] = payoff(A, node_i_nn, node_i,
    game_states[:,t-1])  # Calculate payoff
114
115      for idx, node_i in enumerate(G.nodes()):
116          node_i_nn_ = [neig for neig in G.neighbors(node_i)]        #
    get neighbors
117          node_i_nn  = np.array([id_node[nn] for nn in node_i_nn_ ])
118
119          modelnode = random.randint(1, node_i_nn.size)   # Choose a
    model node.
120          P_change = Pij(payoffs[node_i,t-1], payoffs[modelnode,t-1])
    # Calculate probability Pij with Fermi function.
121
122          th = 0.5              # Threshold to change.
123          game_states[node_i, t] = (P_change>th)*game_states[modelnode,t
    -1]+(P_change<=th)*game_states[node_i,t-1]
124
125
126      ## Run SIS Infection
127      for idx, node_i in enumerate(G.nodes()):
128          if epid_states[node_i, t-1] == 1:      # If node is already
    infected
129              continue                           # Skip
130          cur_node_nn_ = [neig for neig in G.neighbors(node_i)]
             # Get neighbors
131          cur_node_nn = np.array([id_node[nn] for nn in cur_node_nn_ ])
132          cur_node_inf_nn = np.nonzero(epid_states[cur_node_nn,t-1])[0]
             # Get infected neighbors
133          cur_game_nn = game_states[cur_node_inf_nn,t-1]
             # Get game states infected neighbors
134
135          n_infected_neighbors = np.sum(epid_states[cur_node_nn,t-1])
136          # if local is False:
137          p_inf_node_i = beta*np.exp(-np.sum(game_states[curInf_nodes,t
    ]==0)/N)        # Infection probability due to cooperation state of all
     network infected nodes
```

66

```python
138
139             if local is True:         # If only considering local neighbors
140                 if n_infected_neighbors > 0:     # If there is infection in
    my neighbors
141                     p_inf_node_i = beta*np.exp(-np.sum(cur_game_nn==0)/
    cur_game_nn.size)  # Infection probability due to cooperation state of
    infected neighbors
142                 else:
143                     p_inf_node_i = 0.00000001
144
145             if n_infected_neighbors == 0:   # If no neighbor is infected,
    infection probability ~ 0
146                 p_inf_node_i = 0.00000001
147
148             # Update new infection
149             if random.random() < p_inf_node_i:
150                 epid_states[node_i, t] = 1
151                 # Update recovery time
152                 recTime[node_i] = maxRecTime
153
154     return game_states, epid_states, payoffs
155
156
157
158 ### SIR-PD REPLICATOR FUNCTION
159 '''
160     Input:
161     graph - Networkx graph.
162     max_time -  period of examination
163     beta - Contagion strenght.
164     sigma - Awareness level.
165     gamma - Recovery factor
166     local - Boolean for using global (False) or local (True) information.
167 '''
168 def sir_prisoner(graph, max_time, beta, sigma, gamma=1/14, local=True):
169
170     N = len(graph.nodes())
171     G = graph
172
173     id_node = {node:idx for idx,node in enumerate(G.nodes())}
174
175     T = max_time
176
177     maxRecTime = int(1/gamma)                 # Recovery time
```

```python
178
179      # Initial strategies
180
181      epid_states = np.zeros([N, T])              # 1: infected, 0: susceptible
182      game_states = np.zeros([N, T])              # 1: defectors, 0: coperator
183
184      recTime = np.zeros(N,)                           # Count of recovery time
    of each node.
185      payoffs = np.zeros([N, T])              # Payoff's of each iteration.
186
187      ## Initial infected nodes
188
189      if initInfected is None:     # If initial conditions are given
190          initInfected = random.sample(range(int(N)), 1)
191          epid_states[initInfected,0] = 1
192
193      else:
194          epid_states[initInfected,0] = 1
195
196      # Start recovery of infected nodes
197      recTime[initInfected] = maxRecTime
198
199
200      ## Initial defectors nodes
201      if initDefectors is None:     # If initial conditions are given
202          initDefectors = random.sample(range(int(N)), int(N/2))
203          game_states[initDefectors,0] = 1
204      else:
205          game_states[initDefectors,0] = 1
206
207
208      ## Simulation
209      for t in range(1, T):
210
211          curInf_nodes = np.nonzero(epid_states[:,t-1]==1)[0]      # Get
    current infected nodes
212          curRec_nodes = np.nonzero(epid_states[:,t-1]==2)[0]      # Get
    current recovered nodes
213          epid_states[curInf_nodes, t] = 1                            # Update
    infected nodes
214          epid_states[curRec_nodes, t] = 2                            # Update
    recovered nodes
215
216          if local is False:
```

```python
        FracInf = np.count_nonzero(epid_states[:,t]==1)/N          #
Get fraction of infected nodes
        g_func = sigma*FracInf                # Calculate sigma
infection

    recTime = recTime -1                                # Update
infection state depending on recovery time

    epid_states[ recTime>0, t] = 1                # Keep the
infected state if recover time is not reached
    epid_states[ recTime==-1, t] = 0              # Keep the
suceptible state if not been infected yet
    epid_states[ recTime==0, t] = 2               # Go to
recovereds state if recovery time is reached
    epid_states[ recTime==-2, t] = 2              # Keep the
recovered stated if recovery time was reached

    recTime[epid_states[:,t]==0] = 0
    recTime[epid_states[:,t]==2] = -1


    ## Run Evolutionary Game
    for idx, node_i in enumerate(G.nodes()):
        node_i_nn_ = [neig for neig in G.neighbors(node_i)]          #
 Get neighbors
        node_i_nn  = np.array([id_node[nn] for nn in node_i_nn_ ])

        if local is True:
            FracInf = np.count_nonzero(epid_states[node_i_nn, t-1]==1)
/node_i_nn.size          # Calculate fraction of infected neighbors
            g_func = sigma*FracInf
    # Calculate g local awareness function

        A = np.array([[1,-0.5], [1.5-g_func, 0-g_func]])
    # Game payoff matrix

        payoffs[node_i, t-1] = payoff(A, node_i_nn, node_i,
game_states[:,t-1])  # Calculate payoff

    for idx, node_i in enumerate(G.nodes()):
        node_i_nn_ = [neig for neig in G.neighbors(node_i)]         #
Get neighbors
        node_i_nn  = np.array([id_node[nn] for nn in node_i_nn_])
```

```
248            modelnode = random.randint(1, node_i_nn.size)   # Choose a
      model node.
249            P_change = Pij(payoffs[node_i,t-1], payoffs[modelnode,t-1])
      # Calculate probability Pij with Fermi function.
250
251            th = 0.5              # Threshold to change.
252            game_states[node_i, t] = (P_change>th)*game_states[modelnode,t
      -1]+(P_change<=th)*game_states[node_i,t-1]
253
254
255        ## Run SIR Infection
256        for idx, node_i in enumerate(G.nodes()):
257            if epid_states[node_i, t-1] == 1 or epid_states[node_i, t-1]
      == 2:      # If node is already infected
258                continue                               # Skip
259            cur_node_nn_ = [neig for neig in G.neighbors(node_i)]
                 # Get neighbors
260            cur_node_nn = np.array([id_node[nn] for nn in cur_node_nn_ ])
261            cur_node_inf_nn = np.nonzero(epid_states[cur_node_nn,t-1])[0]
                 # Get infected neighbors
262            cur_game_nn = game_states[cur_node_inf_nn,t-1]
                 # Get game states infected neighbors
263
264            n_infected_neighbors = np.sum(epid_states[cur_node_nn,t-1])
265            # if local is False:
266            p_inf_node_i = beta*np.exp(-np.sum(game_states[curInf_nodes,t
      ]==0)/N)       # Infection probability due to cooperation state of all
       network infected nodes
267
268            if local is True:        # If only considering local neighbors
269                if n_infected_neighbors > 0:     # If there is infection in
       my neighbors
270                    p_inf_node_i = beta*np.exp(-np.sum(cur_game_nn==0)/
      cur_game_nn.size)   # Infection probability due to cooperation state of
      infected neighbors
271                else:
272                    p_inf_node_i = 0.00000001
273
274            if n_infected_neighbors == 0:    # If no neighbor is infected,
      infection probability ~ 0
275                p_inf_node_i = 0.00000001
276
277            # Update new infection
278            if random.random() < p_inf_node_i:
```

```python
279                     epid_states[node_i, t] = 1
280                     # Update recovery time
281                     recTime[node_i] = maxRecTime
282
283         return game_states, epid_states, payoffs
284
285
286
287 ### SEIS-PD REPLICATOR FUNCTION
288 '''
289     Input:
290     graph - Networkx graph.
291     T -  period of examination
292     beta - Contagion strenght.
293     sigma - Awareness level.
294     gamma - Recovery factor
295     local - Boolean for using global (False) or local (True) information.
296 '''
297 def seis_prisoner(graph, max_time, beta, sigma, gamma=1/14, lambda = 1/3,
    local=True):
298
299     N = len( graph.nodes() )
300     G = graph
301
302     id_node = {node:idx for idx,node in enumerate(G.nodes())}   # In case
    nodes are not ints
303
304     T = max_time
305
306     maxRecTime = int(1/gamma)
307     maxExpTime = int(1/lambda)
308
309     # Initial strategies
310
311     epid_states = np.zeros([N, max_time])    # 1: exposed, 0: susceptible
    , 2: infected
312     game_states = np.zeros([N, max_time])     # 1: defectors, 0: coperator
313
314     recTime = np.zeros(N,)
315     expTime = np.zeros(N,)
316     payoffs = np.zeros([N, T])         # Payoff's of each iteration
317
318     # Initial infected and exposed nodes
319     if initInfected is None:     # If initial conditions are given
```

71

```
320        initInfected = random.sample(range(int(N)), int(N/30))
321        epid_states[initInfected,0] = 2
322
323    else:
324        epid_states[initInfected,0] = 2
325
326    # Start recovery of infected nodes
327    recTime[initInfected] = maxRecTime
328
329    # Initial defectors nodes
330    if initDefectors is None:   # If initial conditions are given
331        initDefectors = random.sample(range(int(N)), int(N/30))
332        game_states[initDefectors,0] = 1
333    else:
334        game_states[initDefectors,0] = 1
335
336    ## Simulation
337    for t in range(1, T):
338
339        curExp_nodes = np.nonzero(epid_states[:,t-1]==1)[0]      # Get
    current exposed nodes
340        curInf_nodes = np.nonzero(epid_states[:,t-1]==2)[0]      # Get
    current infected nodes
341
342        epid_states[curExp_nodes, t] = 1                         # Update
    exposed nodes
343        epid_states[curInf_nodes, t] = 2                         # Update
    infected nodes
344
345        if local is False:
346            FracInf = np.count_nonzero(epid_states[:,t]==2)/N        #
    Get fraction of infected nodes
347            #FracExp = np.count_nonzero(epid_states[:,t]==1)/N        #
    Get fraction of exposed nodes
348            g_func = sigma*FracInf          # Calculate g global
    awareness function
349
350        expTime = expTime - 1
351        recTime = recTime - 1                                    # Update
    infection state depending on recovery time
352
353        epid_states[ expTime==-1, t] = 0
354        epid_states[ recTime==-1, t] = 0
355
```

```
356         epid_states[ expTime >0, t] = 1                       # Keep the
     exposed state if exposition time is not reached
357         epid_states[ expTime ==0, t] = 2                      # Go to infected
     state if exposition time is reached
358         #epid_states[ expTime ==-2, t] = 0                    # Keep the
     recovered stated if recovery time was reached
359
360         epid_states[ recTime >0, t] = 2                       # Keep the
     infected state if recover time is not reached
361         epid_states[ recTime ==0, t] = 0                      # Go to
     susceptible state if recovery time is reached
362         #epid_states[ recTime ==-2, t] = 0                    # Keep the
     susceptible state if recovery time was reached
363
364         recTime[expTime ==0] = maxRecTime                     # Give the
     recovery time to new infected nodes
365         recTime[epid_states[:,t]==0] = 0                      # Return the
     counter to 0
366         expTime[epid_states[:,t]==0] = 0                      # Return the
     counter to 0
367
368
369         ## Run Evolutionary Game
370         for idx, node_i in enumerate(G.nodes()):
371             node_i_nn_ = [neig for neig in G.neighbors(node_i)]           #
      get neighbors
372             node_i_nn  = np.array([id_node[nn] for nn in node_i_nn_ ])
373
374             if local is True:
375                 FracInf = np.count_nonzero(epid_states[node_i_nn, t-1]==2)
     /node_i_nn.size            # Calculate fraction of infected neighbors
376                 g_func = sigma*FracInf
     # Calculate g local awareness function
377
378             A = np.array([[1,-0.5], [1.1-g_func, 0-g_func]])
     # Game payoff matrix
379
380             payoffs[node_i, t-1] = payoff(A, node_i_nn, node_i,
     game_states[:,t-1])   # Calculate payoff
381
382         for idx, node_i in enumerate(G.nodes()):
383             node_i_nn_ = [neig for neig in G.neighbors(node_i)]         #
     get neighbors
384             node_i_nn  = np.array([id_node[nn] for nn in node_i_nn_ ])
```

73

```
385
386            node2change = random.randint(1, node_i_nn.size)
387            P_change = Pij(payoffs[node_i,t-1], payoffs[node2change,t-1])
       # Calculate probability Pij
388
389            th = 0.5
390            game_states[node_i, t] = (P_change>th)*game_states[node2change
      ,t-1]+(P_change<=th)*game_states[node_i,t-1]
391

392

393       ## Run SEIS Infection
394       for idx, node_i in enumerate(G.nodes()):
395            if epid_states[node_i, t-1] == 2:      # If node is already
      infected
396                continue                                # Skip
397
398            cur_node_nn_ = [neig for neig in G.neighbors(node_i)]
                 # Get neighbors
399            cur_node_nn = np.array([id_node[nn] for nn in cur_node_nn_ ])
400            cur_node_inf_nn = np.nonzero(epid_states[cur_node_nn,t-1]==2)
      [0]            # Get infected neighbors
401            cur_game_nn = game_states[cur_node_inf_nn,t-1]
                 # Get game states infected neighbors
402
403            n_neighbors = float(len(cur_node_nn_))
404            n_infected_neighbors = np.count_nonzero(epid_states[
      cur_node_nn,t-1]==2)
405            n_exposed_neighbors = np.count_nonzero(epid_states[cur_node_nn
      ,t-1]==1)
406
407            if n_infected_neighbors == 0:
408                n_infected_neighbors = 0.00000001
409
410            if n_exposed_neighbors == 0:
411                n_exposed_neighbors = 0.00000001
412
413            infFrac = n_infected_neighbors/n_neighbors
414
415            expFrac = n_exposed_neighbors/n_neighbors
416
417            beta2 = round(expFrac/(2.0 + 0.4*expFrac),6)
418            beta1 = round(infFrac/(1.0 + 0.1*infFrac),6)
419            probfact = float(beta1 + beta2)
420            #print(probfact)
```

```python
                # if local is False:
                p_inf_node_i = beta*np.exp(-np.sum(game_states[curInf_nodes,t
    ]==0)/N)*probfact          # Infection probability due to cooperation
    state of all network infected nodes

            if local is True:        # If only considering local neighbors
                if n_infected_neighbors > 0 or n_exposed_neighbors > 0:
     # If there is infection in my neighbors
                    p_inf_node_i = beta*np.exp(-np.count_nonzero(
    cur_game_nn==0)/n_neighbors)*probfact   # Infection probability due to
    cooperation state of infected neighbors

            if n_infected_neighbors == 0 and n_exposed_neighbors == 0:   #
     If no neighbor is infected, infection probability ~ 0
                p_inf_node_i = 0.00000001

            # Update new infection/exposition
            if random.random() < p_inf_node_i:
                epid_states[node_i, t] = 1
                # Update exposed time
                expTime[node_i] = maxExpTime

    return game_states, epid_states, payoffs


### SIS-VPG REPLICATOR FUNCTION
'''
    Input:
    graph - Networkx graph.
    max_time -  period of examination
    beta - Contagion strenght.
    sigma - Awareness level.
    gamma - Recovery factor
    local - Boolean for using global (False) or local (True) information.
'''
def sis_vpg(graph, max_time, beta, sigma, gamma=1/7, local=True):

    N = len(graph.nodes())
    G = graph

    id_node = {node:idx for idx,node in enumerate(G.nodes())}

    T = max_time
    maxRecTime = int(1/gamma)                # Recovery time
```

```python
459
460     # Initial strategies
461
462     epid_states = np.zeros([N, T])              # 1: infected, 0: susceptible
463     game_states = np.zeros([N, T])              # 1: defectors, 0: coperator
464
465     recTime = np.zeros(N,)                              # Count of recovery time
    of each node.
466     payoffs = np.zeros([N, T])              # Payoff's of each iteration.
467
468     ## Initial infected nodes
469
470     if initInfected is None:     # If initial conditions are given
471         initInfected = random.sample(range(int(N)), 1))
472         epid_states[initInfected,0] = 1
473
474     else:
475         epid_states[initInfected,0] = 1
476
477     # Start recovery of infected nodes
478     recTime[initInfected] = maxRecTime
479
480
481      # Initial defectors (1) and loners (2) nodes
482     if initDefectors is None:     # If initial conditions are given
483         sarr = [i for i in range(0,N)]
484         random.shuffle(sarr)
485         initDefectors = sarr[:int(len(sarr)/3)]
486         initLoners = sarr[int(2*len(sarr)/3):]
487         game_states[initDefectors,0] = 1
488         game_states[initLoners,0] = 2
489     else:
490         game_states[initDefectors,0] = 1
491         game_states[initLoners,0] = 2
492
493
494     ## Simulation
495     for t in range(1, T):
496
497         curInf_nodes = np.nonzero(epid_states[:,t-1])[0]   # Get current
    infected nodes
498         epid_states[curInf_nodes, t] = 1                              # Update
    infected nodes
499
```

```python
500        if local is False:
501            FracInf = np.sum(epid_states[:,t])/N          # Get fraction of
     infected nodes
502            g_func = sigma*FracInf                    # Calculate g global
     awareness function
503
504        recTime = np.maximum(recTime-1,0)                      # Update
     infection state depending on recovery time
505        epid_states[ recTime>0, t] = 1
506        epid_states[ recTime==0, t] = 0
507
508
509        ## Run Evolutionary Game
510        for idx, node_i in enumerate(G.nodes()):
511            node_i_nn_ = [neig for neig in G.neighbors(node_i)]          #
     Get neighbors
512            node_i_nn  = np.array([id_node[nn] for nn in node_i_nn_ ])
513
514            if local is True:
515                FracInf = np.sum(epid_states[node_i_nn, t-1])/node_i_nn.
     size          # Calculate fraction of infected neighbors
516                g_func = sigma*FracInf
         # Calculate g local awareness function
517
518            A = np.array([[1,-0.5,0.5], [1.5-g_func, 0-g_func,-0.5-g_func
     ],[0,0-g_func,0.75-g_func]])                    # Game payoff matrix
519
520            payoffs[node_i, t-1] = payoff(A, node_i_nn, node_i,
     game_states[:,t-1])  # Calculate payoff
521
522        for idx, node_i in enumerate(G.nodes()):
523            node_i_nn_ = [neig for neig in G.neighbors(node_i)]        #
     get neighbors
524            node_i_nn  = np.array([id_node[nn] for nn in node_i_nn_ ])
525
526            modelnode = random.randint(1, node_i_nn.size)   # Choose a
     model node.
527            P_change = Pij(payoffs[node_i,t-1], payoffs[modelnode,t-1])
     # Calculate probability Pij with Fermi function.
528
529            th = 0.5              # Threshold to change.
530            game_states[node_i, t] = (P_change>th)*game_states[modelnode,t
     -1]+(P_change<=th)*game_states[node_i,t-1]
531
```

```python
        ## Run SIS Infection
        for idx, node_i in enumerate(G.nodes()):
            if epid_states[node_i, t-1] == 1:       # If node is already
    infected
                continue                                 # Skip
            cur_node_nn_ = [neig for neig in G.neighbors(node_i)]
                # Get neighbors
            cur_node_nn = np.array([id_node[nn] for nn in cur_node_nn_ ])
            cur_node_inf_nn = np.nonzero(epid_states[cur_node_nn,t-1])[0]
                # Get infected neighbors
            cur_game_nn = game_states[cur_node_inf_nn,t-1]
                # Get game states infected neighbors

            n_infected_neighbors = np.sum(epid_states[cur_node_nn,t-1])
            # if local is False:
            p_inf_node_i = beta*np.exp(-np.sum(game_states[curInf_nodes,t
    ]==0)/N)       # Infection probability due to cooperation state of all
     network infected nodes

            if local is True:       # If only considering local neighbors
                if n_infected_neighbors > 0:    # If there is infection in
     my neighbors
                    p_inf_node_i = beta*np.exp(-np.sum(cur_game_nn==0)/
    cur_game_nn.size)  # Infection probability due to cooperation state of
    infected neighbors
                else:
                    p_inf_node_i = 0.00000001

            if n_infected_neighbors == 0:   # If no neighbor is infected,
    infection probability ~ 0
                p_inf_node_i = 0.00000001

            # Update new infection
            if random.random() < p_inf_node_i:
                epid_states[node_i, t] = 1
                # Update recovery time
                recTime[node_i] = maxRecTime

    return game_states, epid_states, payoffs
```