Influencers Dynamics in Viral Marketing Networks

BEATRIZ BATISTA COIMBRA
(Licenciatura em Matemática Aplicada à Tecnologia e à Empresa)

Dissertação para obtenção do grau de mestre em Matemática Aplicada à Indústria, na Área de Especialização de Tratamento de Dados

Orientador(es):
Prof. Doutor José Leonel Rocha
Profª. Doutora Sónia Carvalho

Júri:
Presidente: Prof. Doutor Luís Silva
Vogais:
Profª. Doutora Maria Cristina Serpa
Prof. Doutor José Leonel Rocha

Março de 2024

Influencers Dynamics in Viral Marketing Networks

BEATRIZ BATISTA COIMBRA
(Licenciatura em Matemática Aplicada à Tecnologia e à Empresa)

Dissertação para obtenção do grau de mestre em Matemática Aplicada à Indústria, na Área de Especialização de Tratamento de Dados

Orientador(es):
Prof. Doutor José Leonel Rocha, CEAUL, ISEL
Profª. Doutora Sónia Carvalho, CEAUL, ISEL

Júri:
Presidente: Prof. Doutor Luís Silva, CIMA, ISEL
Vogais:
Profª. Doutora Maria Cristina Serpa, CMAFCIO, ISEL
Prof. Doutor José Leonel Rocha, CEAUL, ISEL

Março de 2024

Agradecimentos

Agradeço aos meus orientadores, os professores José Leonel Rocha e Sónia Carvalho, por terem estado sempre disponíveis para discutirem todos os processos deste trabalho, pelos seus seus conselhos e ajuda que me forneceram ao longo destes meses e pelo apoio que me deram nesta importante etapa.

Um grande obrigada aos meus amigos e colegas de Mestrado, por ter sido um grande apoio e pelos bons momentos que passámos.

E finalmente o maior agradecimento vai para os meus pais, que são as pessoas que mais acreditam em mim e nas minhas capacidades, e que mais me apoiaram. Obrigado por acreditarem em mim, mesmo quando nem eu própria acreditava.

Statement of integrity

I declare that this dissertation is the result of my personal and independent research. Its content is original, and all sources listed in the bibliographic references were consulted and are duly mentioned in the text. I further declare that all scientific and technical references relevant to the development of the work are duly cited and included in the bibliographic references.

Lisbon, September 6, 2024

Resumo

Atualmente as redes sociais são o tipo de media mais utilizado para a visualização de conteúdo, que pode ser notícias, informações ou entretenimento. Como tal, estamos mais expostos a informações e influências provenientes de outros. A influência que uma pessoa exerce noutra está relacionada com vários fatores, desde o tipo de interesse que a pessoa tem no assunto, quem é a pessoa que está a transmitir a informação e que tipo de relação se tem com ela. Existem pessoas mais influenciáveis que outras, e pessoas com capacidade de influenciar um maior número de pessoas. As pessoas que têm a capacidade de influenciar um grande número de pessoas, são chamadas de *influencers*.

Hoje o termo *influencer* está associado a um trabalho, em que estes se aliam a marcas ou empresas de forma a conseguir que os utilizadores adquiram os produtos ou serviços apresentados. À técnica de escolher um grupo de *influencers* de forma que a informação escolhida chegue ao maior número de pessoas é dado o nome de Marketing Viral.

Neste trabalho exploramos alguns dos métodos já existentes que estudam o processo de propagação de informação e influências. Em primeiro lugar apresentamos o problema de Maximização de Influência, um dos modelos utilizados na sua modelação e como este está relacionado com o Marketing Viral. De seguida, nomeamos dois modelos de epidemiologia e relacionamos a propagação de doenças com a propagação de informação. O terceiro capítulo explora um modelo que propagação de informação que tem em consideração o interesse que os utilizadores têm na mensagem que está a ser propagada. O capítulo seguinte foca-se na identificação de *influencers* nas redes sociais, através de dois métodos: medida EVC e algoritmos de Maximização de Influência. Finalmente, no último capítulo são apresentadas as conclusões deste trabalho e os desenvolvimentos a fazer para trabalhos futuros.

Palavras chave

Influencers 1; Marketing Viral; Redes Sociais 3; Maximização de Influências 4; Modelos de Propagação 5.

Abstract

Nowadays, social media is the most used type of media for viewing content, which can be news, information or entertainment. As such, we are more exposed to information and influences from others. The influence that one person exerts on another is related to various factors, from the type of interest the person has in the subject, who the person transmitting the information is and what kind of relationship they have with them. There are people who are more influenceable than others, and people who are able to influence a greater number of people. People who have the ability to influence a large number of people are called *influencers*.

Today, the term *influencer* is associated with a job in which *influencers* team up with brands or companies in order to get users to buy the products or services presented. The technique of choosing a group of *influencers* so that the chosen information reaches the greatest number of people is called Viral Marketing.

In this paper we explore some of the existing methods that study the process of spreading information and influences. Firstly, we present the Influence Maximization problem, one of the models used to model it and how it is related to Viral Marketing. Next, we name two epidemiology models and relate the spread of diseases to the spread of information. The third chapter explores an information propagation model that takes into account the interest users have in the message that is being propagated. The next chapter focuses on identifying *influencers* on social networks, using two methods: EVC measure and Influence Maximization algorithms. Finally, the last chapter presents the conclusions of this work and the developments to be made for future works.

Key words

Influencer 1; Viral Marketing 2; Social Networks 3; Influence Maximization 4; Propagation Models 5.

Contents

List of Figures

List of Tables

Symbols and abbreviations

Symbology

Latin

G	Network
V	Node set
E	Edge set
W	Weight set
$G_{(n,p)}$	Erdös-Rényi network
n	Nodes of the network
p	Probability of connection in a Erdös-Rényi network
$N(x_i)$	Neighborhood of node x_i
$f(x_i(t))$	Influence activation mapping
$\mathcal{H}$	Seed set
$c_D(x_k)$	Unweighted degree centrality of node x_k
$c_C(x_k)$	Unweighted closeness centrality of node x_k
$c_B(x_k)$	Unweighted betweenness centrality of node x_k
$c_{eig}(x_k)$	Unweighted eigenvector centrality of node x_k
$G_{(n,n_0,m)}$	Barabási-Albert random networks
n_0	Number of initial nodes of a Barabási-Albert random network
m	Number of edges added in each iteration of a Barabási-Albert random network
a_n	Affinity of a node
A_T	Affinity threshold
$(r_1)_n$	Number of messages sent by node n
q	Fraction of nodes with zero affinity
$H_{\alpha,\beta}$	Normalization constant
$diam(G)$	Diameter of the network
$\epsilon(u)$	Eccentricity of node u
$trans(G)$	Transitivity of the network
d_i^w	Weighted degree centrality of node i
b_i^w	Weighted betweenness centralit of node i
c_i^w	Weighted closeness centrality of node i

$m(A)$	Mass function
w_i	Strength of node i
k_M	Maximum degree
k_m	Minimum degree
w_M	Maximum strength
w_m	Minimum degree
$m_{di}(h)$	Probabilities of high influence for the degree
$m_{di}(l)$	Probabilities of low influence for the degree
$m_{wi}(h)$	Probabilities of high influence for the strength
$m_{wi}(l)$	Probabilities of low influence for the strength
$M_d(i)$	BPA of node i for the degree
$M_w(i)$	BPA of node i for the strength
$M(i)$	Influence value of node i
$M_i(h)$	Probability of high for node i
$M_i(l)$	Probability of low for node i
$evc(i)$	EVC value for node i
M	Diffusion model

Greek

θ	Threshold value
$\delta(x_i)$	Degree of the node x_i
β	Infection rate
γ	Recovery rate
τ	Epidemic threshold
β_s	Power-law distribution parameter for the seed set
β_v	Power-law distribution parameter for the remaining nodes
α	Power-law distribution parameter
Ω	Frame of discernment
2^Ω	Power-set
$\sigma_M(S)$	Expected number of influenced nodes

Abbreviations

WOM	Word-of-mouth
PWOM	Positive word-of-mouth
NWOM	Negative word-of-mouth
LTM	Linear Threshold Model
ICM	Independent Cascade Model
ER	Erdös-Rényi
SIR	Susceptible-Infected-Recovered
SIS	Susceptible-Infected-Susceptible
BA	Barabási-Albert
MAM	Message Affinity Model
BPA	Basic probability assignment
EVC	Evidential centrality
CELF	Cost Effective Lazy Forward
SPM	Shortest-Path Model
WIC	Weighted Independent Cascade

Capítulo 1

Introduction

1.1 Introduction

The contemporary world is characterized by the pervasive influence of the internet, which facilitates the dissemination of a vast array of information on a near-continuous basis. The ease with which information can be accessed and shared has profound implications for our social interactions, financial decisions, political opinions and even our general outlook. Prior to the advent of the internet, our opinions were shaped by various forms of traditional media, including television and radio advertising, as well as by our personal networks.

Prior to the advent of television and radio, the primary mode of disseminating information was through word-of-mouth (WOM), which is the act of transferring information verbally in an informal, person-to-person manner (Argaiz, 2015). Nowadays, WOM is adapting to the current reality created by the online media.

The term "social media" is defined as websites and applications that enable users to create and share content, or to participate in social networking. Social media allows users to share their opinions and thoughts, and the natural human inclination to share information in a viral way makes users want to gain reputation, influence, trustworthiness, or popularity (Bruyn and Lillien, 2008). WOM is the primary method of information diffusion in the online world, and thus plays a significant role in our everyday lives. Studies such as those cited in (Dye, 2000; Berry and Keller, 2003) demonstrate that WOM influences purchasing decisions, accounting for approximately two-thirds of the United States economy. Furthermore, WOM is of significant importance in the context of sales and customer value, (Kumar et al., 2007; Schmitt et al., 2011) opinion formation, (Lerman and Galstyan, 2008) rumor spreading in social networks, (Barrat et al., 2008; Castellano et al., 2009) and determining the influence of an individual within their social neighborhood (Watts and Dodds, 2007). These are all areas that can be informed by an understanding of WOM. Each individual is influenced or influences someone, to a greater or lesser extent. Different people hold several levels of influence, which are related to their social status or the direct relationship with other users. The users with the greatest influence, that is, those who can influence a large number of people, are usually referred to as *influencers*.

The use of online WOM makes information more influential due to its speed, convenience,

one-to-many reach and absence of face-to-face human pressure (Phelps et al., 2004). Online users demonstrate fewer inhibitions, display less social anxiety, and exhibit less public self-awareness than the face-to-face ones and that results in users being more willing to share personal information and be more honest about their point of view (Argaiz, 2015). Online WOM utilizes written text, which is advantageous as it allows individuals to process information at their own pace. Text can also transmit information in a more intact manner and makes it sound more formal and credible. Another benefit of online WOM is that information remains archived and accessible for a long time, which is particularly appealing to companies as it can be used in marketing activities. For example, the density of online ratings (Dellarocas and Narayan, 2006), that is, the ratio of number of people that posted online ratings for a product during a given period of time versus the number of people who bought that product in that period. In a society dominated by audiovisual culture, online WOM is the perfect way of spreading images and video content. This new way of receiving information changed the classical interpersonal communication archetype (sender-message-receiver) and introduced a new communicator form: the forwarder and the transmitter (Argaiz, 2015). This makes influence easier to spread because one does not need to have new ideas or thoughts, users can use information that already exists and pass it along the network.

The term *influencer* is currently used to describe an individual who has a significant impact on the opinions and behaviors of others through the use of social media platforms. Their influence is not solely limited to the promotion of services or products, but also encompasses a wide range of online activities, including the spread of rumors, fads, fashions, opinions, innovations, hoaxes, marketing messages and cultural trends. Depending on the objective of the disseminated message, online WOM can be either positive (PWOM) or negative (NWOM). Furthermore, depending on the employed dissemination tool, there is transitory online WOM (instant messages, email), indirect feedback (social bookmarking, rating, voting, product reviews) or searchable online WOM (blog postings and comments, wikis, forums, social networking sites) (Argaiz, 2015).

Brands and companies use *influencers* to introduce a certain product, service or even the company itself. This strategy is called Viral Marketing. This term was coined by venture capitalists Steve Jurvetson and Thomas Draper (Jurvetson, 2000) to designate Hotmail's practice of including an ad of their free-email service in each email sent. That strategy alone increased service users base from 0 to 12 million in just 18 months and that made Jurvetson equated it to "network virus."

The analogy, of the word viral, with biology remained and practitioners and scholars would describe the viral marketing process as a "pass from a customer to the next like a rampant flu virus" (Montgomery, 2001), or would constitute the "newest recognized type of virus whose transmission does not differ from that of the biological ones except for the structure of the networks they exploit" (Boase and Wellman, 2001). The disease propagation analogy, brought the notion of exponential growth and that was used by marketeers as a selling point: "Viral marketing is a compounding function. A marketeer does something and then a consumer tells

five or ten people. And it repeats. And grows and grows. Like a virus spreading through a population" (Godin, 2001).

Marketeers realized that most of the time, the majority of viral tactics did not work, and that provided multiple definitions for the term Viral Marketing depending on the author's focus: some view it as the "aggregate of all person-to-person communication about a product" (Rosen, 2000), others highlight its potential network effects because "the value of the virus to the original consumer is related to the number of other users it attracts" (Modzelewski, 2000), point to the fact that Viral Marketing "leverages the considerable power of individuals to influence others" (Subramani and Rajagopalan, 2003), or praise the trust value it conveys by "encouraging honest communication among consumers" (Phelps et al., 2004) or, focusing on the media used, "the creation of entertaining or informative items to be passed along electronically" (WOM, 2005). Based on the online WOM concept, Viral Marketing can be defined as intentional online WOM with a commercial purpose (Argaiz, 2015).

The primary objective of viral marketing, as elucidated in the various approaches presented above, is to enhance the impact of commercial messages. The underlying premise is that these messages disseminate throughout a population that receives them, thereby leading to an increase in the purchase of the goods or services being promoted or advertised.

For viral marketing to be employed as intended, it is essential that, as soon as a person receives a commercial message, they disseminate it to other individuals, thereby increasing its dissemination. It is not always certain that the forwarding of the message occurs, therefore, it is crucial to identify the factors that influence this decision. In order for a commercial message to be resend in a positive manner, it is generally necessary that the consumer has a need to test and recommend it, wants to gain more knowledge about the offer, and finally, the consumer knows the offer but lacks sufficient information to understand it.

The factors mentioned above are not always sufficient for the decision to be positively taken and, that being said, it is important to take into account some external factors that may condition the decision for the dissemination of information among people (Argaiz, 2015):

- The relationships between individuals involved in the diffusion: when someone receives a message and the only information they have is the receiver, this knowledge can completely condition decision-making, like whether or not to resend the message;
- The topology of the social network inherent to its individuals: the social network in which the message was received completely conditions the decision of whether or not to forward the message, since everything will depend on the connections of that network;
- Understanding the message that is being transmitted: the decision to resend or not a message will also depend on the individual's understanding. This understanding will determine the decision to resend it or not;
- The existence and impact of *influencers*: not all individuals have the same tendency to transmit a certain message. *Influencers* tend to make their decision to convey the message positive.

In this work we are interested in studying how influence propagates between users and the role these *influencers* have in the propagation process. What characteristics an *influencer* has in a social network and how can we choose them in order to increase the influence propagation.

In the literature there are a wide range of works that are interested in the study and on the simulation of this phenomenon. Several papers, such as (Choudhury et al., 2010; Rodriguez et al., 2010; Gomez-Rodriguez et al., 2011) study explanatory models - see (Guille et al., 2013) which study the information propagation process. Other type of models are the predictive models where is simulated the propagation traces in the network - see (Granovetter, 1978; Goldenberg et al., 2001; Kempe et al., 2003, 2015).

Another fundamental topic that we will address is the influence maximization problem. This problem concerns a set of k users who can trigger the diffusion process throughout the network. It is known that this is a NP-Hard problem, as stated in reference (Kempe et al., 2003). There are different approaches to this problem, although the quality of the initial set, which is usually called a seed set, is not always guaranteed. Indeed, the majority of existing solutions employ solely the network structure to identify users. However, as demonstrated by Goyal *et al*. (Goyal et al., 2012), these solutions are not optimal. While the network structure is useful for identifying well-positioned users, it fails to account for the possibility that those users may not be active. Consequently, it is essential to utilize more data and consider additional influence aspects of each problem.

1.2 Structure and goals of the report

This report will examine three distinct topics pertaining to the existing approaches to the information propagation process. In Chapter 2, the first topic will be the Influence Maximization problem, which aims to identify a set of k influencers that maximize the propagation process. In the same chapter, the first Section will also present a definition of the seed set, the procedure of the Linear Threshold model and some results obtained that demonstrate the significance of the chosen seed set. The subsequent Section will introduce two epidemic models, namely the SIR and SIS model. In this Section, the seed set will be redefined for these cases, the probabilistic procedures created for both models will be presented, and the similarities between the social influence propagation process and the disease propagation process will be established through the presentation of simulations obtained with them. To conclude the chapter, we will introduce some centrality measures that identify the *influencers* of a network based on its topology.

In Chapter 3, we will examine a model that simulates the propagation of information in real-life networks. This model will be used to illustrate the concept of viral marketing, which is the propagation of information through social networks. We will present the characteristics of this model and demonstrate its applicability to social networks. To illustrate the model's utility, we will present examples of different social networks and identify the most influential nodes according to the model.

The following chapter is dedicated to the identification of *Influencers*. This is achieved

through the use of two distinct tools: the EVC measure and the Influence Maximization algorithms. The chapter comprises two phases. The first compares the results obtained using the EVC measure with those obtained using the weighted centrality measures. The second presents a comparison between the results yielded by the Influence Maximization algorithms and those obtained using the unweighted centrality measures.

The last chapter presents the conclusions drawn throughout the course of this research project, offering a reflective analysis of the results obtained from the experiments conducted. Furthermore, this chapter outlines avenues for future research, including potential topics for future works.

Capítulo 2

Propagation Models

In this chapter, we introduce the Influence Maximization Problem and elucidate its most salient features. We define the seed set, emphasis on the importance of this set in this context, and present some results. Furthermore, we illustrate the procedure described in (Rocha et al., 2023c) for the Linear Threshold model in an Erdös-Rényi random network.

The following Section is dedicated to two of the most prominent epidemic models, namely the SIR and SIS models, and their relationship with the information propagation process. In this Section, we will redefine the seed set for this problem, demonstrate the probabilistic procedures of both models as presented in references (Rocha et al., 2023a,b), and present some of the results obtained, which once again highlight the significance of the seeds in this type of problem.

In the final Section of this chapter, we will introduce a number of centrality measures based on the topology of a network and demonstrate how they can be applied in social networks to identify *influencers*.

2.1 Influence Maximization Problem

The emergence and significance of social networks in recent times is a topic worthy of further investigation. Social networks represent the manner in which ideas and information spread in modern society and could be represented through a graph. As previously mentioned, a considerable portion of our opinions and decisions are directly or indirectly influenced by our social contacts (Ibarra and Andrews, 1993; Bruning et al., 2020; Rocha et al., 2023c).

Consequently, it is possible to apply dynamic diffusion models to a selected network in order to gain insight into this phenomenon. These propagation processes are referred to as social contagion, as they bear resemblance to the manner in which a disease is transmitted between individuals in a population (Rocha et al., 2023c). A plethora of studies have been conducted on this subject, see (Rodriguez et al., 2010; Myers et al., 2012; Zhang and Li, 2022), yet this Section is mainly based on the work presented in (Rocha et al., 2023c).

Influence Maximization was first proposed in (Domingos and Richardson, 2001), which was later interpreted as a discrete optimization problem. This typification was a landmark in

the research on Influence Maximization and was soon proved to be an NP-Hard problem (class of problems that have at least exponential complexity, see (Kempe et al., 2003)), when the given information propagation model is an independent cascade or a linear threshold model.

The Linear Threshold Model (LTM) was initially proposed by Granovetter (Granovetter, 1978) to model collective behavior in which binary decisions are observed, such as the diffusion of diseases or rumors. It has been utilized to model information propagation processes in social networks. The Independent Cascade Model (ICM) was introduced in the context of marketing by Goldenberg (Goldenberg et al., 2001), who drew inspiration from works in interacting practical systems and probability theory. Both the LTM and ICM were employed by Kempe (Kempe et al., 2003) to simulate the dissemination of information in social networks.

The LTM and ICM have some similarities, particularly, in both models is considered a social graph, represented by $G = (V, E)$ where the nodes have two possible states: active or inactive. A node $x \in V$ is said to be active if it receives the information and accepts it and x is inactive if it does not receive the information or rejects it. An inactive node becomes active, if it receives and accepts the message. In the LTM, there is an associated weight $w(u, v)$ to each edge (u, v), which represents the contribution of each neighbor to the node, the higher the weight, the higher the influence on the node (Menczer et al., 2020), and a threshold θ to each node u. The values of θ can depend on u. A node u will be activated if the sum of weights, between u and its activated neighbors, is at least θ, that is:

$$\sum_{v:\text{active}} w(u, v) \geq \theta.$$

The threshold θ is randomly chosen from the interval $[0, 1]$ and represents the tendencies of the nodes to accept the message delivered by its neighbors (Jendoubi, 2016). This case concerns a weighted LTM. In a unweighted LTM, it is necessary that the fraction of activated neighbors of a node u exceeds its threshold value θ_u, that is:

$$\frac{1}{n_u^{active}} \geq \theta_u,$$

where n_u^{active} represents the number of active neighbors of u.

A substantial number of publications on this topic have been presented, with a particular focus on two categories of classical Influence Maximization algorithms: greedy algorithms and heuristic algorithms, see (Chen et al., 2009, 2010, 2012; Li et al., 2018). On one hand, greedy algorithms exhibit high accuracy but are computationally time-consuming. On the other hand, heuristic algorithms are more efficient but may sacrifice accuracy (Rocha et al., 2023c). A significant proportion of existing studies are based on static networks. However, there are also researchers who have devoted their attention to the study of Influence Maximization in dynamic networks. For further reading on this topic, please see the following references: (Hao et al., 2011; Teng et al., 2021; Li et al., 2021; Zhang and Li, 2022).

In the following Sections, we will be examining the influence activation mapping for the LTM, which incorporates an activation threshold θ, as defined in (Granovetter, 1978). For this case we are going to consider Erdös-Rényi random networks as the diffusion networks.

Definition 1 *An Erdös-Rényi (ER) random network is characterized by having n nodes, with $n \in \mathbb{N}$, and each pair of nodes is linked by a connection or edge with a certain probability $p \in [0,1]$. They are usually represented by $G_{(n,p)} = (V, E)$, where $V = \{x_1, x_2, .., x_n\}$ is the node set and represents the individuals, $|V| = n$ indicates that there are a total of $n \in \mathbb{N}$ individuals, and the edge set E represents the connections between different individuals.*

The edges between the nodes are independently and randomly generated with the same probability p, see (Erdös and Rényi, 1959, 1960).

Information propagates through the connections in the network. We denote by $N(x_i)$ the neighborhood of x_i, i.e., the set of nodes of V that are linked to x_i, and by $\delta(x_i) = |N(x_i)|$ the degree of the node x_i with $i = 1, ..., n$. For more details on ER networks, see for example (Bollobás, 2001; Newman, 2010; Barabási, 2016; Hofstad, 2016) and references therein. We consider the LTM, where every node at each instant has one of two possible states, inactive or active, see (Shakarian et al., 2015a,b). We deem the network as unweighted and a constant threshold value for every node. The choice of the threshold value θ is a very important feature in terms of linear threshold based influence maximization. This is one of two vital points, along with the degree distribution of the network, see (Talukder et al., 2019). These considerations lead to the following definition, for more, see (Chakrabarti et al., 2008).

The approach for the activation threshold is based in the network topological entropy concept used in (Rocha and Caneco, 2013; Rocha et al., 2015; Rocha and Carvalho, 2021, 2023). The topological entropy of the ER network characterizes the topological dynamics of this type of network through the spectral radius λ_A of the corresponding adjacency matrices A. Considering this information, the definition for the activation threshold θ is the following:

Definition 2 *(Rocha et al., 2023c) Let $G_{(n,p)}$ be an Erdös-Rényi random network, A the adjacency matrix associated with the spectral radius λ_A. The activation threshold θ of the network $G_{(n,p)}$ is defined by*

$$\theta = \frac{1}{np + O(\sqrt{np})} = \frac{1}{\lambda_A}. \tag{2.1}$$

Considering this characterization of the epidemic threshold for ER networks, given by Equation (2.1), and attending to the monotony of the rational function, we can conclude that the epidemic threshold θ decreases with the growth of p and when the network order n is fixed and sufficiently large (Rocha et al., 2023c).

As we can see, the activation threshold depends on the network order n and the connection probability p. For this case we are going to consider the network to be in one of these two topological regimes: supercritical, $\frac{1}{n} < p < \frac{ln(n)}{n}$, or connected, $\frac{ln(n)}{n} < p < 1$. In the supercritical regime the giant component may have cycles and the other connected components are mainly trees, while in the connected regime the order of the giant component is almost n (Rocha et al., 2023c).

In the LTM, a node x_i becomes active if and only if the ratio between the active neighbors of x_i at time t ($\epsilon(x_i(t))$) and the degree of that node ($\delta(x_i)$), is higher than the threshold value θ. The following definition introduces the concept of influence activation mapping in a network, in particular for the ER random network $G_{(n,p)}$.

Definition 3 *(Rocha et al., 2023c) Let $G_{(n,p)}$ be an Erdös-Rényi random network, with $x_i(t) \in V \times \mathbb{N}_0$ being a node $x_i \in V$ at discrete time $t \in \mathbb{N}_0$ for $i = 1, 2, ..., n$ and θ being the activation threshold provided by Equation* (2.1)*. The influence activation mapping $f : V \times \mathbb{N}_0 \longrightarrow \{0,1\}^n$, where*

$$f(x_1, x_2, ..., x_n)(t) = (f(x_1(t)), f(x_2(t)), ..., f(x_n(t))),$$

is defined as follows:

$$f\left(x_i(t)\right) = \begin{cases} 1, & if \ \dfrac{\epsilon\left(x_i(t-1)\right)}{\delta\left(x_i\right)} \geq \theta \\ \\ 0, & if \ \dfrac{\epsilon\left(x_i(t-1)\right)}{\delta\left(x_i\right)} < \theta \end{cases} . \tag{2.2}$$

The influence activation mapping f is defined according to the local dynamics of each node $x_i \in V$ of the network $G_{(n,p)}$ under analysis.

When considering the influence activation mapping f provided by Equation (2.2) the mapping of the active neighbors of a node can be recursively defined, $x_i, \epsilon : V \times \mathbb{N}_0 \longrightarrow \mathbb{N}_0$, in terms of the influence activation mapping f in the following way:

$$\epsilon(x_i(t)) = \sum_{x_j \in N(x_i)} f(x_j(t)) \tag{2.3}$$

with $x_i \in V$ for $i = 1, 2, ..., n$ and for each discrete time $t = t_0$, where t_0 represents the initial instant of the diffusion process.

Let us consider the seed set as the subset of V that specifies the initial conditions for the recursion mapping of the active neighbors, as provided by Equation (2.3). This tell us which nodes are activated at the initial state of the diffusion process. The concept of a seed set already exists, see for example (Shakarian et al., 2015a,b) and references therein. However, the next definition of the seed set is characterized by the influence activation mapping f at the initial instant t_0.

Definition 4 *(Rocha et al., 2023c) Let $G_{(n,p)}$ be a Erdös-Rényi contact network and let $f(x_i(t_0))$ be the node state at the initial instant t_0, with $x_i \in V$. The seed set $\mathcal{H}$ of the network $G_{(n,p)}$ is defined by*

$$\mathcal{H} = \{x_i \in V : f(x_i(t_0)) = 1\},$$

where $\emptyset \neq \mathcal{H} \subset V$.

In (Rocha et al., 2023c) is presented the theoretical procedure for the linear threshold model applied to ER random networks. As previously mentioned, in the LTM, a node is activated if and only if the influence exerted on it by its neighbors exceeds a certain value. This means

that if the value exceeds the threshold θ for a specific node, the node becomes active. This indicates that it has adopted an idea, obtained information, see (Shakarian et al., 2015a) and references therein.

Procedure: Let $G_{(n,p)} = (V, E)$ be an ER random network, f the influence activation mapping provided by Equation (2.3), and $\mathcal{H}$ the seed set provided by Definition 4. The theoretical procedure of the LTM for the network $G_{(n,p)}$ follows the next steps:

1. At the initial instant $t = t_0$, the nodes in the seed set $\mathcal{H} \subset V$ are all active, i.e.,

$$f(x_i(t_0)) = 1, \forall x_i \in \mathcal{H};$$

2. At instant $t^* \neq t_0$, if the node $x_i \in V \setminus \mathcal{H}$ is activated, then the node x_i remains activated, i.e.,

$$\text{if } \exists t^* \neq t_0 : f(x_i(t^*)) = 1, \text{ then } f(x_i(t)) = 1, \forall t \geq t^*;$$

3. If at instant $t^* \neq t_0$ a node $x_i \in V \setminus \mathcal{H}$ is not activated, then the influence activation function is verified as provided in Equation (2.2);

4. At each instant $t \neq t_0$, steps 2 and 3 are repeated until reaching the following stopping criteria:

$$(f(x_1(t)), f(x_2(t)), ..., f(x_n(t))) = (f(x_1(t-1)), f(x_2(t-1)), ..., f(x_n(t-1)))$$

   i.e., at instant t the vector at iteration t is equal to the vector in the previous iteration.

The procedure above has a finite number of steps. In step 1, the seed set $\mathcal{H}$ corresponds to the active nodes in the beginning of the process. Step 2 guarantees that an active node remains in that state throughout the procedure. In Step 3 the nodes that are not activated can change their state. And finally, step 4 provides the stopping condition for the procedure.

The next images show how the choice of the seed element can alter the influence propagation process. Both images show the same random network with the same parameters, i.e. $n = 100$ nodes, $p = 0.05$, a threshold value $\theta = 0.2$ throughout the same number of iterations, which were twenty.

By looking at Figure 2.1, we can see that the seed element chosen for this case was the node x_{76} and that the process stopped after two iterations, where that same node was the only one in the activated state. However, in Figure 2.2 where node x_{17} is the seed element, we see that the process stopped after nine iterations and in the end only one node, x_{19}, remained inactive.

(a) Initial instant $t = t_0$ **(b)** Instant $t = t_0 + 2$

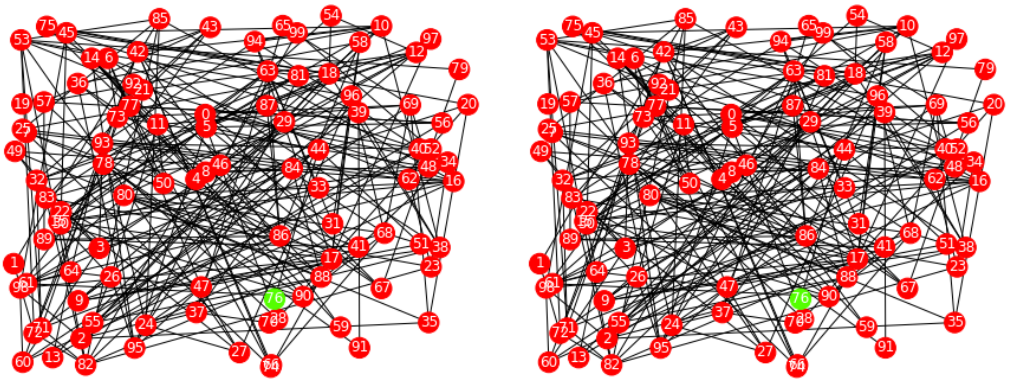


Figura 2.1 **(a)** Erdös-Rényi network $G_{(n,p)}$ with $n = 100$ and $p = 0.05$ at instant $t = t_0$, where $\theta = 0.20$ is the activation threshold and the active node x_{76} the seed; **(b)** at instant $t = t_0 + 2$, the procedure verifies step 4, i.e., the stopping condition; the diffusion process ended after two iterations.

(a) Initial instant $t = t_0$ **(b)** Instant $t = t_0 + 9$

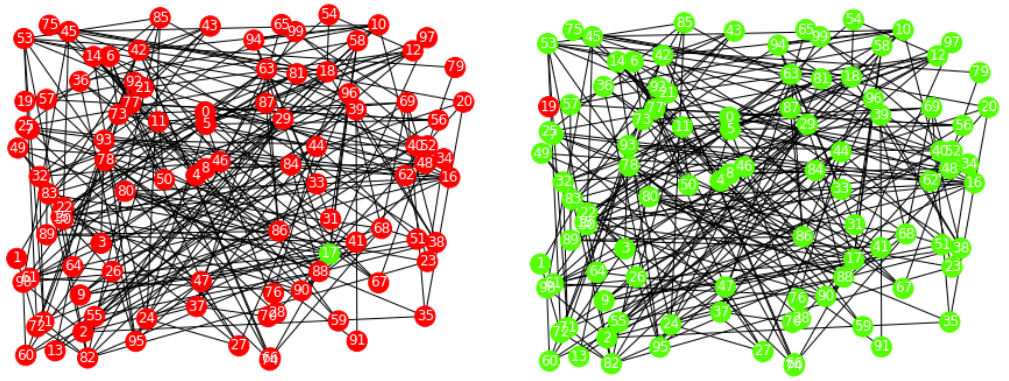


Figura 2.2 **(a)** Erdös-Rényi network $G_{(n,p)}$ with $n = 100$ and $p = 0.05$ at instant $t = t_0$, where $\theta = 0.20$ is the activation threshold and the active node x_{17} the seed; **(b)** at instant $t = t_0 + 9$, the procedure verifies step 4, i.e., the stopping condition; the diffusion process ended after nine iterations.

One of the most popular applications of the Influence Maximization Problem is in Viral Marketing. The main purpose of this problem is to detect a set of k *influencers* (the seed set) in a social network that are able to trigger a large cascade of influence. This set will receive the Viral Marketing message and propagate it through the network (Jendoubi, 2016).

2.2 Probabilistic Approach to Epidemic Models

This Section, like previously mentioned, is going to focus on the epidemic models and their propagation process and the results presented were obtained through the works of (Rocha et al., 2023a,b). Epidemic theoretical models are employed to investigate the dissemination of a specific disease within a population. These models enable us to comprehend the mechanisms underlying this process and to identify potential strategies for its mitigation. In the contemporary era, the advent of technological innovations has led to the emergence of diverse forms of contagion, including computer viruses and the proliferation of social media, which has given rise to phenomena such as the spread of rumors, fake news, and conspiracy theories (Rocha et al., 2023a).

The dissemination of information exhibits many similarities with epidemic processes, as evidenced by the work of (Pastor-Satorras et al., 2015; Menczer et al., 2020) and the references therein. In this context, we will introduce two classical models for this type of problem: the SIR and SIS models.

2.2.1 The SIR Model

The Susceptible-Infected-Recovered (SIR) model is a compartmental model, which means that the population is divided in partitions, where every individual, at each instant t, belongs to only one of the possible subsets, see (Ross and Hudson, 1917; Kermack and McKendrick, 1927; Daley and Gani, 1999; Diekmann and Heesterbeek, 2000; Okabe and Shudo, 2021). This model is a classical model for disease spread, where all the individuals of the population are in one of three possible states: susceptible (S) (able to be infected), infected (I) or recovered (R) (no longer able to infect or be infected), see (Anderson and May, 1979; Allen, 2008; Kitsak et al., 2010; Macdonald et al., 2012; Pastor-Satorras et al., 2015; Kiss et al., 2017) and references therein.

At each instant $t \in \mathbb{N}$, the individuals in the infected state are able to infect the individuals that lie in a susceptible state, with probability β. On the other hand, γ is the proportion of infected individuals at time t which recover at time $t+1$. The β and γ, previously mentioned, are known as the infection rate and the recovery rate, respectively (Rocha et al., 2023a).

The sets for each of the states in the SIR model, in discrete time $t \in \mathbb{N}$ are represented by,

- $S(t) \equiv$ the set of individuals in the susceptible state at the instant t;
- $I(t) \equiv$ the set of individuals in the infected state at the instant t;
- $R(t) \equiv$ the set of individuals in the recovered state in the instant t.

The proportion of elements in the sets are given by,

$$s(t) = \frac{\#S(t)}{n}, \;\; i(t) = \frac{\#I(t)}{n}, \;\; r(t) = \frac{\#R(t)}{n},$$

and the population size is constant throughout the time, i.e.,

$$s(t) + i(t) + r(t) = 1, \forall t \in \mathbb{N}.$$

For the discrete epidemic model, three equations are considered, one for each group, where we can obtain the number of elements of a given group, in the next instant $t+1$, based on the elements of instant t (Rocha et al., 2023a).

$$\begin{cases} s(t+1) = s(t) - \beta(t)s(t)i(t) \\ \\ i(t+1) = i(t) + \beta(t)s(t)i(t) - \gamma(t)i(t) \\ \\ r(t+1) = r(t) + \gamma(t)i(t) \end{cases}.$$

Both the infection rate and recovery rate are considered constant throughout time:

$$\beta(t) \equiv \beta \text{ and } \gamma(t) \equiv \gamma.$$

For the SIR model, the ratio between β and γ gives information about the behavior of the propagation process throughout time. That ratio is denoted by R_0, see (Heesterbeek, 2002; Delamater et al., 2019) and references therein. The ratio $R_0 = \dfrac{\beta}{\gamma}$, also known as, the basic reproduction number, gives us information on the propagation process depending on its value:

- if $R_0 > 1$, then the number of infected individuals increases, meaning that, each infected individual has the ability to infect more than one susceptible individual, so there is a possibility of existing an epidemic;
- if $R_0 = 1$, every existing infection causes a new infection. The disease will remain alive and stable, but there will be no epidemic.
- if $R_0 < 1$ the number of infected individuals decreases, which causes the propagation to be slower and its natural eradication over time.

As in the previous Section, the procedure of this model needs an initial set of infected nodes, that is a seed set. For this case the seed set is defined as:

Definition 5 *(Rocha et al., 2023a,b) Let $G_{(n,p)}$ be a Erdös-Rényi contact network, $I(t)$ be the set of individuals in the infected state at instant t and t_0 be the initial instant. The seed set $\mathcal{H}$ of the network $G_{(n,p)}$ is defined by*

$$\mathcal{H} = \{x_i \in V : x_i \in I(t_0)\} \tag{2.4}$$

where $\emptyset \neq \mathcal{H} \subset V$.

For these cases, the ER networks will also be in the supercritical or connected regimes, therefore, according to (Rocha et al., 2023a,b) the epidemic threshold τ can be defined the following way:

Definition 6 *(Rocha et al., 2023a,b) Let $G_{(n,p)}$ be a Erdös-Rényi contact network and A the adjacency matrix associated with the spectral radius λ_A. The epidemic threshold τ of the network $G_{(n,p)}$ is defined by,*

$$\tau = \frac{1}{\lambda_A}. \tag{2.5}$$

Probabilistic Procedure for the SIR model: Let us consider a ER contact network $G_{(n,p)}$, with $0 \leq p \leq 1$, where $V = \{x_1, ..., x_n\}$ is the set of n nodes, $\mathcal{H}$ the seed set given by Equation (2.4), and $N(x_i)$ is the neighborhood of x_i, i.e., the set of nodes in V connected to x_i. It is assumed that, at each step, the infection and recovery events are independent of each other. The probabilistic procedure follows the next steps:

- **Step 1:** At the initial instant $t = t_0$, with $\mathcal{H} = \{x_i \in V : x_i \in I(t_0)\}$:
  If $x_i \in \mathcal{H}$:

$$\begin{cases} \mathbb{P}\left[x_i \in S(t_0+1) \mid x_i \in I(t_0)\right] = 0 \\ \mathbb{P}\left[x_i \in I(t_0+1) \mid x_i \in I(t_0)\right] = 1-\gamma \\ \mathbb{P}\left[x_i \in R(t_0+1) \mid x_i \in I(t_0)\right] = \gamma \end{cases} \tag{2.6}$$

  and if $x_j \in N(x_i)$, with $x_i \in \mathcal{H}$ and $i \neq j$, then:

$$\begin{cases} \mathbb{P}\left[x_j \in S(t_0+1) \mid x_i \in I(t_0)\right] = 1-\beta \\ \mathbb{P}\left[x_j \in I(t_0+1) \mid x_i \in I(t_0)\right] = \beta \\ \mathbb{P}\left[x_j \in R(t_0+1) \mid x_i \in I(t_0)\right] = 0 \end{cases} . \tag{2.7}$$

  At the initial instant t_0 we consider two different cases: the seed set and the nodes connected to the seeds. The first three probabilities, given by Equation (2.6), consider the first case. Since these nodes are infected, then the nodes remain infected, with probability $1-\gamma$, or recover, with probability γ. The other three probabilities, given by Equation (2.7), refer to the neighbors of the seeds. In the initial instant t_0 these nodes are susceptible,

therefore they remain susceptible, with probability $1-\beta$, or transition to the infected set, with probability β. The nodes that are not on the seed set neither in its neighborhood remain susceptible.

- **Step 2:** For $t \neq t_0$, with $t \in \mathbb{N}$:
  - If $x_j \in S(t)$, then:

$$\begin{cases} \mathbb{P}\left[x_j \in S(t+1) \mid N(x_j) \cap I(t) = \emptyset\right] = 1 \\ \mathbb{P}\left[x_j \in S(t+1) \mid N(x_j) \cap I(t) \neq \emptyset\right] = 1-\beta \\ \mathbb{P}\left[x_j \in I(t+1) \mid N(x_j) \cap I(t) = \emptyset\right] = 0 \\ \mathbb{P}\left[x_j \in I(t+1) \mid N(x_j) \cap I(t) \neq \emptyset\right] = \beta \\ \mathbb{P}\left[x_j \in R(t+1)\right] = 0 \end{cases} . \tag{2.8}$$

  - If $x_j \in I(t)$, then:

$$\begin{cases} \mathbb{P}\left[x_j \in S(t+1)\right] = 0 \\ \mathbb{P}\left[x_j \in I(t+1)\right] = 1-\gamma \\ \mathbb{P}\left[x_j \in R(t+1)\right] = \gamma \end{cases} . \tag{2.9}$$

  - If $x_j \in R(t)$, then $\mathbb{P}\left[x_j \in R(t+1)\right] = 1$.

In step 2, it is described the transitions of the individuals among the three possible sets in other instants $t \neq t_0$. If a node belongs to the susceptible group, then we are going to analyze its connections, see Equation (2.8). If one of the neighbors is infected, then the susceptible node is infected, with probability β or stays susceptible with probability $1-\beta$. In case none of its neighbors is infected, then the node remains susceptible. Considering the infected nodes at instant t, then in instant $t+1$ the nodes remain infected, with probability $1-\gamma$ or recover, with probability γ, see Equation (2.9). Finally, the case where the node belongs to the recovered group, there is no transitions in this set, if a node is recovered at instant t then the node remains recovered for every instant $t^* > t$.

- **Step 3:** The procedure stops:

$$\text{If } \forall x \in I(t),\ N(x) \cap S(t) = \emptyset \ \text{ or } \ r(t) \gg i(t).$$

This process stops when none of the infected nodes have susceptible neighbors, or when the number of recovered nodes is much greater than (represented by $\gg$) the infected ones.

The next images illustrate an example of how a disease can spread in a network according to the SIR model, depending on the chosen seed set. In Figure 2.3 we have a random network with $n = 200$ nodes, $p = 0.028$, infection rate $\beta = 0.105$ and a recovery rate $\gamma = 0.1$. The process happens during 30 iterations, the R_0 ratio is greater than 1 and the seed set was randomly selected.

In Figure 2.3 (a), we have the ER network $G_{(n,p)}$, at the initial instant $t = t_0$, where the node x_{129} is the seed element (yellow color). In the beginning of the process we have 1 infected node and 199 susceptible (red color). When we observe Figure 2.3 (b), which corresponds to iteration 9, at this point, there are individuals in all of the sets. Namely, 149 susceptibles, 42 infected and 9 recovered (green color). At instant $t = t_0 + 18$, Figure 2.3 (c), we observe that the proportion of nodes in the susceptible state is decreasing, while the proportion in the recovered state is increasing, as expected. In the last instant, the proportion of recovered nodes, is much greater than the other two, which means that the majority of the population is not able to get infected anymore and eventually the disease will disappear.

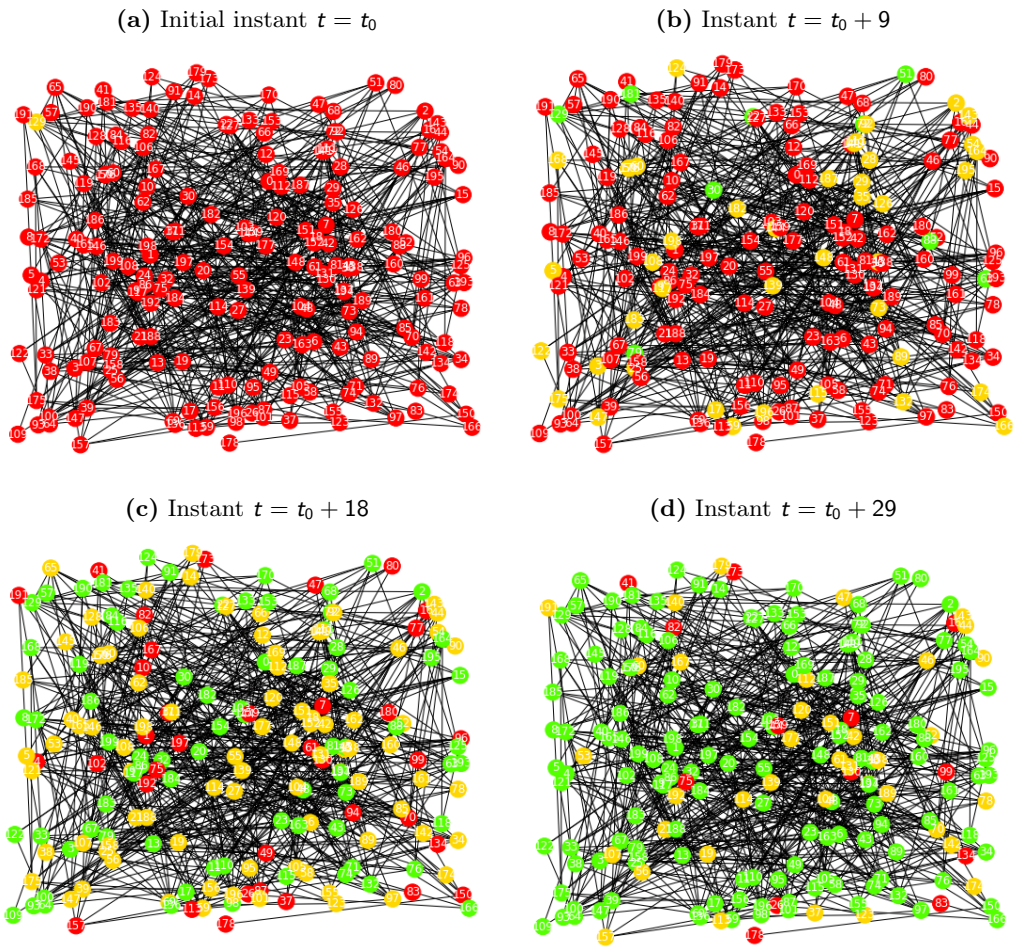


Figura 2.3 Probabilistic procedure for the SIR model over the Erdös-Rényi network $\mathcal{G}_{(n,p)}$, with $n = 200$, $p = 0.028$, $\beta = 0.105$, $\gamma = 0.1$.

However, in Figure 2.4 we have a different case. For the same network, under the same conditions the propagation did not happen. In Figure 2.4 (a) the procedure starts with the node x_{178} who has only one neighbor, node x_{150}. For this case, the process stopped in instant $t = t_0 + 2$, as showed by Figure 2.4 (b). In conclusion, in Figure 2.3 the propagation process exists, meanwhile in Figure 2.4 there is no epidemic occurring.

(a) Initial instant $t = t_0$ **(b)** Instant $t = t_0 + 2$

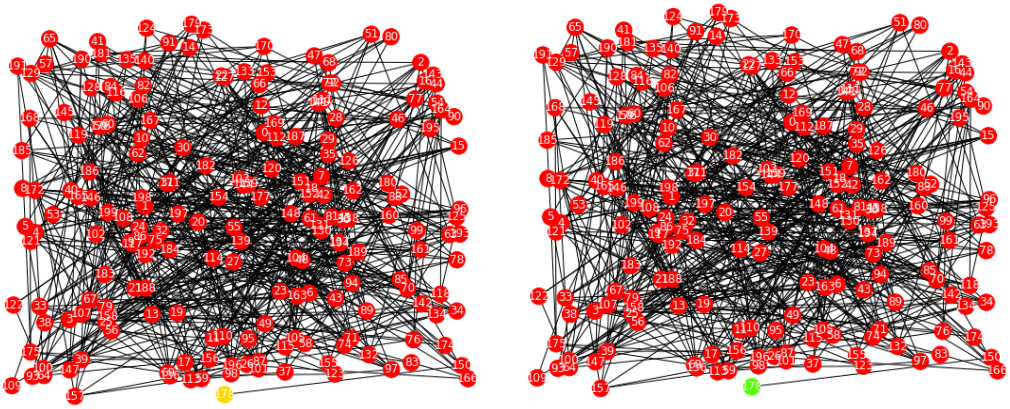


Figura 2.4 Probabilistic procedure for the SIR model over the Erdös-Rényi network $\mathcal{G}_{(n,p)}$, with $n = 200$, $p = 0.028$, $\beta = 0.105$, $\gamma = 0.1$.

2.2.2 The SIS Model

As in the preceding Section, the results presented here are based on the work of (Rocha et al., 2023a,b). The Susceptible-Infected-Susceptible (SIS) model is also a compartmental model, but in this case, the individuals lie only between two states: susceptible (S) and infected (I), see (Pastor-Satorras et al., 2015; Kiss et al., 2017). The main difference between the two models is the fact that, in the SIS model, an individual in the infected state can return to the susceptible state, with probability γ. Like in the previous model, we have a infection rate β and a recovery rate γ (Rocha et al., 2023a). Lets consider also the SIS model in discrete time, therefore we have

$$s(t) = \frac{\#S(t)}{n}, \quad i(t) = \frac{\#I(t)}{n},$$

and

$$s(t) + i(t) = s(t+1) + i(t+1), \forall t \in \mathbb{N}.$$

For the SIS discrete model, two equations are defined, where the proportion of susceptible or infected elements in the instant $t+1$ are given by the elements in instant t (Rocha et al., 2023a),

$$\begin{cases} s(t+1) = s(t) - \beta(t)s(t)i(t) + \gamma(t)i(t) \\ \\ i(t+1) = i(t) + \beta(t)s(t)i(t) - \gamma(t)i(t) \end{cases}.$$

The infection and recovery rates will also be considered constant $\beta(t) \equiv \beta$ and $\gamma(t) \equiv \gamma$, just like in the SIR model, previously presented.

Probabilistic Procedure for the SIS model: Lets consider an ER random contact network $G_{(n.p)}$, with $0 \leq p \leq 1$ and the seed set $\mathcal{H}$ given by Equation (2.4). For this case, it is also assumed that, at each step, the infection and recovery events are independent of each other. The procedure follows the next steps:

- **Step 1:** At the initial instant $t = t_0$, with $\mathcal{H} = \{x_i \in V : x_i \in I(t_0)\}$:
  If $x_i \in \mathcal{H}$, then:

$$\begin{cases} \mathbb{P}\left[x_i \in S(t_0+1) \mid x_i \in I(t_0)\right] = \gamma \\ \\ \mathbb{P}\left[x_i \in I(t_0+1) \mid x_i \in I(t_0)\right] = 1-\gamma \end{cases}. \tag{2.10}$$

  and if $x_j \in N(x_i)$, with $x_i \in \mathcal{H}$ and $i \neq j$, then:

$$\begin{cases} \mathbb{P}\left[x_j \in S(t_0+1) \mid x_j \in N(x_i) \ \wedge \ x_i \in I(t_0)\right] = 1-\beta \\ \\ \mathbb{P}\left[x_j \in I(t_0+1) \mid x_j \in N(x_i) \ \wedge \ x_i \in I(t_0)\right] = \beta \end{cases}. \tag{2.11}$$

  At the initial instant t_0, we focus on the seed set $\mathcal{H}$, which contains the infected nodes, and on the nodes that have connections with the seed set. The first two probabilities, given by Equation (2.10), refer to the nodes in the seed set, and in instant t_0+1 they become susceptible, with probability γ or remain infected with probability $1-\gamma$. In the second system of conditions, given by Equation (2.11), we consider the neighbors of the elements in the seed set. At the instant t_0 the nodes are susceptible, and they remain susceptible in instant t_0+1, with probability $1-\beta$ or get infected, with probability β.

- **Step 2:** For $t \neq t_0$, with $t \in \mathbb{N}$:
  - If $x_j \in S(t)$, then:

$$\begin{cases} \mathbb{P}\left[x_j \in S(t+1) \mid N(x_j) \cap I(t) = \emptyset\right] = 1 \\ \\ \mathbb{P}\left[x_j \in S(t+1) \mid N(x_j) \cap I(t) \neq \emptyset\right] = 1-\beta \\ \\ \mathbb{P}\left[x_j \in I(t+1) \mid N(x_j) \cap I(t) = \emptyset\right] = 0 \\ \\ \mathbb{P}\left[x_j \in I(t+1) \mid N(x_j) \cap I(t) \neq \emptyset\right] = \beta \end{cases}. \tag{2.12}$$

  - If $x_j \in I(t)$, then:

$$\begin{cases} \mathbb{P}\left[x_j \in I(t+1)\right] = 1-\gamma \\ \\ \mathbb{P}\left[x_j \in S(t+1)\right] = \gamma \end{cases}. \tag{2.13}$$

Step 2 states the probabilities for every instant $t \neq t_0$ in the following way. If a node is susceptible at instant t, there are two cases, according to the connections in the network. If the susceptible node does not have infected neighbors, then it remains susceptible. If it does, then can either remains susceptible, at the instant $t+1$ with probability $1-\beta$ or get infected with probability β. In Equation (2.12) lies also the probabilities of the infected nodes at instant t. If a node is infected, then at instant $t+1$ it remains infected, with probability $1-\gamma$ or recovers and transitions to the susceptible set $S(t+1)$, with probability γ, see Equation (2.13).

- **Step 3:** The procedure stops:

$$\text{If } \forall x \in I(t),\ N(x) \cap S(t) = \emptyset.$$

The process stops when we reach instant t, and all the infected nodes no longer have susceptible neighbors. In other words, the process stops when all of the nodes are infected.

Figure 2.5 shows 20 ER networks, during 70 iterations, with $n = 200$ individuals, $p = 0.028$, $\beta = 0.105$, $\gamma = 0.1$ and the initial infected individuals chosen randomly. We can see that all the networks behave similarly. In general, we see that both curves of this model decrease and increase in the same proportion, with the behavior of inverse sigmoid and sigmoid types, respectively. However there are cases where there is no propagation. Notice that, after some iterations, there is an oscillatory behavior, which can be interpreted as peaks of infection.

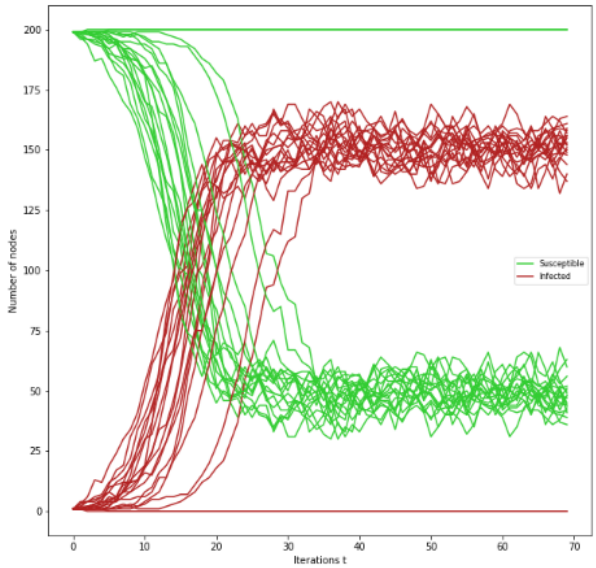


Figura 2.5 Susceptible-infected-susceptible individuals over 20 Erdös-Rényi networks $G_{(n,p)}$, with $n = 200$, $p = 0.028$ and seed nodes randomly chosen.

In conclusion, the SIR model can be used to describe the information propagation process. This model allows us to map a set of initially infected users to a set of active users (those who have received and accepted the information). These infected users attempt to infect their neighbors, as demonstrated by Jendoubi (Jendoubi, 2016). An active user is considered recovered when they are unable to purchase the product again.

As stated in Leskovec (Leskovec et al., 2007a), the issue with these models is that they assume a known social network over which the disease (information) is spreading, and that a single parameter specifies the infectiousness of the disease. In the context of information propagation, this would mean that the entire population is equally susceptible to receiving the information, which is not the case.

2.3 Centrality Measures

In a social network, depending on the information that is being spread, different users hold different levels of importance. This feature can be measured according to several parameters: local or global; deterministic, algebraic, stochastic and more.

Centrality measures are a network metric that study a node in regard to the topology of the chosen network, in other words, they assess the position of an individual within a social group. There are several types of centrality measures and each one studies a different characteristic of the nodes of the network. The degree centrality concerns the network's local dynamics, while the centrality measures such as closeness and betweenness take into account the network's global dynamics, see (Hafiene et al., 2019). However, these will be the measure approached:

Definition 7 *Let $G = (V, E)$ be a random undirected network and $x_k \in V$ with $k = \{1, 2, ..., n\}$:*

- *the degree centrality is $c_D(x_k) = \delta(x_k)$, where $\delta(x_k)$ is the degree of the node x_k;*
- *the closeness centrality is $c_C(x_k) = \dfrac{1}{\sum_{u \in V} d(u, x_k)}$, where $d(u, x_k)$ represents the distance from $u \in V$ to x_k;*
- *the betweenness centrality is $c_B(x_k) = \sum_{i<j \;\; k \neq i,j} b_{ij}(x_k)$, where*

$$b_{ij}(x_k) = \begin{cases} 0, & \text{no path between } x_i \text{ and } x_j \\ \dfrac{g_{ij}(x_k)}{g_{ij}}, & \text{otherwise} \end{cases}$$

  where g_{ij} is the number of paths between x_i and x_j, and $g_{ij}(x_k)$ the number of paths between x_i and x_j that contain x_k;

- *the eigenvalue centrality is $c_{eig}(x_k) = u_k$, where u_k is the k-th component of the unitary eigenvector of A associated with the spectral radius.*

The next Sections illustrate how the different centrality measures influence the propagation process, with both the LTM and the SIR model. The results for both models are from the works of (Rocha et al., 2023c) and (Rocha et al., 2023b), respectively.

2.3.1 Linear Threshold Model

For the influence propagation process using the LTM, we will analyze two cases, see (Rocha et al., 2023c), when the seed element corresponds to the node with the highest closeness centrality, and with the highest betweenness centrality. For both cases we will consider an ER network with $n = 100$ nodes, $p = 0.05$ and $\theta = 0.2$.

Figure 2.6 (a) shows the ER network $G_{(n,p)}$ at the initial instant $t = t_0$, where node x_{47} is the node with highest closeness centrality. The only inactive node, x_{19}, is an isolated nodes, i.e., it is not connected to any node. As we see in Figure 2.6 (b), the stopping condition is achieved at instant $t = t_0 + 12$ and the influence maximization is achieved over the entire connected component of the network $G_{(n,p)}$.

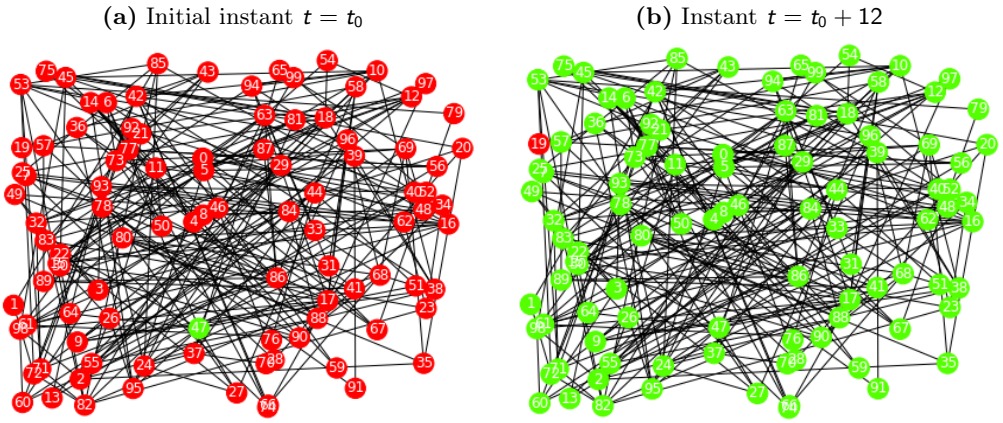
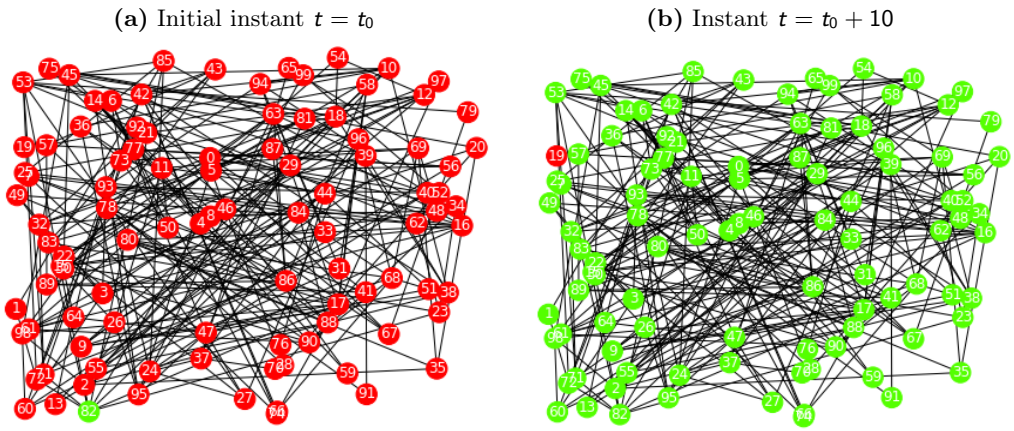


Figura 2.6 **(a)** Erdös-Rényi network $G_{(n,p)}$ with $n = 100$ and $p = 0.05$ at instant $t = t_0$, where $\theta = 0.20$ is the activation threshold and the active node x_{47} the seed; **(b)** at instant $t = t_0 + 12$, the procedure verifies step 4, i.e., the stopping condition; the diffusion process ended after twelve iterations.

In this case, the seed was chosen according to the betweenness centrality, where x_{82} was the resulting node. Figure 2.7 (a), illustrates the ER network $G_{(n,p)}$ at the initial instant $t = t_0$, while in Figure 2.7 (b) the maximum influence was obtained in instant $t = t_0 + 10$ and the only remaining inactive node is, again, x_{19}. In this case, we conclude that the influence maximization was accomplished faster than in previous example.



(a) Initial instant $t = t_0$ **(b)** Instant $t = t_0 + 10$

Figura 2.7 **(a)** Erdös-Rényi network $G_{(n,p)}$ with $n = 100$ and $p = 0.05$ at instant $t = t_0$, where $\theta = 0.20$ is the activation threshold and the active node x_{82} the seed; **(b)** at instant $t = t_0 + 10$, the procedure verifies step 4, i.e., the stopping condition; the diffusion process ended after ten iterations.

2.3.2 SIR Model

In Section 2.2.1 the SIR model was presented, as well as, its probabilistic procedure and two examples with different seed sets chosen randomly. Now, we will illustrate another two examples, see (Rocha et al., 2023b), where the seed sets contain the nodes with the highest betweenness centrality, and the highest eigenvalue centrality. In both cases, the seed sets consist of six elements, which correspond to the nodes with the highest values of the specified centrality measure. The networks considered will be ER networks, with $n = 120$ nodes, $p = 0.042$, $\beta = 0.12$ and $\gamma = 0.1$.

In Figure 2.8 (a) is plotted the ER network $G_{(n,p)}$, at instant $t = t_0$ and the seeds are x_3, x_{25}, x_{31}, x_{49}, x_{66} and x_{84}, chosen by the highest betweenness centrality values. At instant $t = t_0 + 29$, given by Figure 2.8 (d), there are 10 susceptible, 14 infected and 96 recovered nodes. In this case, there is 48% of recovered individuals and the same ratio of infected ones.

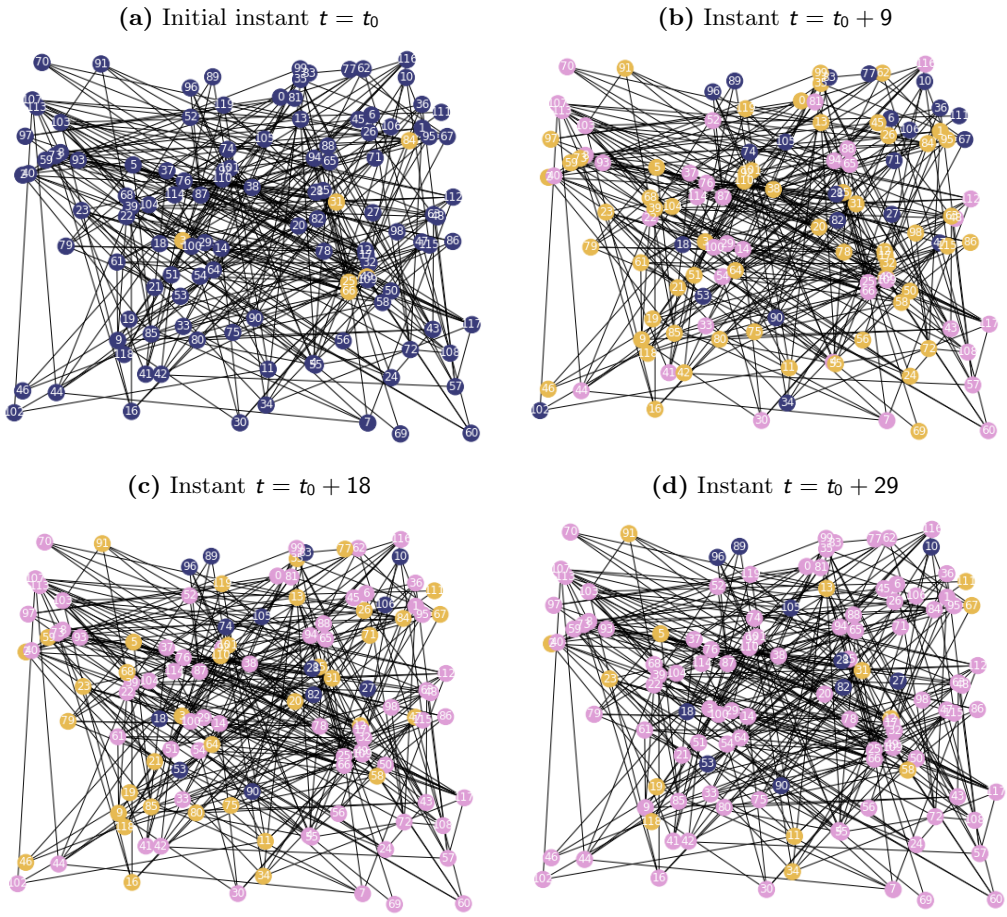


Figura 2.8 Probabilistic procedure for the SIR model over the Erdös-Rényi network $G_{(n,p)}$, with $n = 120$, $p = 0.042$, $\beta = 0.12$, $\gamma = 0.1$ and seeds with the highest betweenness centrality.

In this last case, the seeds are chosen according to the highest eigenvalue centrality. This measure is related with the neighbor of the neighbors of a node.

In Figure 2.9 (a) is plotted the ER network, at instant $t = t_0$ with the seeds x_3, x_{13}, x_{14}, x_{25}, x_{31} and x_{66}. At instant $t = t_0 + 29$, Figure 2.9 (d), there are 8 susceptible, 21 infected and 91 recovered nodes. We observe that 56% of the individuals are not susceptible. Remark that, eigenvalue is one of the measures where propagation process has evolved in a slower pace, see (Rocha et al., 2023b). This statement suggests that, perhaps, in the propagation process, a seed node closer to the others is more important, than a node with a higher number of connections.

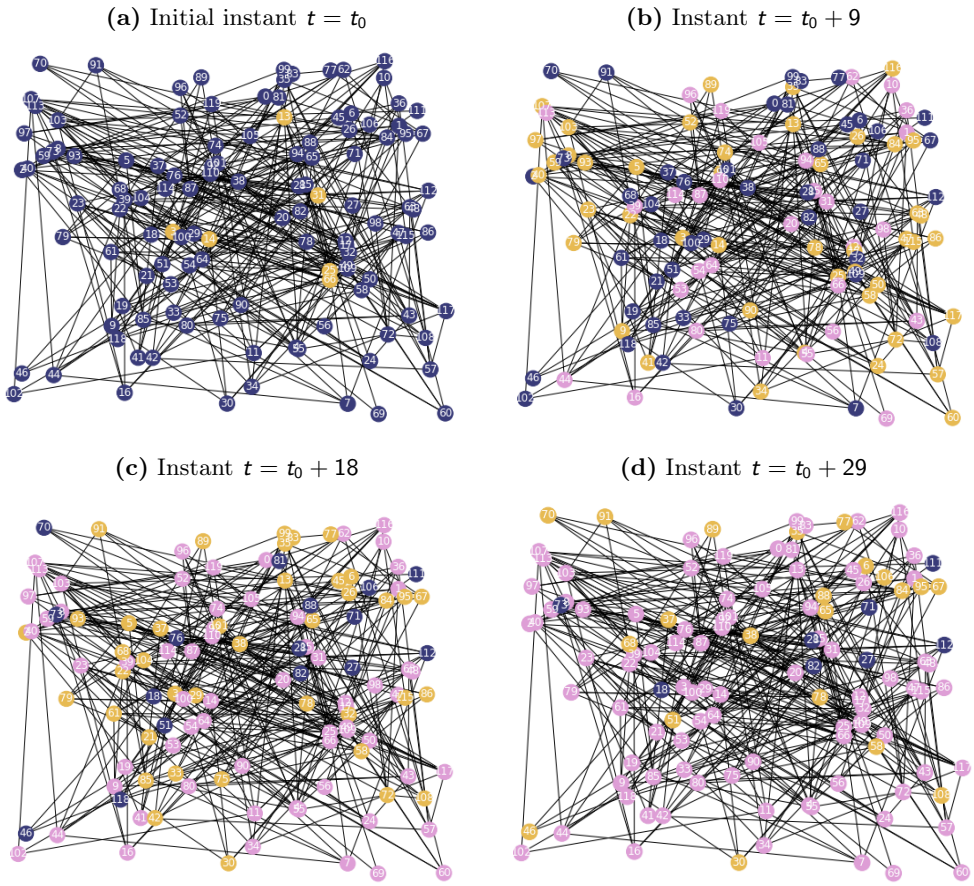


Figura 2.9 Probabilistic procedure for the SIR model over the Erdös-Rényi network $G_{(n,p)}$, with $n = 120$, $p = 0.042$, $\beta = 0.12$, $\gamma = 0.1$ and seeds with the highest eigenvalue centrality.

It should be noted that these measures only consider the position of the nodes on the network and do not take into account other external factors. Consequently, there are other ways of considering such external factors, such as the type of information, whether it is positive or negative, and numerous other features. In order to gain an understanding of how the propagation of various agents occurs, a number of propagation models have been defined and simulated over time. In the next chapter, we will be exploring the Message Affinity Model, which was studied by Argaiz (Argaiz, 2015).

Capítulo 3

Viral Marketing

Viral marketing or viral advertising is a commercial technique that is aimed to promote a product or even an idea, primarily through existing social media networks. The word Viral relates to the way customers share information about a product with others, similar to how a virus spreads from one infected person to the susceptible ones (Reichstein and Brusch, 2019). This topic has recently attracted interest due to its potential to reach a large consumer base with minimal effort and financial resources. The search for the most suitable *influencers* is a primary focus (Kiss and Bichler, 2008; Roelens et al., 2016).

The online WOM, called Viral Marketing, is a somewhat new form of communication, that uses the internet and more specifically social media, in order to promote a product, service, brand, etc.. Viral Marketing is based on the social influence that one user has over others (Eckler and Rodgers, 2010; Jendoubi, 2016). The idea behind this concept, is to select a small set of people, *influencers*, that are able to trigger the process of influence propagation, so that the information reaches a large amount of users that are able to accept the message. Therefore, companies that have a small marketing budget can still reach a great amount of users throughout a social network (Ahmed and Ezeife, 2013).

But the Viral Marketing problem can also be translated into the Influence Maximization Problem, where the main goal is to select a small set of nodes of a social network in a way that makes it able to maximize the global influence in the network.

In the next Sections, we will introduce social networks and their characteristics and a model, studied by Argaiz (Argaiz, 2015), that was created to analyze the influence propagation problem, based on the interest that an individual has on the information that is being spread.

3.1 BA Networks as Social Networks

Networks are a highly useful tool for modeling a population in a variety of scenarios. Given the importance of social interactions as a topic of study, there has been a clear need to employ this tool for the purpose of modeling this event. Consequently, it is imperative to utilize networks that exhibit the defining attributes of a real social network (Bond and Harrigan, 2011). Since real social networks are constantly growing, whether through the addition of new users or new

connections, it makes sense to use networks generated by growing processes (Argaiz, 2015).

The Barabási-Albert (BA) or scale-free network model is a growing model (Barabási and Albert, 1999) that consists of an algorithm that creates randomly scale-free networks. A network is defined as scale-free if the degree distribution follows a power-law distribution (see Appendix B.1). The algorithmic foundation of these networks is distinct from that of other random networks, as it adheres to two fundamental principles: growth and preferential models. In contrast to the ER networks, mentioned in the previous chapter, and the Watts-Strogatz networks, the order of the network is not fixed. Instead, new connections are determined by the degree of the nodes where they will be incident. This implies that the higher the degree of a node, the more likely it is to receive new links or connections. This propensity to establish connections with high-degree nodes results in a highly heterogeneous network, characterized by the presence of hubs. In a scale-free network, hubs are a natural phenomenon whereas in a random ER or Watts-Strogatz network, hubs are exceedingly rare.

The construction of BA networks can be approached in a variety of ways. In this paper, we will present three algorithms used for the creation of these networks: the Barabasi-Albert random networks $G_{(n,n_0,m)}$ (Barabási and Albert, 1999), Dorogovtsev-Mendes networks (Dorogovtsev and Mendes, 2003) and BA with Tunable Clustering (Holme and Kim, 2002).

- **Algorithm 1. BA random networks: model** $G_{(n,n_0,m)}$: Let $G_0 = (V_0, E_0)$ be an ER network of (relatively small) order $n_0 \in \mathbb{N}$, for each instant $t > 0$:

  1. Add a new vertice v_t to V_{t-1}, i.e., $V_t = V_{t-1} \cup \{v_t\}$;
  2. Add $m \leq n_0$ edges to the network, each edge is incident on vertice v_t and on a vertice $u \in V_{t-1}$, chosen with probability

     $$\mathbb{P}_t[u] = \frac{\delta_t(u)}{\sum_{w \in V_{t-1}} \delta_t(w)}$$

     that is, the choice of a vertice u is proportional to its degree at instant t. Vertice u cannot be chosen more than once during time t;
  3. Stop when all n vertices have been added, otherwise repeat the two previous steps.

The resulting BA networks have some distinct properties, such as, the degree distribution follows a power-law distribution (see Appendix B.1) with parameter $\alpha = 3$ and the average path length increases logarithmically with the number of nodes:

$$\overline{d}(G_{(n,n_0,m)}) \cong \frac{ln(n)}{ln(ln(n))}.$$

- **Algorithm 2. Dorogovtsev-Mendes Networks:** Let $G_0 = (V_0, E_0)$ be a ER network of (relatively small) order $n_0 \in \mathbb{N}$ and without edges. For each instant $t > 0$:

  1. Add a new vertice v_t to V_{t-1}, i.e., $V_t = V_{t-1} \cup \{v_t\}$;
  2. Add $m \leq n_0$ edges to the network, each edge is incident on vertice v_t and on a vertice $u \in V_{t-1}$, chosen with probability

  $$\mathbb{P}_t[u] = \frac{\delta_t(u)}{\sum_{w \in V_{t-1}} \delta_t(w)};$$

  3. For some constant $a > 0$ add another $a.m$ edges between the vertices of V_{t-1}, where the probability of adding an edge between vertices u and w is proportional to the product of $\delta(u).\delta(w)$, and with the condition that the edge $\{u, w\}$ has not yet been added;
  4. Stop when all n vertices have been added.

The Dorogovtsev-Mendes networks (Dorogovtsev and Mendes, 2003, 2002) also possess certain distinctive characteristics, such as, when $a = 0$ the degree distribution follows a power-law with parameter $\alpha = 3$ and when $a > 0$, the power-law parameter converges to $\alpha = 2$. In studies applied to real BA networks of this type, it has been found that $2 < \alpha < 3$.

- **Algorithm 3. BA with Tunable Clustering:** Let $G_0 = (V_0, E_0)$ be a ER network of (relatively small) order $n_0 \in \mathbb{N}$ and without edges. Considering each instant $t > 0$:

  1. Add a new vertice v_t to V_{t-1};
  2. Select a vertice $u \in V_{t-1}$ that is not adjacent to v_t and with a probability proportional to its degree $\delta(u)$. Add the edge $\{v_t, u\}$. Add the $m - 1$ remaining edges in the following way:
     - 2.1 If the $m - 1$ edges have already been added, continue with step 3. Otherwise, proceed to the next step;
     - 2.2 With probability q: select a vertice w that is adjacent to u, but not to v_t. If there is no vertice with these characteristics, continue with step 2.3. Otherwise, add edge $\{v_t, w\}$ and continue with step 2.1;
     - 2.3 Select a vertice $r \in V_{t-1}$ that is not adjacent to v_t and with a probability proportional to its degree $\delta(r)$. Add the edge $\{v_t, r\}$ and define $r \to u$. Continue with step 2.1;
  3. Stop when n vertices have been added, otherwise, repeat from step 1.

The BA algorithm with tunable clustering reconciles freedom of scale with the construction of the clustering coefficient of the added vertices, guaranteeing the existence of triangles, which are constructed by adding an edge to a triple at each step of the process. It can be reasonably concluded that the BA networks exhibit the small-world properties observed in social networks, as proven in the works of (Bollobás, 1985; Newman, 2001; Cohen et al., 2002; Dorogovtsev et al., 2002; Chung and Lu, 2002; Bollobás and Riordan, 2002; Cohen and Havlin, 2003).

3.2 The Message Affinity Model

The Message Affinity Model (MAM) emerged with the aim of creating a model that simulates the dissemination of information in a real network, by combining elements from the transition of the SIR epidemic model with the stochastic evolution of a pseudo-Markov Galton-Watson Branching model (see Appendix A.1). MAM has the particularity that it can identify the interest that each node has in the information that is being transmitted. In contrast with the SIR model, it is not consider a global probability for the transition of states for the nodes, but the interest that each person has in passing on the message. Furthermore, it is also considered the knowledge that the node has about the predisposition of the neighborhood for the information that is passing by. The MAM considers the heterogeneity of human behavior, but disregards the impact of social networks on the spread of information (Argaiz, 2015).

In the work of Argaiz (Argaiz, 2015) is presented a study of the propagation of messages in e-mail based networks, with the aim of formalizing a model for the propagation of message information. This study permitted the investigation of each individual's reaction to the same message. The study was conducted in eleven European markets and consisted of acquiring new subscriptions to the newsletter of an information technology company through the recommendation of friends and/or colleagues via email. Those who made recommendations for the newsletter received an incentive. The process was monitored using a form in which the subscriber disclosed which contact they had transmitted the information to.

Cascade networks were obtained as a result. Cascade networks are obtained through the propagation of a message, and whose nodes are the nodes of a social network, to whom the message arrived, and the edges are the ones through which the message was sent. The study of these networks was the basis for the development of the MAM, as they reflect the properties of the propagation model (Argaiz, 2015). It was possible to see that cascades are originated by the diffusion of messages, in which one node transmits the message to another, through *a priori* knowledge of the interests of the possible receiving nodes. That is what makes this model different from the others created to date, whose transmission is related to the density of the network (Argaiz, 2015).

The MAM model postulates that the decision to disseminate information and the number of recipients to which this information is disseminated are correlated. This correlation arises *a posteriori* from the affinity between the disseminator and the content of the message (Argaiz, 2015).

The MAM model simulates all the states in which a person may find themselves with regard to the information propagation process. Consequently, at any given step, the nodes or individuals may find themselves in one of the following states:

- Susceptible (S): the node did not receive the message;
- Informed (I): the node is propagating the message;
- Refractory (R): The node does not spread the information anymore.

The state of the nodes results from the interaction between the willingness of the node to pass the message and the "suitability" of the message to be propagated. The affinity of a node $a_n \in [0, 1]$ represents its inclination to pass the message, whose ability to trigger interest, or activate another node, is represented by the affinity threshold $A_T \in [0, 1]$, which defines the lowest value for the message to impose the Informed state. Messages with low thresholds are able to activate more nodes and are, hence, forwarded more frequently, than messages with high threshold values.

The propagation in the network starts, with the seed nodes, which initially are in the Informed state, while the remaining nodes are in the Susceptible state. From that, the process follows the next steps:

1. Susceptible nodes that receive the message become Informed, if their affinity value is greater than the affinity threshold value ($a_n > A_T$), and become Refractory otherwise. If Informed or Refractory nodes receive the message they stay in the same state.

2. An Informed node n sends $(r_1)_n = (a_n - A_T) \times r_1$ messages, with r_1 drawn from a power-law distribution $P(r_1)$ (see Appendix B.1). The neighbors that receive the message are:

   (a) those with highest affinity value a_n with probability $(a_n - A_T)$;

   (b) chosen randomly with probability $1 - (a_n - A_T)$.

3. Informed nodes became Refractory immediately after they send the message and the process stops when there are any Informed nodes.

The quantity $(a_n - A_T)$ plays several roles in the message propagation. The choice that each node does in step 2 is based on its affinity value: if a node has an affinity value lower than the affinity threshold value ($a_n < A_T$) it implies that the node is not aware of its neighbors interest in the message and, therefore, the message is passed to randomly chosen neighbors. Otherwise, the node is considered to have local knowledge, and sends the message to nodes with higher affinity values. The level of knowledge that a node has regarding the possible interest of its neighbors in the message, is measured by the affinity threshold A_T, whose maximum value is $A_T = 0$ and minimum is $A_T = 1$. The A_T value may vary throughout the process, however, it is assumed to be constant.

The propagation of a message through the MAM is described in Algorithm 1. MAM depends on the local dynamics of the network: it is necessary to have knowledge of the node's neighbors, and the affinity value of the node and its neighbors (Argaiz, 2015).

Algorithm 1 Viral propagation according to MAM

while there are Informed nodes **do**

1. Determine which nodes are in the Informed state
2. Select a node randomly to continue the propagation
3. Select the neighbors to share the message with

if number of messages > number of neighbors **then**

Send to all neighbors

else if number of messages < number of neighbors **then**

if $a_n > A_T$ **then**

Select neighboring nodes with the highest value of a_{n+1} with probability $(a_n - A_T)$

else if $a_n < A_T$ **then**

Select neighbors randomly with probability $1 - (a_n - A_T)$

end if

end if

4. From the selected nodes, only those with $a_{n+1} > A_T$ become Informed

if $a_{n+1} > A_T$ **then**

The node becomes Informed

else if $a_{n+1} < A_T$ **then**

The node becomes Refractory

end if

end while

It is important to mention that even for values of $A_T = 1$, active nodes are forced to send at least one message, in order to prevent active nodes from becoming inactive, if $(a_n - A_T) \times r_1 < 1$.

3.2.1 Affinity

The random affinity value a_n, could be taken from a uniform distribution $U \in [0, 1]$, however, it may be necessary in several situations to have a non-uniform distribution with a probability density function $p_a(x)$. Given a probability density function $p_a(x)$ with cumulative distribution function $F(a) = \int_0^a p_a(x)dx$ and F^{inv} a random variate, i.e. a particular outcome of a random variable, distributed as p_a results from (Devroye, 1986):

$$A = F^{inv}(U), \tag{3.1}$$

where U is a uniform random variable $U \in [0, 1]$. The results of the Inversion Method, applied to Equation (3.1), for the truncated uniform distribution and for the exponential distribution are going to be addressed. The use of different distributions has little impact on the propagation process (Argaiz, 2015). For simplicity purposes, only the truncated uniform distribution will be used.

For the truncated uniform distribution, a fraction $q \in [0, 1]$ of the nodes has affinity $a_n = 0$ and the remainder has a uniform distribution. The probability function has a delta function at

$x = 0$ and is constant elsewhere. Since q is a value taken from the uniform probability function, we have

$$p_a(x) = \begin{cases} \delta(x)q, & x = 0 \\ (1-q), & 0 < x \leq 1 \end{cases} . \tag{3.2}$$

The probability function is normalized when $x \in [0,1]$ $\left(\int_0^1 p_a(x)dx = q + (1-q) = 1\right)$ and has a cumulative distribution function defined by

$$F(x) = \int_0^a p_a(x)dx = \begin{cases} q, & a = 0 \\ q + (1-q)a, & 0 < a \leq 1 \end{cases} , \tag{3.3}$$

which translates into an affinity value $a_n = 0$ for a fraction of q nodes and an affinity value $a_n = U \in [0,1]$ for the remaining $1-q$ nodes. The mean of the affinity values in the network is given by

$$\langle A_{unif} \rangle = \int_0^1 x p_a(x)dx = \int_{0^+}^1 (1-q)xdx = \frac{1-q}{2}. \tag{3.4}$$

For an exponential distribution, the probability density function is $p_a(x) = qe^{-qx}/(1-e^{-q})$, with $q > 0$, normalized for $x \in [0,1]$. The cumulative distribution function is given by $F(x) = (1-e^{-qx})/(1-e^{-q})$. By applying the Inversion Method we get the exponentially distributed random variate:

$$A_{exp} = -\frac{1}{q} ln[1 - U(1-e^{-q})]. \tag{3.5}$$

To calculate the affinity value of the nodes, it was taken into account that a pair of random variables A and U are related as follows $A = g(U) = g[f(u)]$ (Stirzaker, 1999), the expected value of A is given by

$$E[A] = \int_{\mathbb{R}} g(u)f(u)du, \tag{3.6}$$

from where we obtain the mean of the affinity values using the exponential distribution:

$$\langle A_{exp} \rangle = -\frac{1}{q}\int_0^1 ln[1 - U(1-e^{-q})]du = \frac{1-(1+q)e^{-q}}{q(1-e^{-q})}. \tag{3.7}$$

The allocation of values on the fraction q follows the next pattern: the higher the value of q, the fewer the number of nodes that are likely to share the message. The process of allocating affinity values, is described in Algorithm 2.

Algorithm 2 Allocating Affinity Values

Input: Let q be a fraction of nodes:

1. Obtain probability values from a uniform distribution function
2. Randomly choose q nodes of the network
3. Give $a_n = 0$ to the q nodes and to the remaining $1-q$ give the values obtained in 1.

3.2.2 Forwarded Messages

In social online settings, most people follow an average behavior, but a significant portion of the population shows bursts of activity, like the number of e-mail sent per day (Ebel et al., 2002), telephone calls placed by the user (Aiello et al., 2000), blog posts by the user, web page clicks per user (Pitkow, 1997; Gruhl et al., 2004) or the time spent between receiving and replying to an e-mail (Barabási, 2005). Therefore, the number of messages sent by an individual can be explained by a power law (Argaiz, 2015).

Based on empirical results, the number of messages $r \geq 1$ is taken from a power law distribution $r \sim PL(\alpha, \beta)$ with probability density function, given by:

$$P_{PL}(r) = \frac{H_{\alpha,\beta}}{\beta + r^{\alpha}}, \quad r = 1, 2, 3, ... \tag{3.8}$$

which asymptotically decreases like a Pareto distribution (Johnson et al., 1992) (see Appendix B.2) (for large values of r, the function behaves like $P_{PL} = r^{-\alpha}$) and has a limit for small numbers of recommendations ($r^* \simeq \beta^{1/\alpha}$). This value represents the point at which $P_{PL}(r^*) \sim \frac{1}{2}P_{PL}(1)$ and marks the beginning of the behavior of the tail's distribution.

The parameters are constant and α is a parameter of the power law distribution (see Appendix B.1). The value of the normalization constant $H_{\alpha,\beta}$ is estimated such that $\sum_{r=1}^{\infty} P_{PL}(r) = 1$, and this process is represented in Algorithm 3.

Algorithm 3 Calculation of the normalization constant $H_{\alpha,\beta}$

Input: Let k_{max} be the maximum degree of the chosen network and β and α two constants

1. Calculate $\sum_{r=1}^{k_{max}} \frac{1}{r^{\alpha}}$
2. Calculate $H_{\alpha,\beta}$, such that $\sum_{r=1}^{k_{max}} \frac{1}{r^{\alpha}} \times H_{\alpha,\beta} = 1$

The calculation of $H_{\alpha,\beta}$ is made every time that k_{max} changes. The determination of the number of messages is given by Equation (3.8), for the normalization constant previously obtained.

Algorithm 4 Calculation of the number of messages r_1

Input: Let k_{max} be the maximum degree of the chosen network, r the possible degree values of the nodes and β and α two constants

1. Calculate $P_{PL}(r) = \frac{H_{\alpha,\beta}}{\beta + r^{\alpha}}, \quad r = 1, 2, ..., k_{max}$
2. From a uniform distribution, get a probability $U \in [0, 1]$
3. Get r, for the probability value obtained in 2.

There is an imposition on the maximum number of messages a node can send, where any node in the network does not exceed the maximum degree of the network. In fact, the maximum number of messages a node can send depends on the number of neighbors it has, this cannot exceed $k_n - 1$, where k_n is the degree of the node. This condition imposes a limit

on the tail of the network distribution, after the propagation, and is a result of the second rule of the model. This limit, which is regulated by the network distribution, is more important for seed nodes than for the remaining nodes, since the latter are able to send more messages, due to the non-randomness in their choice. In fact, since the network is heterogeneous, $\frac{\langle k \rangle}{\langle k \rangle^2} \gg 1$ it is expected that the number of messages sent by seed nodes will be smaller than the number sent by the remaining nodes, $\langle r_v \rangle > \langle r_s \rangle$. To guarantee the heterogeneity of the model, two values of β will be imposed, for the seed nodes will be β_s and for the rest it will be β_v, where, $\beta_s < \beta_v$, therefore there will be two probability density functions.

3.3 Case Studies

In this Section we will present some case studies on the information propagation process according to the MAM model. The different examples will show how the information spreads throughout the networks depending on the seed set chosen, which will be performed according to the topological centrality measures shown in Section 2.3. In all the cases, the social networks used will be BA random networks, model $G_{(n,n_0,m)}$ (Barabási and Albert, 1999), with $n = 150$ nodes and $m = 2$, representing the number of edges added in each iteration to form the network. A descriptive study will be conducted to examine the characteristics of all social networks that have been created.

Regarding the MAM model parameters, the examples will have $q = 0.2$, affinity threshold $A_T = 0.4$, $\alpha = 1.9$, $\beta_s = 10$ and $\beta_v = 40$. This last three values were given according to the empirical results in the work of (Argaiz, 2015). The implementation of these case studies can be found in Appendix D.

3.3.1 Case Study 1

For the first case, we have a BA network with $n = 150$ individuals, as shown in Figure 3.1. This network was obtained with the BA random networks algorithm, where we started the process with $n_0 = 1$, that is, a single vertex and no edges in the first time step. Then the algorithm proceeds to execute the requisite steps in order to attain the network with 150 nodes. Once the final network configuration has been determined, the MAM model may be applied.

Table 3.1, shows some characteristics of the network, like its maximum degree, minimum degree, diameter, average path length and transitivity.

Max degree	Min degree	Diameter	Average path length	Transitivity
17	2	6	3.43	0.042

Tabela 3.1 Descriptive characteristics of the Barabási-Albert random network $G_{(n,n_0,m)}$ with $n = 150$ nodes, $n_0 = 1$ and $m = 2$.

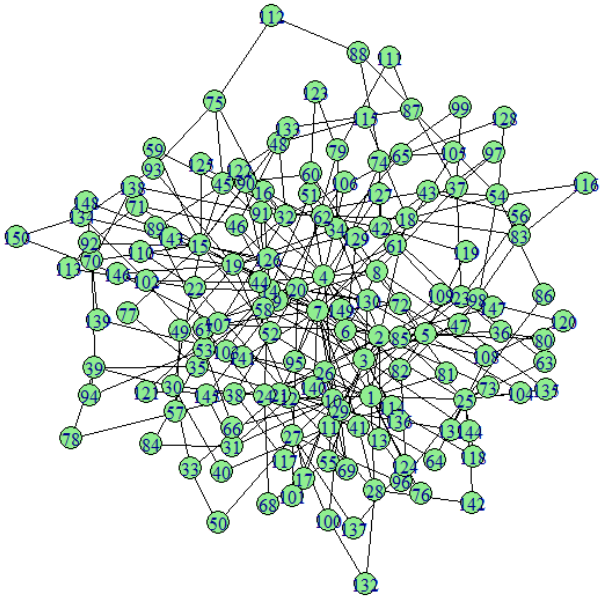


Figura 3.1 Barabási-Albert random network $G_{(n,n_0,m)}$ with $n = 150$ nodes, $n_0 = 1$ and $m = 2$.

According to Table 3.1 this BA network has diameter $diam(G) = 6$. Recall that, the diameter of a network is the highest eccentricity, that is,

$$diam(G) = max\{\epsilon(v) : v \in V\}.$$

Where the eccentricity of a node u, $\epsilon(u)$, is the maximum distance of u from any $v \in V$. This is represented by,

$$\epsilon(u) = max\{d(u, v) : v \in V\}.$$

It can be concluded that a path with length less than 6 edges is sufficient to connect any two given nodes. Moreover according to the average path length it usually takes 3.43 edges to get to one node to another.

The transitivity is a measure to see to what "extent" the neighbors of a node v are adjacent to each other, but in a global perspective. This measure consists of analyzing the number of triples and triangles (cycles of order 3) in a network, and is given by

$$trans(G) = \frac{n_\Delta(G)}{n_\wedge(G)}$$

where $n_\Delta(G)$ is the number of triangles and $n_\wedge(G)$ the number of triples of the network.

Figure 3.2 (a) shows the histogram of the BA network in question. We can see that the majority of the nodes have degree lower than five and there is a small number of nodes with high degree. Figure 3.2 (b) shows the proportion of nodes with a given degree. For example, 47% of the nodes have two neighbors, 16% three and 15% four, that is, 78% of the nodes have degree lower that five.

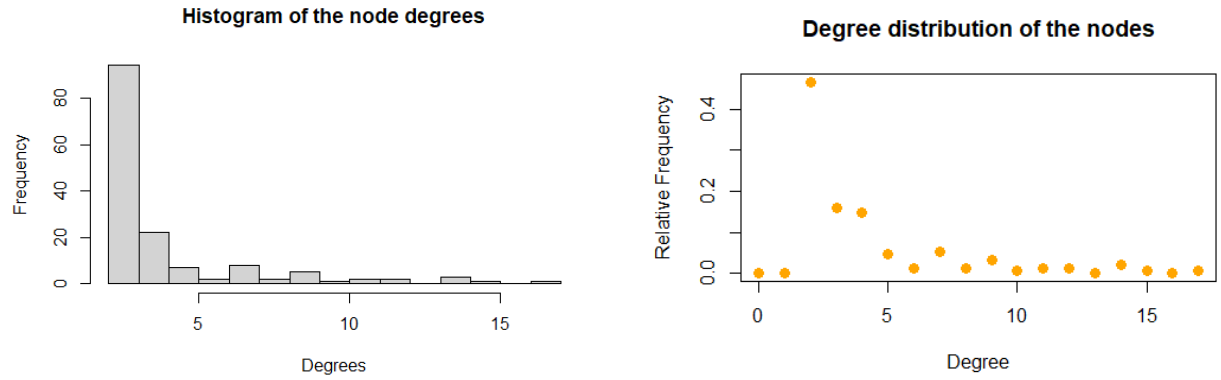


Figura 3.2 **(a)** Histogram of the node degrees of the Barabási-Albert random network $G_{(n,n_0,m)}$ with $n = 150$ nodes, $n_0 = 1$ and $m = 2$; **(b)** Degree distribution of the nodes of the Barabási-Albert random network $G_{(n,n_0,m)}$ with $n = 150$ nodes, $n_0 = 1$ and $m = 2$.

In Figure 3.3 are represented the hubs of this BA network. In network theory, a hub is a node with a high degree of connectivity, that is, a node with a large number of connections. In Figure 3.3, the size of the node is proportional to its degree.

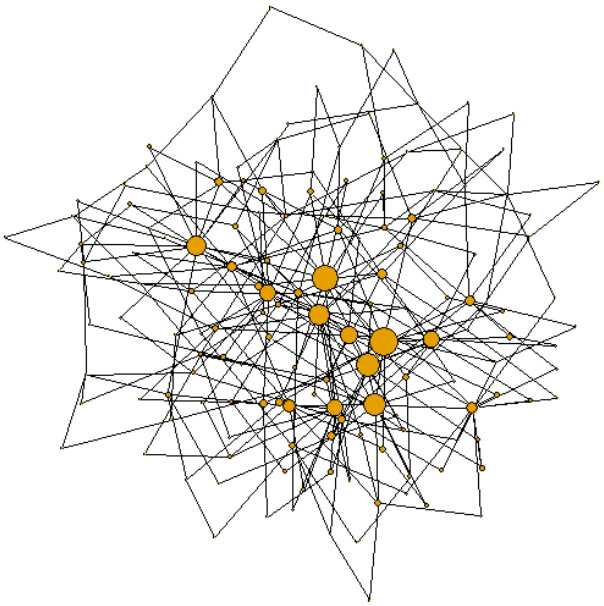


Figura 3.3 Hubs of the Barabási-Albert random network $G_{(n,n_0,m)}$ with $n = 150$ nodes, $n_0 = 1$ and $m = 2$.

To initiate the propagation process, we select the seed nodes, i.e., the nodes initially informed and that will propagate the message throughout the network. In this case we will use the centrality measures mentioned in Section 2.3, where Table 3.2 shows the nodes with the highest values of the centrality measures, i.e., degree, closeness, betweenness and eigenvector. It is crucial to highlight that, in the context of this case study, the central nodes depicted in Table 3.2 can also be classified as hubs.

Degree centrality	Closeness centrality	Betweenness centrality	Eigenvector centrality
15	7	15	2

Tabela 3.2 Nodes with the highest centrality values for the Barabási-Albert random network $G_{(n,n_0,m)}$ with $n = 150$ nodes, $n_0 = 1$ and $m = 2$.

The requisite elements have now been assembled to initiate the simulation of the information propagation process with the MAM model.

Figure 3.4 illustrates the propagation process when the seed set is the node x_{15} (red color), which is the node with the highest degree and betweenness centrality. The nodes in red represent the individuals who have received the message and are propagating it. The nodes in orange correspond to those who have just received the message. The nodes in light green are those that are susceptible to receiving the message. The process ceased after 33 iterations, with 66 nodes informed of the content of the message. However, the remaining susceptible nodes were not sufficiently interested in receiving it.

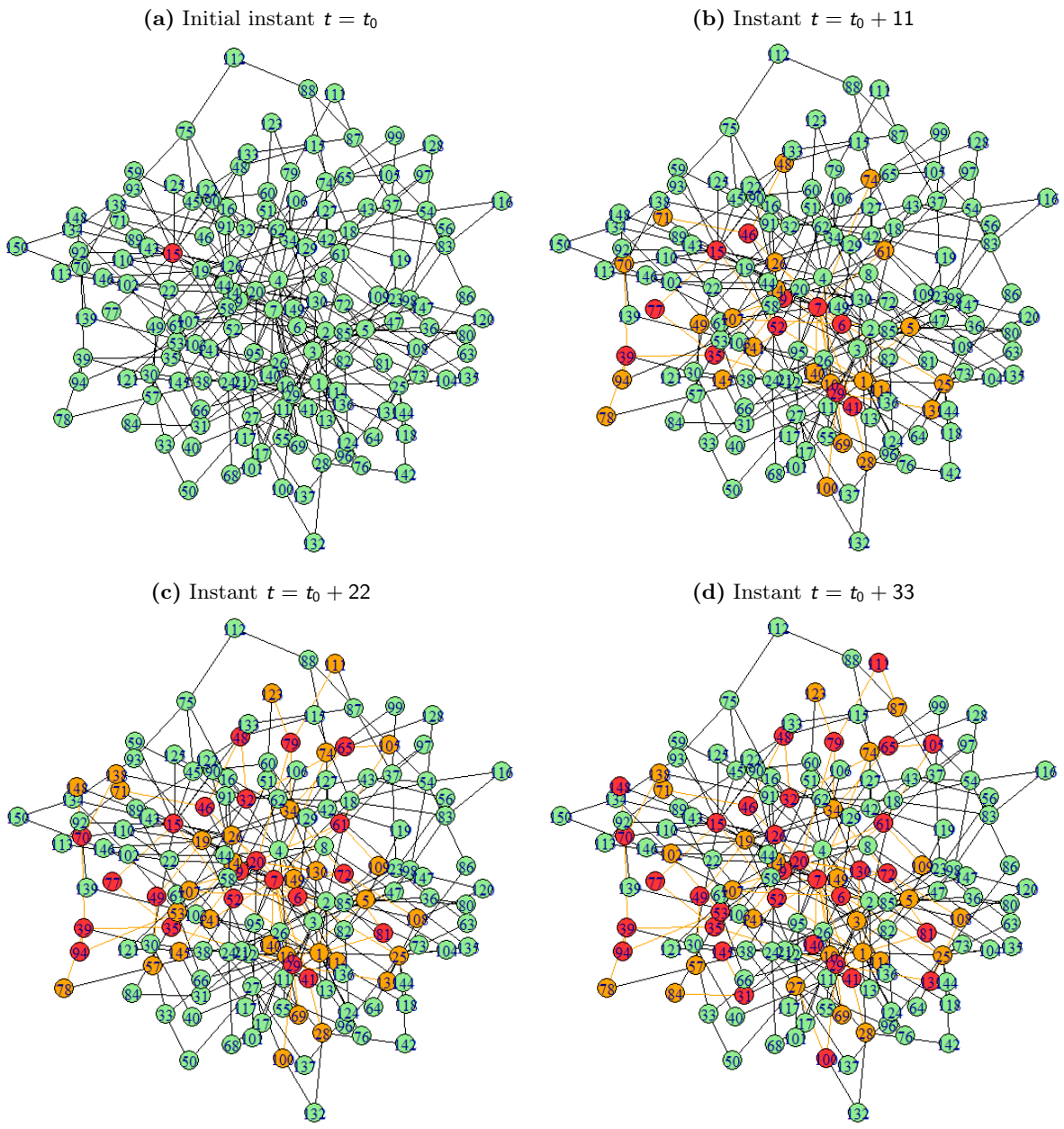
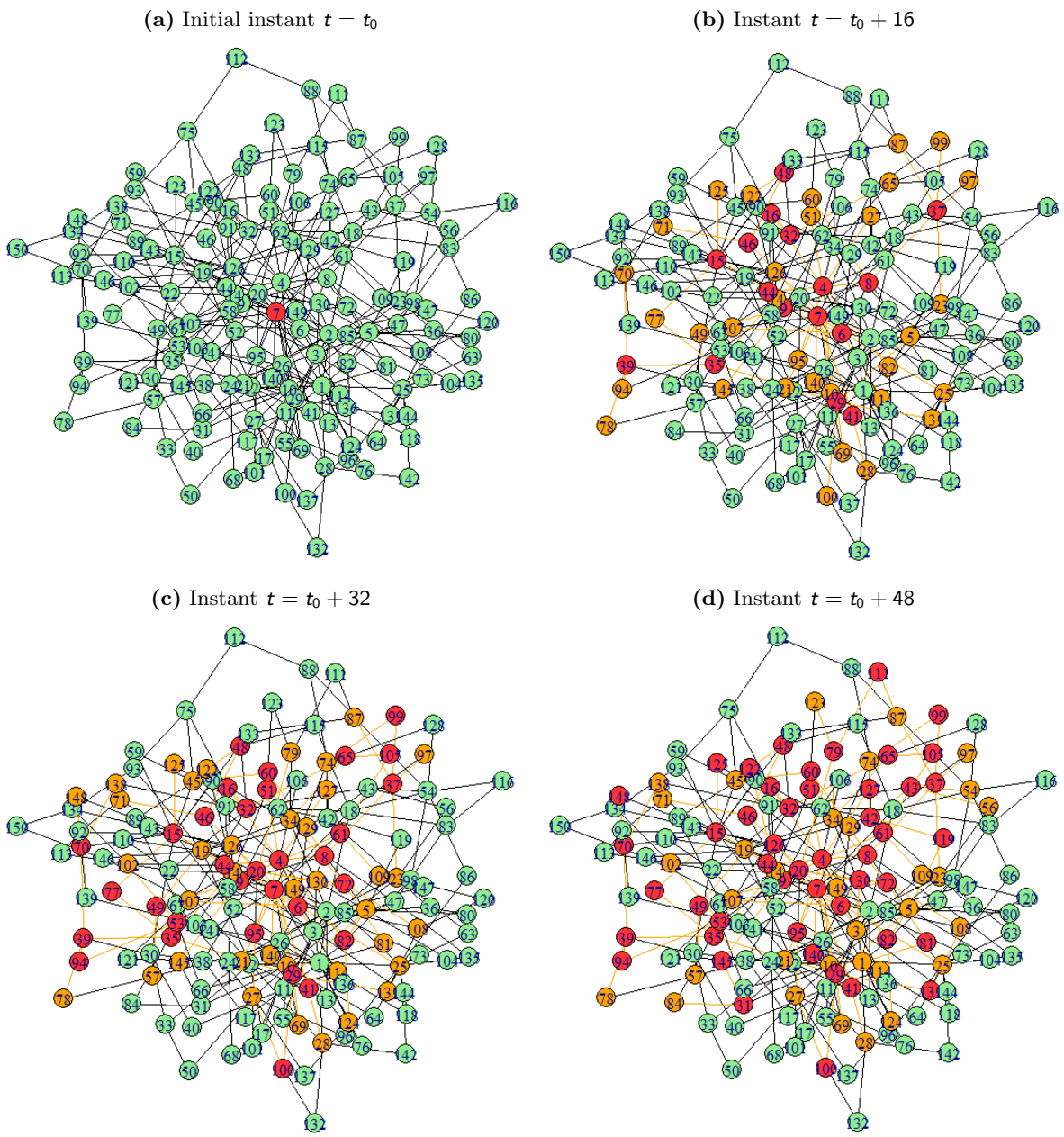


Figura 3.4 Information propagation for the MAM model over the Barabási-Albert random network $G_{(n,n_0,m)}$, with $n = 150$, $n_0 = 1$, $m = 2$, $q = 0.2$, $A_T = 0.4$, $\alpha = 1.9$, $\beta_s = 10$, $\beta_v = 40$ and seed with the highest degree and betweenness centrality.

In Figure 3.5, we initiate the process with node x_7, which has the highest closeness centrality. In this example, the information diffusion process terminated after 48 iterations, resulting in a total of 92 informed nodes. This represents approximately 61% of the social network's nodes that received the message. A comparison with the previous example reveals that the number of informed nodes is approximately 1.4 times greater.



(a) Initial instant $t = t_0$ **(b)** Instant $t = t_0 + 16$

(c) Instant $t = t_0 + 32$ **(d)** Instant $t = t_0 + 48$

Figura 3.5 Information propagation for the MAM model over the Barabási-Albert random network $G_{(n,n_0,m)}$, with $n = 150$, $n_0 = 1$, $m = 2$, $q = 0.2$, $A_T = 0.4$, $\alpha = 1.9$, $\beta_s = 10$, $\beta_v = 40$ and seed with the highest closeness centrality.

Figure 3.6 illustrates the scenario where the information propagation began with the node exhibiting the highest value of eigenvector centrality, that is, node x_2. It can be observed that this example represents a case where the information did not spread throughout the network. This is evidenced by the fact that the process ceased after one iteration, with a total of two nodes receiving the message.

In conclusion, it can be posited that the most influential node on this social network is node x_7, which is defined as the node with the highest closeness centrality.

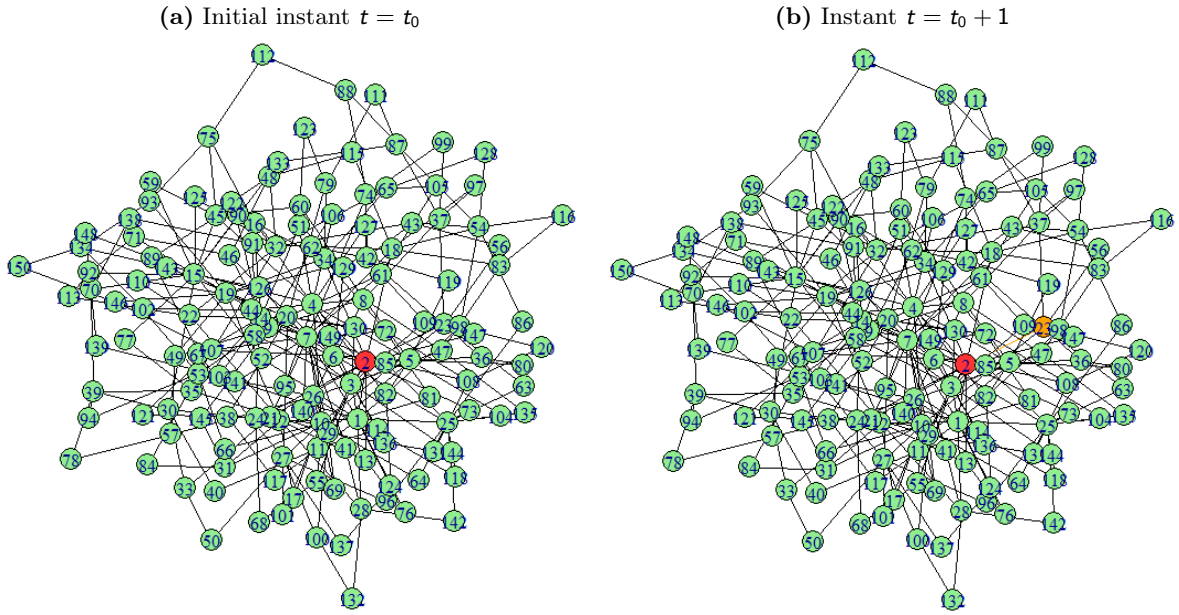


Figura 3.6 Information propagation for the MAM model over the Barabási-Albert random network $G_{(n,n_0,m)}$, with $n = 150$, $n_0 = 1$, $m = 2$, $q = 0.2$, $A_T = 0.4$, $\alpha = 1.9$, $\beta_s = 10$, $\beta_v = 40$ and seed with the highest eigenvector centrality.

3.3.2 Case Study 2

We have once again constructed a BA network with $n = 150$ nodes and $m = 2$, as illustrated in Figure 3.7. A descriptive summary of the network's characteristics is presented in Table 3.3.

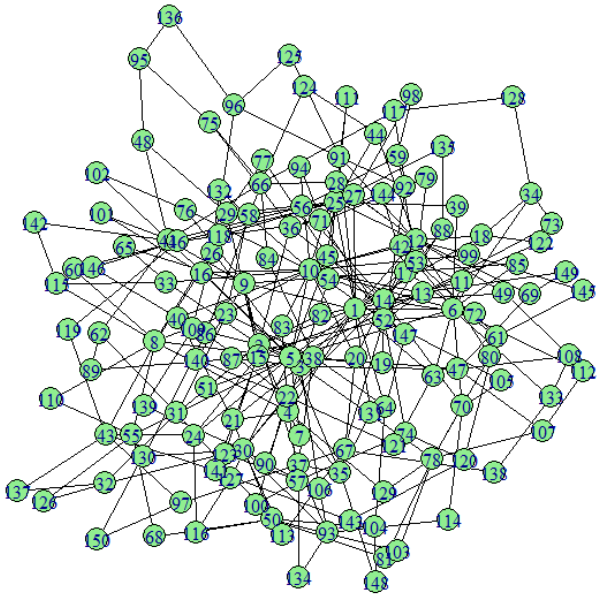


Figura 3.7 Barabási-Albert random network $G_{(n,n_0,m)}$ with $n = 150$ nodes, $n_0 = 1$ and $m = 2$.

Max degree	Min degree	Diameter	Average path length	Transitivity
18	2	7	3.47	0.049

Tabela 3.3 Descriptive characteristics of the Barabási-Albert random network $G_{(n,n_0,m)}$ with $n = 150$ nodes, $n_0 = 1$ and $m = 2$.

According to Table 3.3 this BA network has diameter $diam(G) = 7$. It can therefore be stated that a path of no more than seven edges is sufficient to traverse between any two given nodes. The average path length indicates that it typically takes 3.47 edges to connect one node with another.

Figure 3.8 (a) depicts the histogram of the BA network in question. It is evident that the majority of nodes possess a degree less than four, with a minimal number of nodes exhibiting high degree values. Figure 3.8 (b) illustrates the proportion of nodes with a given degree. For instance, approximately 43% of the nodes have two neighbors, while 23% have three. This indicates that more than 66% of the nodes have a degree of less than four.

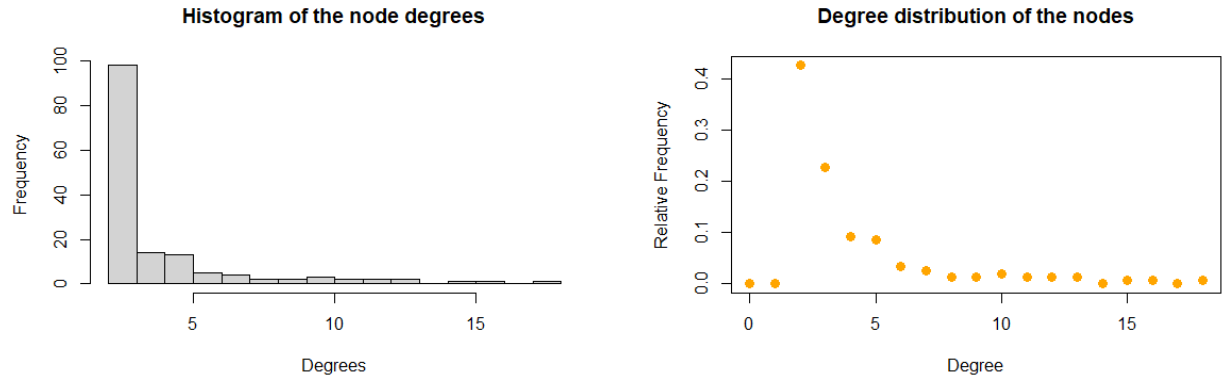


Figura 3.8 **(a)** Histogram of the node degrees of the Barabási-Albert random network $G_{(n,n_0,m)}$ with $n = 150$ nodes, $n_0 = 1$ and $m = 2$; **(b)** Degree distribution of the nodes of the Barabási-Albert random network $G_{(n,n_0,m)}$ with $n = 150$ nodes, $n_0 = 1$ and $m = 2$.

Figure 3.9 depicts the hubs of the BA network. For this network, it can be observed that there are approximately nine hubs.

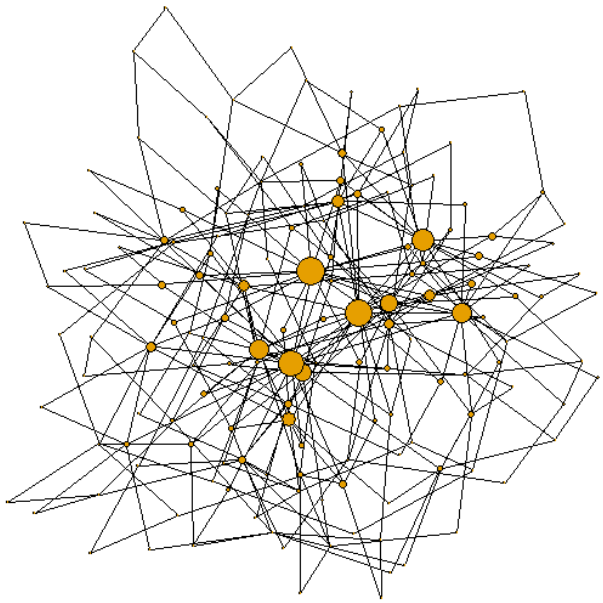


Figura 3.9 Hubs of the Barabási-Albert random network $G_{(n,n_0,m)}$ with $n = 150$ nodes, $n_0 = 1$ and $m = 2$.

Regarding the seed nodes, Table 3.4 presents the most central nodes for each of the measures utilized. For this network, it can be observed that the node x_{10} exhibits the highest values of both degree and eigenvector centrality. Of the nine identified hubs, three are nodes of Table 3.4.

Degree centrality	Closeness centrality	Betweenness centrality	Eigenvector centrality
10	1	5	10

Tabela 3.4 Nodes with the highest centrality values for the Barabási-Albert random network $G_{(n,n_0,m)}$ with $n = 150$ nodes, $n_0 = 1$ and $m = 2$.

Figure 3.10 illustrates the propagation process when the seed set is equal to the node x_{10}, which has the highest value for degree and eigenvector centrality, as previously mentioned. The process ceased after one iteration, with only two nodes receiving the message. In this instance, it can be observed that this node is not a influencer.

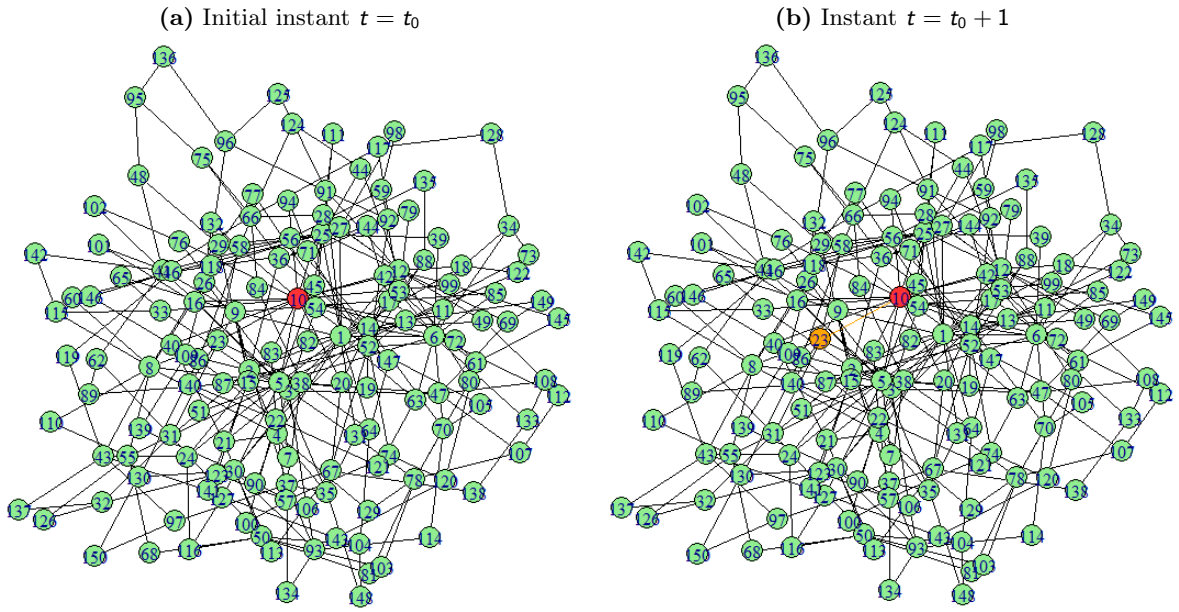


Figura 3.10 Information propagation for the MAM model over the Barabási-Albert random network $G_{(n,n_0,m)}$, with $n = 150$, $n_0 = 1$, $m = 2$, $q = 0.2$, $A_T = 0.4$, $\alpha = 1.9$, $\beta_s = 10$, $\beta_v = 40$ and seed with the highest degree and eigenvector centrality.

In Figure 3.11, we commence the process with node x_1, which has the highest closeness centrality. In this example, the information diffusion process did not occur, as evidenced by the fact that only one iteration was completed, resulting in a total of two informed nodes.

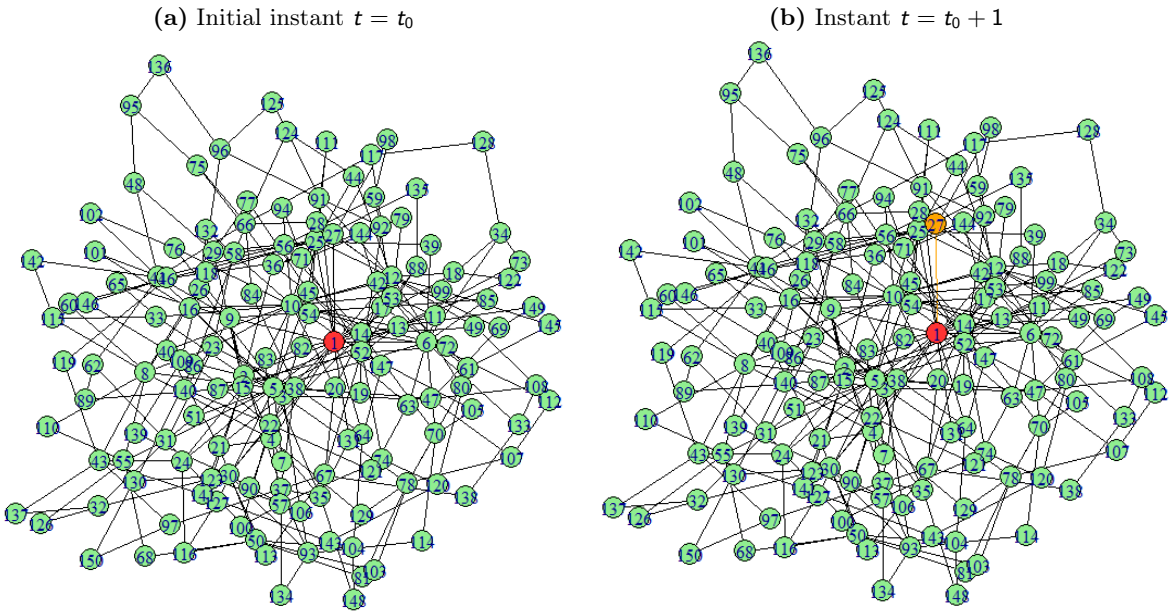
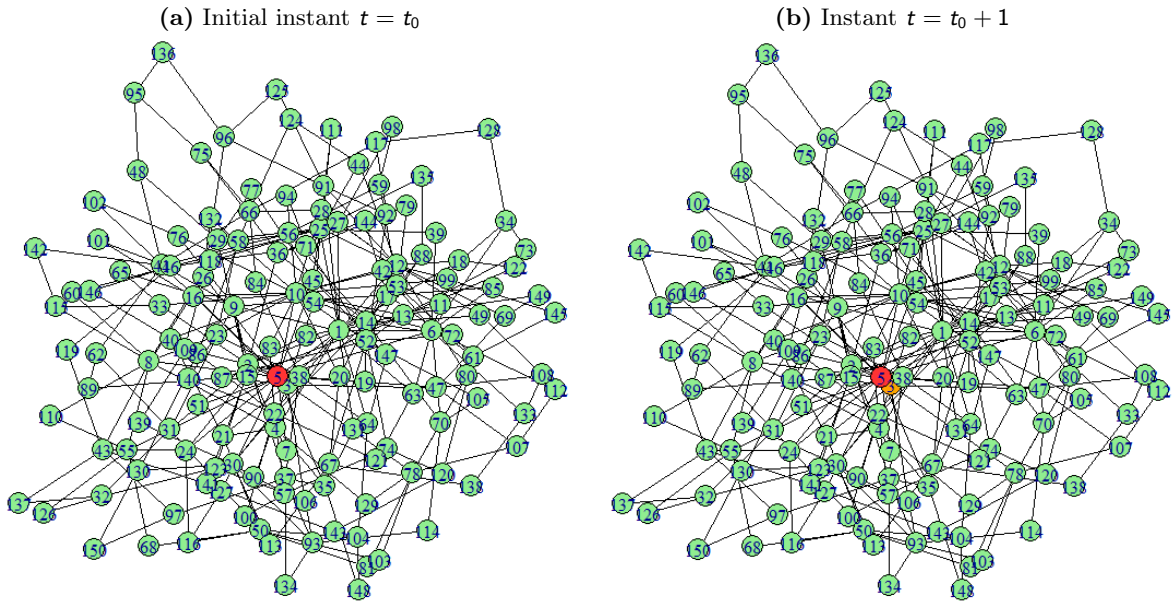


Figura 3.11 Information propagation for the MAM model over the Barabási-Albert random network $G_{(n,n_0,m)}$, with $n = 150$, $n_0 = 1$, $m = 2$, $q = 0.2$, $A_T = 0.4$, $\alpha = 1.9$, $\beta_s = 10$, $\beta_v = 40$ and seed with the highest closeness centrality.

Figure 3.12 illustrates the scenario where the information propagation began with the node exhibiting the highest value of betweenness centrality, that is, node x_5. It can be observed, once again, that the information did not spread throughout the network.



(a) Initial instant $t = t_0$ **(b)** Instant $t = t_0 + 1$

Figura 3.12 Information propagation for the MAM model over the Barabási-Albert random network $G_{(n,n_0,m)}$, with $n = 150$, $n_0 = 1$, $m = 2$, $q = 0.2$, $A_T = 0.4$, $\alpha = 1.9$, $\beta_s = 10$, $\beta_v = 40$ and seed with the highest betweenness centrality.

We are able to conclude, that using central nodes is not a sufficient condition to attain a large information diffusion within the network. Consequently, these nodes are unsuitable as influencers. To see if we have success in this process, lets consider as seed nodes, some of the hubs of the network.

Figure 3.13 illustrates the propagation process when the seed set is the node x_{15}, which is a hub of the network. In this instance, the process ceased after 51 iterations, with 100 informed nodes. This represents approximately 67% of the social network's nodes that received the message.

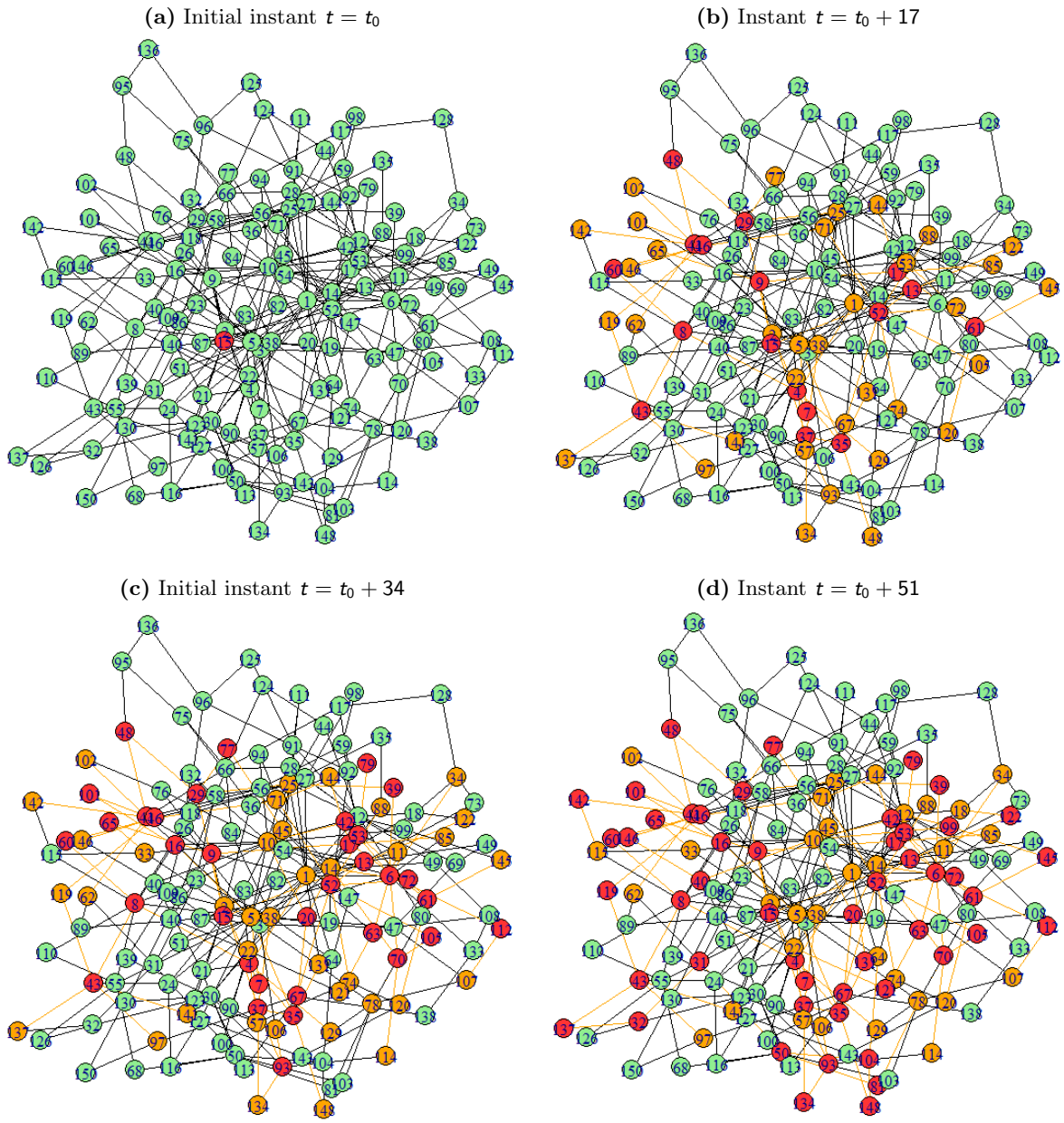


Figura 3.13 Information propagation for the MAM model over the Barabási-Albert random network $G_{(n,n_0,m)}$, with $n = 150$, $n_0 = 1$, $m = 2$, $q = 0.2$, $A_T = 0.4$, $\alpha = 1.9$, $\beta_s = 10$, $\beta_v = 40$ and seed node x_{15} a hub of the network.

As illustrated in Figure 3.14, the propagation process commences with the node x_6, which is another hub of the network. For this case, the process also ceased after 51 iterations, with 99 informed nodes. This represents approximately 66% of the social network's nodes that received the message.

In conclusion, it can be observed that the hubs are a more suitable fit to be considered influencers when compared to the nodes obtained by the centrality measures.

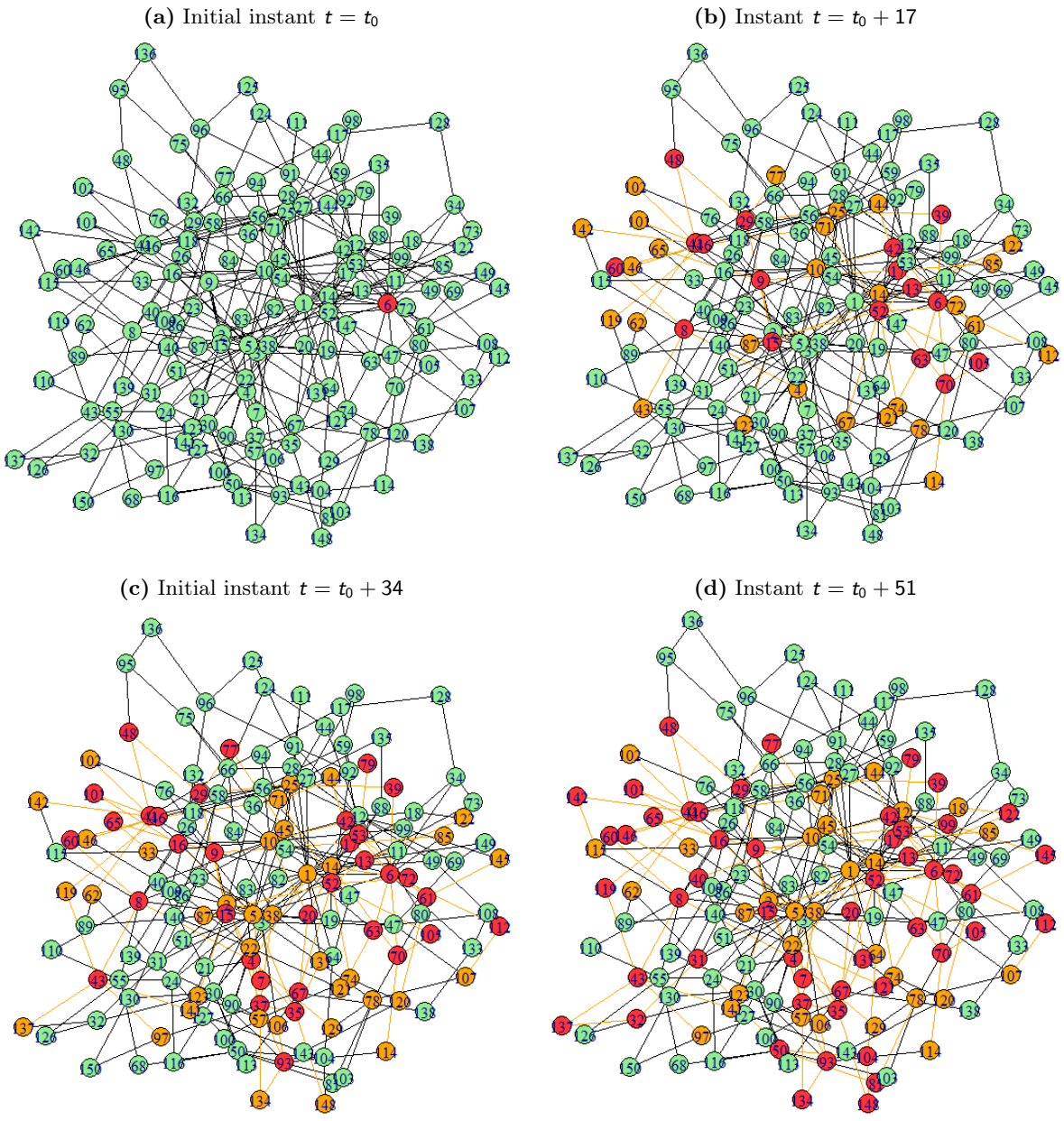


Figura 3.14 Information propagation for the MAM model over the Barabási-Albert random network $G_{(n,n_0,m)}$, with $n = 150$, $n_0 = 1$, $m = 2$, $q = 0.2$, $A_T = 0.4$, $\alpha = 1.9$, $\beta_s = 10$, $\beta_v = 40$ and seed node x_6 a hub of the network.

3.3.3 Case Study 3

In the final case study, the social network will also be a BA network with $n = 150$ and $m = 2$, as illustrated in Figure 3.15. Table 3.5 presents a descriptive summary of the network attributes.

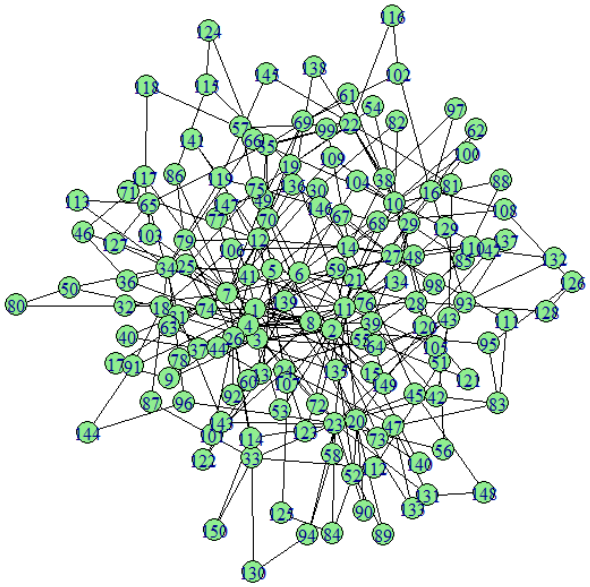


Figura 3.15 Barabási-Albert random network $G_{(n,n_0,m)}$ with $n = 150$ nodes, $n_0 = 1$ and $m = 2$.

Max degree	Min degree	Diameter	Average path length	Transitivity
19	2	6	3.31	0.042

Tabela 3.5 Descriptive characteristics of the Barabási-Albert random network $G_{(n,n_0,m)}$ with $n = 150$ nodes, $n_0 = 1$ and $m = 2$.

According to Table 3.5 this BA network has diameter $diam(G) = 6$, that means, that it takes a path of no more than six edges to get to one node to another. The average path length gives us the information that it typically takes 3.31 edges to connect one node with another.

Figure 3.16 (a) illustrates the histogram of the BA network. It is clear that the majority of nodes possess a degree less than four, with a minimal number of nodes exhibiting high degree, similar to the previous case. Figure 3.16 (b) shows that approximately 45% of the nodes have two neighbors, while 25% have three. This indicates that more than 70% of the nodes have a degree of less than four.

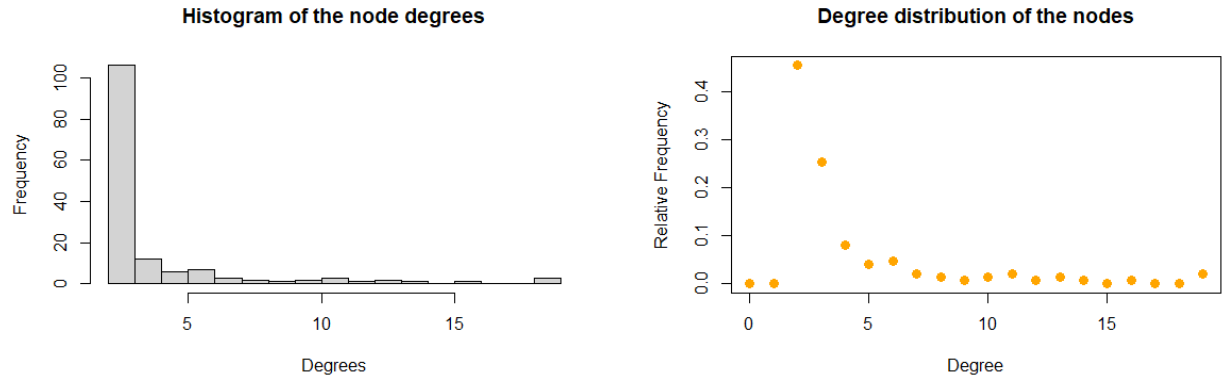


Figura 3.16 **(a)** Histogram of the node degrees of the Barabási-Albert random network $G_{(n,n_0,m)}$ with $n = 150$ nodes, $n_0 = 1$ and $m = 2$; **(b)** Degree distribution of the nodes of the Barabási-Albert random network $G_{(n,n_0,m)}$ with $n = 150$ nodes, $n_0 = 1$ and $m = 2$.

In Figure 3.17 we have the representation of the hubs of the BA network. For this network, it can be observed that there are approximately six hubs.

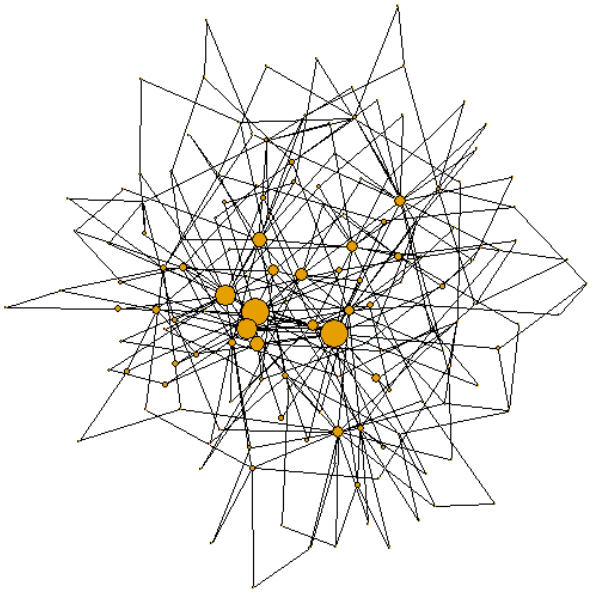


Figura 3.17 Hubs of the Barabási-Albert random network $G_{(n,n_0,m)}$ with $n = 150$ nodes, $n_0 = 1$ and $m = 2$.

In regard to the seed nodes, Table 3.6 presents the most central nodes for each of the measures utilized. For this network, it can be observed that the node x_2 exhibits the highest values for degree, closeness and betweenness centrality and x_1 for both degree and eigenvector centrality. We can also observe that for the degree centrality we have three central nodes, x_1, x_2 and x_7. As illustrated in Figure 3.17, of the six identified hubs, three can be found among the nodes listed in Table 3.6.

Degree centrality	Closeness centrality	Betweenness centrality	Eigenvector centrality
1, 2, 7	2	2	1

Tabela 3.6 Nodes with the highest centrality values for the Barabási-Albert random network $G_{(n,n_0,m)}$ with $n = 150$ nodes, $n_0 = 1$ and $m = 2$.

Figure 3.18 illustrates the propagation process when the seed set is the node x_7, which has the highest degree centrality. The process stopped after 36 iterations, with a total of 72 nodes receiving the message. This implies that approximately 48% of the overall population has received the information.

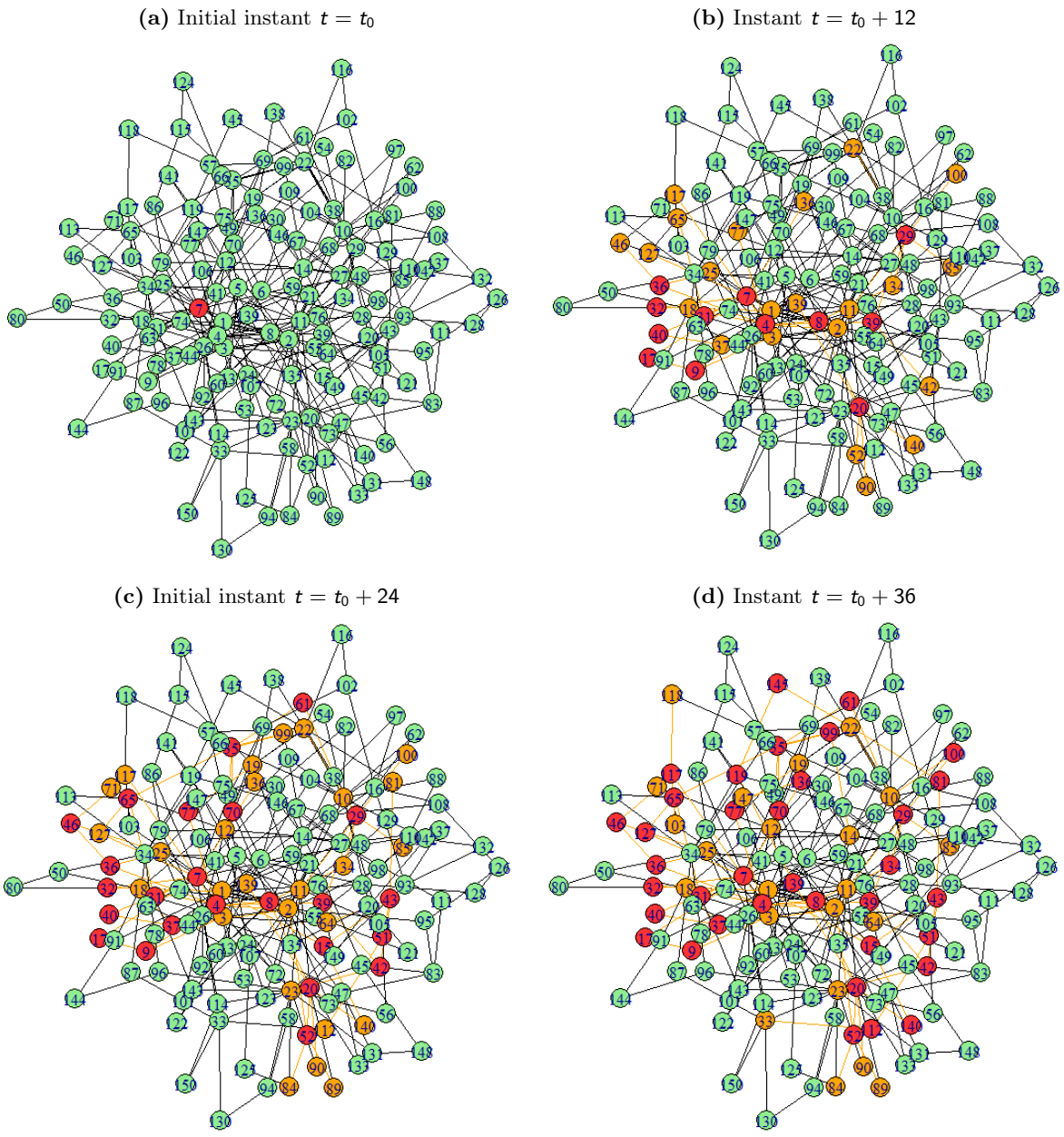


Figura 3.18 Information propagation for the MAM model over the Barabási-Albert random network $G_{(n,n_0,m)}$, with $n = 150$, $n_0 = 1$, $m = 2$, $q = 0.2$, $A_T = 0.4$, $\alpha = 1.9$, $\beta_s = 10$, $\beta_v = 40$ and seed with highest degree centrality.

In Figure 3.19, we commence the process with node x_2, which has the highest closeness and betweenness centrality. In this example, the information diffusion process did not occur, as evidenced by the fact that only one iteration was completed, resulting in a total of two informed nodes.

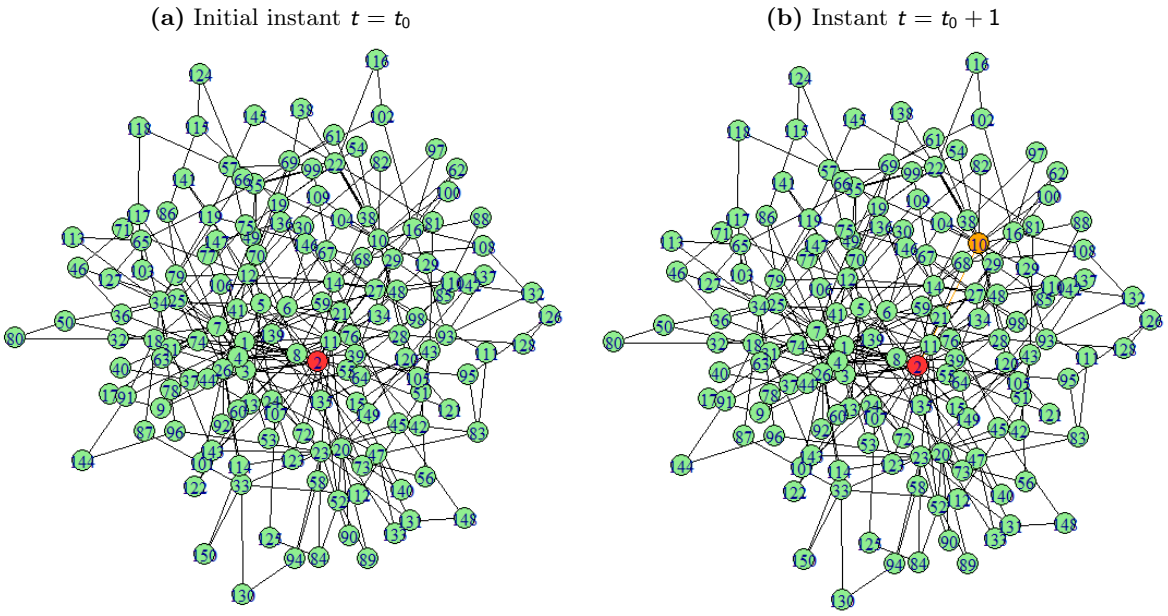


Figura 3.19 Information propagation for the MAM model over the Barabási-Albert random network $G_{(n,n_0,m)}$, with $n = 150$, $n_0 = 1$, $m = 2$, $q = 0.2$, $A_T = 0.4$, $\alpha = 1.9$, $\beta_s = 10$, $\beta_v = 40$ and seed with the highest closeness and betweenness centrality.

Figure 3.20 illustrates the scenario where the information propagation began with the node exhibiting the highest value of eigenvector centrality, that is, node x_1. It can be observed, once again, that the information did not spread throughout the network.

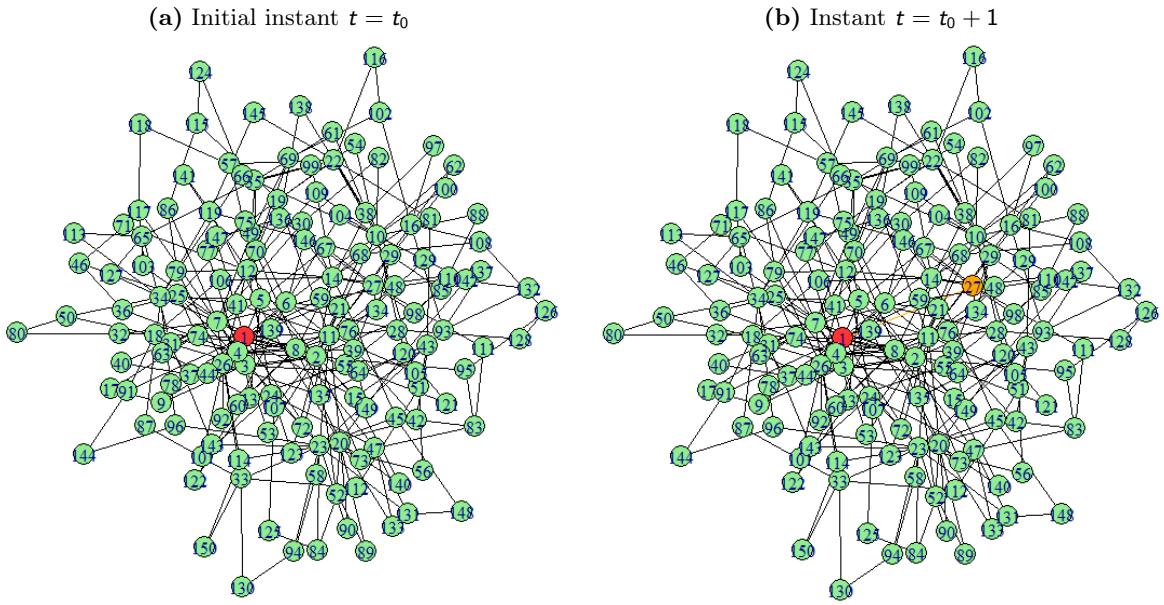


Figura 3.20 Information propagation for the MAM model over the Barabási-Albert random network $G_{(n,n_0,m)}$, with $n = 150$, $n_0 = 1$, $m = 2$, $q = 0.2$, $A_T = 0.4$, $\alpha = 1.9$, $\beta_s = 10$, $\beta_v = 40$ and seed with the highest eigenvector centrality.

Among the centrality measures, the degree centrality yielded the optimal result. Let us now attempt to repeat this process, but with a hub. For this case we use the node x_4 to initiate the process. As illustrated in Figure 3.21, the process reached conclusion after 44 iterations, with a total of 87 informed nodes. This represents a total of 58% of the population that received the messages propagated throughout the network.

As illustrated, when the process initiates with the hub node, there is a greater number of informed nodes when compared to the degree centrality. Consequently, the best choice for an influencer would be node x_4, which is the hub.

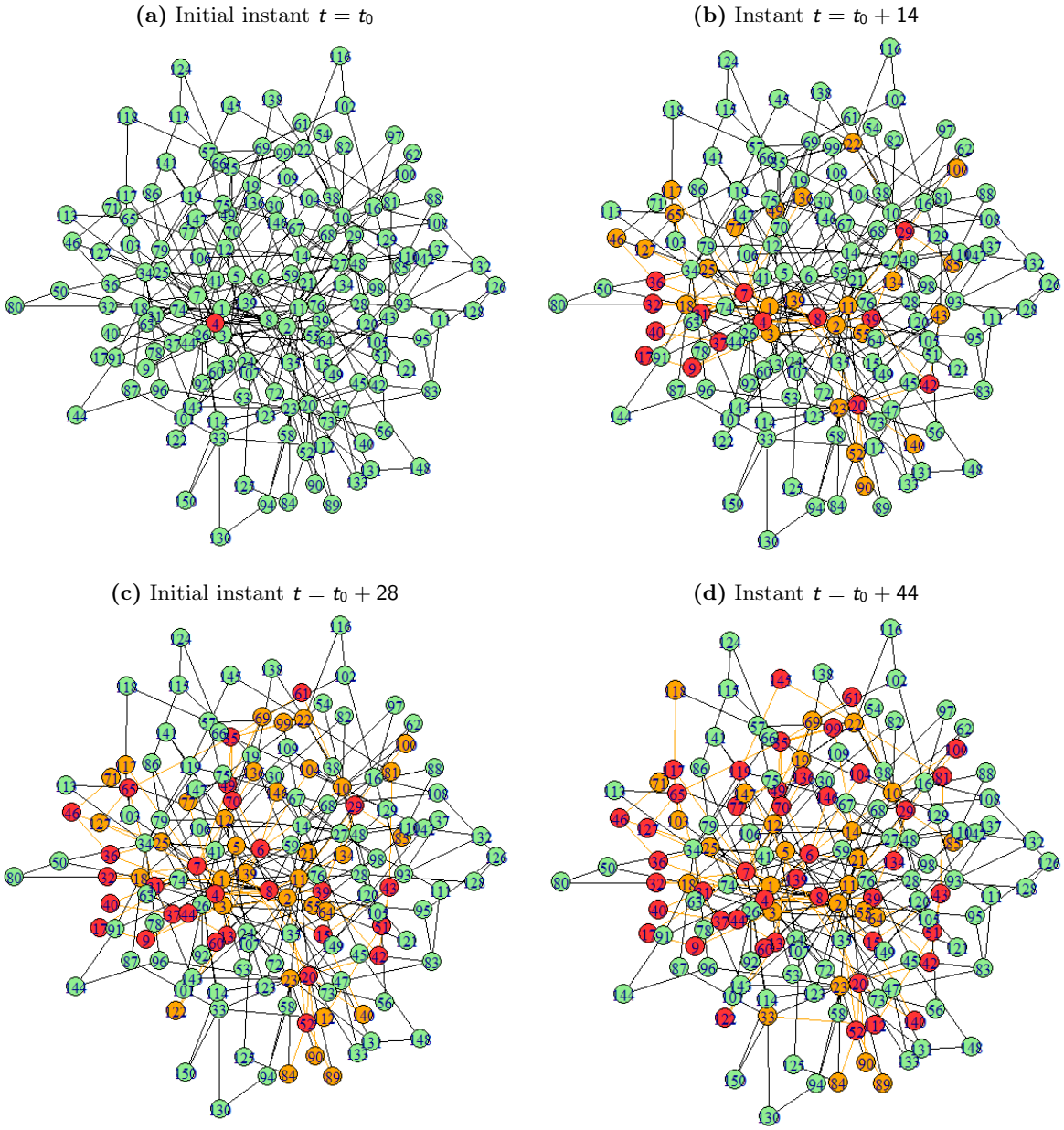


Figura 3.21 Information propagation for the MAM model over the Barabási-Albert random network $G_{(n,n_0,m)}$, with $n = 150$, $n_0 = 1$, $m = 2$, $q = 0.2$, $A_T = 0.4$, $\alpha = 1.9$, $\beta_s = 10$, $\beta_v = 40$ and seed node x_4 a hub of the network.

Capítulo 4

Influencers

This chapter will examine the various methods by which influencers can be identified within a network and subsequently selected to initiate the propagation process. In other words, we will investigate which elements must be in the seed set in order to achieve the highest number of users influenced.

The identification of influencers will be conducted in three distinct ways. The first approach will employ topological centrality measures, specifically in weighted networks. The second will utilize the evidential centrality measure, proposed by Wei *et al*. (Wei et al., 2013), which is based on the theory of belief functions. We will also apply Influence Maximization algorithms, namely a greedy algorithm in (Kempe et al., 2003) and the CELF algorithm by Leskovec *et al.* (Leskovec et al., 2007b).

4.1 Topological Centrality Measures

This Section is devoted to presenting the topological centrality measures, which take into account the structural characteristics of a network. These measures consider various network properties, such as the number of connections, the distance between the nodes or the number of paths within the network. In Section 2.3, topological measures in unweighted networks were discussed. Therefore, the same measures will be reintroduced in this Section, applied to weighted networks.

Let us consider a network $G = (V, E, w)$, where V is the set of nodes, E the set of the edges and w is the function $w : E \mapsto \mathbb{R}^+$ of the weights. The weight of an edge could be an integer or even a probability value, and it can have different meanings depending on the type of problem that the network represents. The weight of the edge $e_{ij} \in E$ is represented by $w_{ij} = w(e_{ij})$, where $w_{ij} \neq 0$.

As previously noted in Wei et al. (2013), there are certain definitions that exhibit slight variations between these two types of networks, such as the definition of distance. In this context, the distance between two nodes in a weighted network is calculated as the minimal sum of the weights between the nodes, corresponding to the value of shortest path (Wei et al., 2013).

The next definition considers the centrality measures adjusted for weighted networks - see (Opsahl et al., 2010):

Definition 8 *Let $G = (V, E, w)$ be a weighted network, where $x_i \in V$ with $i = \{1, 2, ..., n\}$, w_{ij} is the weight of the edge e_{ij}, with $i, j = \{1, 2, ..., n\}$.*

- *Degree centrality in a weighted network: the degree centrality of a node x_i, denoted by d_i^w, is defined by*

$$d_i^w = \sum_j^n w_{ij}$$

where w_{ij} is the weight between nodes i and j, which is greater that 0 if exists a connection between the two nodes.

- *Betweenness centrality in a weighted network: the betweenness centrality of a node x_i, denoted by b_i^w, is given by*

$$b_i^w = \sum_{j,k \neq i} \frac{g_{jk}^w(i)}{g_{jk}^w}$$

where g_{jk}^w is the number of shortest paths between nodes j and k and $g_{jk}^w(i)$ the number of those paths that contain the node x_i.

- *Closeness centrality in a weighted network: the closeness centrality of a node x_i, denoted by c_i^w, is defined by*

$$c_i^w = \left[\sum_j^n d_{ij}^w \right]^{-1}$$

where d_{ij}^w is the minimum distance between the nodes i and j, obtained with the values of w_{ij}.

The distance between two nodes in a weighted network can be calculated through different methods, for example the Dijkstra algorithm (Dijkstra, 1959), for non-negative weights. Consequently, the aforementioned algorithm will be utilized in subsequent simulations with the objective of determining the distances in a weighted network.

4.2 Evidential Centrality Measure

As it was mentioned previously, in this Section we will introduce the evidential centrality measure, proposed by Wei *et al.* (Wei et al., 2013). As this measure is founded upon the theory of belief functions, this topic will be the initial focus of our discussion. Once this foundation has been established, we will proceed to address the measure itself.

4.2.1 Theory of Belief Functions

The theory of belief functions was first introduced by Dempster in (Dempster, 1967) and the basic concepts of this theory were later published by Shafer in (Shafer, 1976). Therefore,

this theory can also be called theory of Dempster-Shafer or Evidence Theory. This theory is frequently regarded as an extension of the Bayesian Theory, because it relies on weaker conditions. The probability attributed to each subset is limited by a lower and an upper bound, which respectively measure the total belief and plausibility for the elements in the subset (Wei et al., 2013). This theory has also the ability of combining pairs of evidence or belief functions to define a new evidence or function. In order to introduce this measure, we will present some of the topics of this theory. For a more detailed overview see Appendix C.

The next definitions illustrate the most important aspects of the Dempster-Shafer Theory used in the formulation of this new centrality measure (Dempster, 1967; Shafer, 1976).

Definition 9 *(Wei et al., 2013) Let $\Omega = \{H_1, H_2, ..., H_N\}$ be a finite set of N elements, and consider the power set $2^{\Omega} = \{\emptyset, H_1, ..., H_N, H_1 \cup H_2, H_1 \cup H_3, ..., \Omega\}$. The power set is called the frame of discernment.*

Definition 10 *(Wei et al., 2013) For a frame of discernment 2^{Ω}, a mass function or a basic probability assignment (BPA) is a mapping $m : 2^{\Omega} \longrightarrow [0, 1]$, that satisfies the following conditions*

$$m(\emptyset) = 0 \qquad \textit{and} \qquad \sum_{A \in 2^{\Omega}} m(A) = 1,$$

where $\emptyset$ is the empty set and A is any element of the frame of the power set 2^{Ω}.

The mass $m(A)$ represents how strongly the evidence supports element A. This theory allows that, given two BPAs, m_1 and m_2, the Dempster Rule can combine them.

Definition 11 *(Wei et al., 2013) The Dempster Rule of Combination, also known as orthogonal sum, is denoted by $m = m_1 \oplus m_2$ and follows the next expression*

$$m(A) = \frac{1}{1 - K} \sum_{B \cap C = A} m_1(B) m_2(C)$$

with

$$K = \sum_{B \cap C = \emptyset} m_1(B) m_2(C),$$

where A, B and C are elements of the power set 2^{Ω} and K is a normalization constant, called the conflict coefficient of two BPAs.

4.2.2 Evidential Centrality Measure

The evidential centrality measure (EVC) is used to identify the influence of a node, taking into account the degree and strength of the nodes of the network, and tries to find a trade off between these two attributes.

Definition 12 *Let $G = (V, E, w)$ be a weighted network, where $w_{ij} \in \mathbb{R}^*$, with $i, j = \{1, 2, ..., n\}$. The strength of a node i is given by:*

$$w_i = \sum_{j}^{n} w_{ij}.$$

To each of these two attributes is assigned a BPA, obtained according to Definition 10. The importance of a node is determined by an evaluation method established by the Dempster Rule of Combination. Wei *et al.* (Wei et al., 2013) mention that "a node with maximum value of degree is more important when only the degree of the node is considered. And the importance of another node is represented by the difference of the degree between two nodes."

BPAs of degree and strength

The first step is to determine a reference value based on the degree and strength of the nodes. They correspond to the maximum and minimum values of the degree and strength, and are given by:

$$\begin{aligned} k_M &= max\{k_1, k_2, ..., k_n\} \\ k_m &= min\{k_1, k_2, ..., k_n\} \\ w_M &= max\{w_1, w_2, ..., w_n\} \\ w_m &= min\{w_1, w_2, ..., w_n\} \end{aligned}$$

with k_i denoting the degree for the ith node.

It is also necessary to define the frame of discernment. In order to apply this method, two evaluation indices are required to assess the influence of the degree and strength: *high* and *low*. So, the frame of discernment Ω will be

$$\Omega = \{high, low\}.$$

The third step is to define the BPAs for the degree and strength of the node. For each node i, we compute the probabilities of *high* and *low* influence for the degree, which are represented by $m_{di}(h)$ and $m_{di}(l)$. The probabilities of *high* and *low* influence for the strength, represented by $m_{wi}(h)$ and $m_{wi}(l)$ respectively, will also be calculated for each node i. The BPAs are given as follows:

$$\begin{aligned} m_{di}(h) &= \frac{|k_i - k_m|}{\sigma} \\ m_{di}(l) &= \frac{|k_i - k_M|}{\sigma} \\ m_{wi}(h) &= \frac{|w_i - w_m|}{\delta} \\ m_{wi}(l) &= \frac{|w_i - w_M|}{\delta} \end{aligned}$$

where σ and δ are given by

$$\begin{aligned} \sigma &= k_M + \mu - (k_m - \mu) = k_M - k_m + 2\mu \\ \delta &= w_M + \varepsilon - (w_m - \varepsilon) = w_M - w_m + 2\varepsilon \end{aligned}$$

with $0 < \mu \leq 1$ and $0 < \varepsilon \leq 1$, where these parameters represent a kind of uncertainty of the order of the node. It is worth highlighting that in a example given by (Wei et al., 2013), they show that the values of μ and ε have no effect on the order of the nodes.

Then, the BPAs for each node i, for the degree and strength, are obtained as follows,

$$\begin{aligned} M_d(i) &= (m_{di}(h), m_{di}(l), m_{di}(\Omega)) \\ M_w(i) &= (m_{wi}(h), m_{wi}(l), m_{wi}(\Omega)) \end{aligned}$$

where

$$m_{di}(\Omega) = 1 - (m_{di}(h) + m_{di}(l))$$

and

$$m_{wi}(\Omega) = 1 - (m_{wi}(h) + m_{wi}(l)).$$

The influence value of the node i is obtained through the use of the Dempster-Shafer Rule of Combination, and represented by

$$M(i) = (m_i(h), m_i(l), m_i(\Omega))$$

where $m_i(\Omega)$ is the probability of *high* or *low*.

Evidential centrality in a weighted network

$M_i(h)$ and $M_i(l)$ are the probabilities of *high* and *low* for the *ith* node, respectively, and follow the next form

$$\begin{aligned} M_i(h) &= m_i(h) + \frac{1}{2m_i(\Omega)} \\ M_i(l) &= m_i(l) + \frac{1}{2m_i(\Omega)}. \end{aligned}$$

The greater the value of $M_i(h)$, the more significant the given node is. Conversely, a node with a greater value of $M_i(l)$ is considered to be less important. The evidence measure is thus given by the next definition.

Definition 13 *(Wei et al., 2013) The EVC measure for each node i, with $i = \{1, 2, ..., n\}$, denoted by $evc(i)$, follows the next expression*

$$evc(i) = M_i(h) - M_i(l) = m_i(h) - m_i(l),$$

where $evc(i)$ is a real number and the higher its value the more important the node is.

The results of the information propagation process using the most influential nodes according to the EVC measure will be shown in Section 4.4. In this Section we will also show the difference in information diffusion, when used the node obtained by the EVC measure and by the weighted centrality measures.

4.3 Influence Maximization Algorithms

The Influence Maximization Problem, as explained in Section 2.1, is the selection of a set of people who have the ability to influence a large fraction of users of a social network. This problem revolves around answering two fundamental questions: how to estimate the influence of each user of the network, and how to find the set that maximizes influence (Jendoubi, 2016). In this Section we will explore Influence Maximization models that use information propagation models in the maximization process.

Domingos and Richardson were the first to introduce the influence identification problem (Domingos and Richardson, 2001). In 2003, Kempe *et al.* (Kempe et al., 2003), formulated the influence problem as an optimization problem and proved that their models were NP-hard. The problem they ran into was finding the seed set that maximizes message propagation in the network.

Considering a social network $G = (V, E)$, with V being the node set and E the edge set, and a diffusion model M, the Influence Maximization Problem is to select a set S of k nodes that maximizes knowledge in the network under study. This means, choosing an initial set S that maximizes the expected number of influenced nodes, $\sigma_M(S)$. Kempe *et al.* (Kempe et al., 2003) propose the use of the LTM and ICM for the estimation of $\sigma_M(S)$ and prove that maximizing σ_M is NP-Hard, and that σ_M is submodular and monotone. For the maximization of σ_M, they use the greedy algorithm with the Monte Carlo simulation.

In the literature, there are several works focused on improving the execution time when using the LTM and ICM, such as:

- Cost Effective Lazy Forward (CELF) by Leskovec *et al.* (Leskovec et al., 2007b), is a greedy algorithm that explores the submodularity property of the function that is being maximized;
- Shortest-Path Model (SPM) introduced by Kimura and Saito (Kimura and Saito, 2006), it is a special case of the ICM and in this model only the shortest paths are considered in the activation process;
- Bozorgi *et al.* (Bozorgi et al., 2016) considered the community structure, where they used the LTM to find the influencers inside each community. Those were the local influencers. Then they estimated the global influence of those nodes using again the LTM. The influence of a node is the combination of its local and global influence. In the end, they select the set of nodes that maximize the influence on the network.

Another point of view on this problem is the models that focus on improving the quality of the selection process of the influencers or they consider other important parameters. Wang *et al.* (Wang et al., 2016) introduced the Weighted Independent Cascade (WIC) model, which is an extension of the ICM, where this one considers the attributes of the nodes of the network. The purpose of this model is to maximize the value of the influenced nodes. In the next Sections the focus will be on the greedy algorithm by Kempe *et al.* (Kempe et al., 2003) and the CELF

algorithm by Leskovec *et al.* (Leskovec et al., 2007b).

4.3.1 Greedy Algorithm

The greedy algorithm is an algorithm to find a solution for the Influence Maximization Problem. This Section demonstrates a greedy algorithm, which can be employed to efficiently approximate an optimal solution within a factor of $(1 - 1/e)$ (Nemhauser et al., 1978; Kempe et al., 2015). To use this greedy-based solution, the objective function σ has to be a submodular and monotone set function, defined on the power set 2^V to $\mathbb{R}$, where V is the node set of the network and 2^V is the set of all subsets of V.

According to Nemhauser *et al.* (Nemhauser et al., 1978) when we consider a finite set N and a real-valued function z defined on the set of subsets of N that satisfies

$$z(S) + z(T) \geq z(S \cup T) + z(S \cap T), \ \ \forall T \in N$$

then z is said submodular.

In this context, the function σ is submodular if it satisfies a "diminishing returns" property, which means, that the marginal gain of adding an element x to a input set S is at least as high as adding the same element to a superset T of S, the following way (Jendoubi, 2016):

$$\sigma(S \cup \{x\}) - \sigma(S) \geq \sigma(T \cup \{x\}) - \sigma(T)$$

whenever $S \subseteq T \subseteq V$ and $x \in V$. Besides, σ is a monotone increasing function if

$$\sigma(S) \leq \sigma(T)$$

whenever $S \subseteq T \subseteq V$. A greedy-based solution can be adapted to maximize any monotone submodular set function σ that has $\sigma(\emptyset) = 0$. The greedy algorithm presented by Kempe *et al.* (Kempe et al., 2003) is given as follows:

Algorithm 5 Greedy Algorithm

```
S = ∅
// S is the seed set
while |S| ≤ k do
    u ← argmax_{x∈V\S} marginalGain(x)
    S ← S ∪ {u}
end while
```

In each step, the algorithm estimates the marginal gain of each node $x \in V$, with respect to S. The marginal gain is defined as the influence gain of a given node with respect to the current S. Then, it chooses the node with the biggest marginal gain, until S gets k elements.

4.3.2 CELF Algorithm

The CELF algorithm introduced by Leskovec *et al.* (Leskovec et al., 2007b), is also a greedy-based maximization solution. This algorithm explores the submodularity property of the

objective fuction, to minimize the number of calls of the marginal gain function. The submodularity garantees that the marginal gain decreases with the size of the solution. The CELF algorithm is given as follows:

Algorithm 6 CELF Algorithm

$S = \emptyset$
// S seed set
$Q = \emptyset$
// Q sorted node list, in decreasing order according to its marginal gain
for each $u \in V$ **do**
    marginalGain(u)
    // a function that estimates the marginal gain of u in relation to S
    Q.add(u)
end for
while $|S| \leq k$ **do**
    $v \leftarrow Q.pop()$
    marginalGain(v)
    if $v.MG \geq Q, getFirst().MG$ **then**
        S.add(v)
    end if
    Else Q.add(v)
end while

This algorithm differs from the greedy algorithm in the sense that it does not estimate the marginal gain for every of the expected nodes in each iteration. Instead, it calculates the marginal gain for each node in the first iteration and then sorts the nodes according to their marginal gain. This sorted list is maintained for subsequent iterations.

In the subsequent iteration, the algorithm eliminates the node that is situated at the initial position in the list, which is the node with the highest value of marginal gain. Then it recalculates the marginal gain for the remaining nodes. If the node that has been removed continues to occupy the initial position in the list, it is added to S. Otherwise, the marginal gain for the node that is in the initial position of the list is recalculated, and the process is repeated.

4.4 Results

This Section will be divided into two parts. First, we will examine the differences in information diffusion when the EVC measure is employed, in comparison to the weighted centrality measures previously mentioned. Second, we will analyze the discrepancies between the solutions of the Influence Maximization algorithms and the nodes obtained through the topological centrality measures, presented in Section 2.3. For the second case, the Influence Maximization algorithms used the ICM to identify the most influential nodes of the correspondent social

network.

In all of the examples, the social networks utilized will be BA random networks, model $G_{(n,n_0,m)}$ (Barabási and Albert, 1999), with $n = 150$, $n_0 = 1$ and $m = 2$. After having the seed nodes identified in both examples, the propagation models employed will be the weighted LTM for the first case, while in the second will the unweighted LTM.

In the first case the weights of the network were obtained with an uniform distribution $U \in [0, 1]$, and regarding the LTM the threshold value is $\theta = 0.4$ for every user of the social network. In the second case the only parameter defined was the threshold value which is also $\theta = 0.4$ for every node.

A descriptive study of the network's characteristics will be conducted for both cases, with measures such as maximum and minimum degree, diameter, average path length, and transitivity being exposed. The implementation of these examples can be found in Appendix D.

4.4.1 EVC Measure

For this case, we have a BA network with $n = 150$ individuals, as shown in Figure 4.1. This network was obtained with the BA random network algorithm, where we started the process with $n_0 = 1$, that is, a single vertex and no edges in the first time step. Then the algorithm proceeds to execute the requisite steps in order to attain the network with 150 nodes. Once the final network configuration has been determined, the weighted LTM may be applied.

Table 4.1, shows some characteristics of the network, like its maximum degree, minimum degree, diameter, average path length and transitivity.

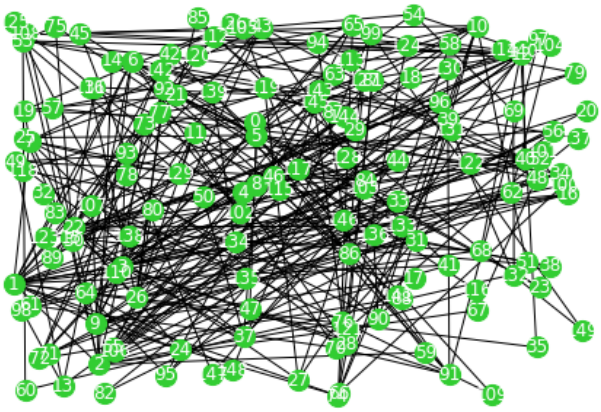


Figura 4.1 Barabási-Albert random network $G_{(n,n_0,m)}$ with $n = 150$ nodes, $n_0 = 1$ and $m = 2$.

Max degree	Min degree	Diameter	Average path length	Transitivity
20	2	2.88	1.2	0.046

Tabela 4.1 Descriptive characteristics of the Barabási-Albert random network $G_{(n,n_0,m)}$ with $n = 150$ nodes, $n_0 = 1$ and $m = 2$.

According to Table 4.1 this BA network has diameter $diam(G) = 2.88$. Therefore, it can be stated that a path of no more than 2.88 edges is sufficient to link two nodes.

Figure 4.2 (a) depicts the histogram of the BA network in question. It is evident that the majority of nodes possess a degree less than four, with a minimal number of nodes exhibiting high degree values. Figure 4.2 (b) illustrates the proportion of nodes with a given degree. For instance, approximately 51% of the nodes have two neighbors, 14% have three and 12% four. This indicates that more than 77% of the nodes have a degree of less than four.

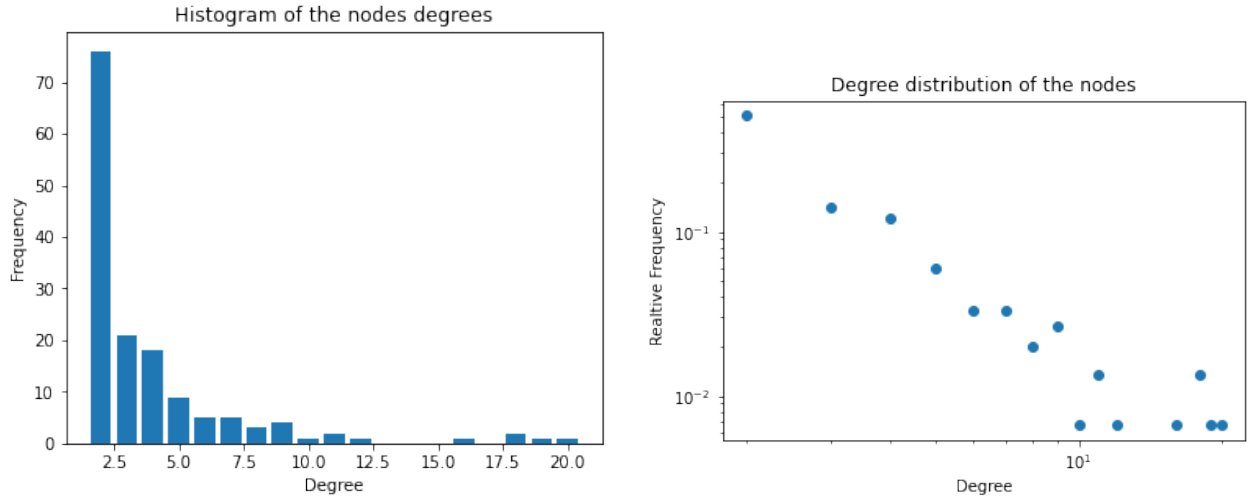


Figura 4.2 **(a)** Histogram of the node degrees of the Barabási-Albert random network $G_{(n,n_0,m)}$ with $n = 150$ nodes, $n_0 = 1$ and $m = 2$; **(b)** Degree distribution of the nodes of the Barabási-Albert random network $G_{(n,n_0,m)}$ with $n = 150$ nodes, $n_0 = 1$ and $m = 2$.

As in Chapter 3, to initiate the propagation process, we need to select the seed nodes. In this case we will use the weighted centrality measures mentioned in Section 4.1, where Table 4.2 shows the nodes with the highest values of the centrality measures, i.e., degree, closeness and betweenness, as well as the node obtained with the EVC measure. For this network, it can be observed that the node x_3 exhibits the highest values for degree and betweenness centrality.

EVC	Degree centrality	Closeness centrality	Betweenness centrality
1	3	5	3

Tabela 4.2 Nodes with the highest centrality values for the Barabási-Albert random network $G_{(n,n_0,m)}$ with $n = 150$ nodes, $n_0 = 1$ and $m = 2$.

The requisite elements have now been assembled to initiate the simulation of the information propagation process with the weighted LTM.

Figure 4.3 illustrates the propagation process when the seed set is the node x_1 (green color), which is the node with the highest value of EVC. The nodes in red represent the individuals who are not informed (inactive). It is important to highlight again, that the nodes for this case, only have two states informed (active) or not informed (inactive). The process ceased after 7

iterations, with 144 nodes informed, that means that 96% of the population was influenced.

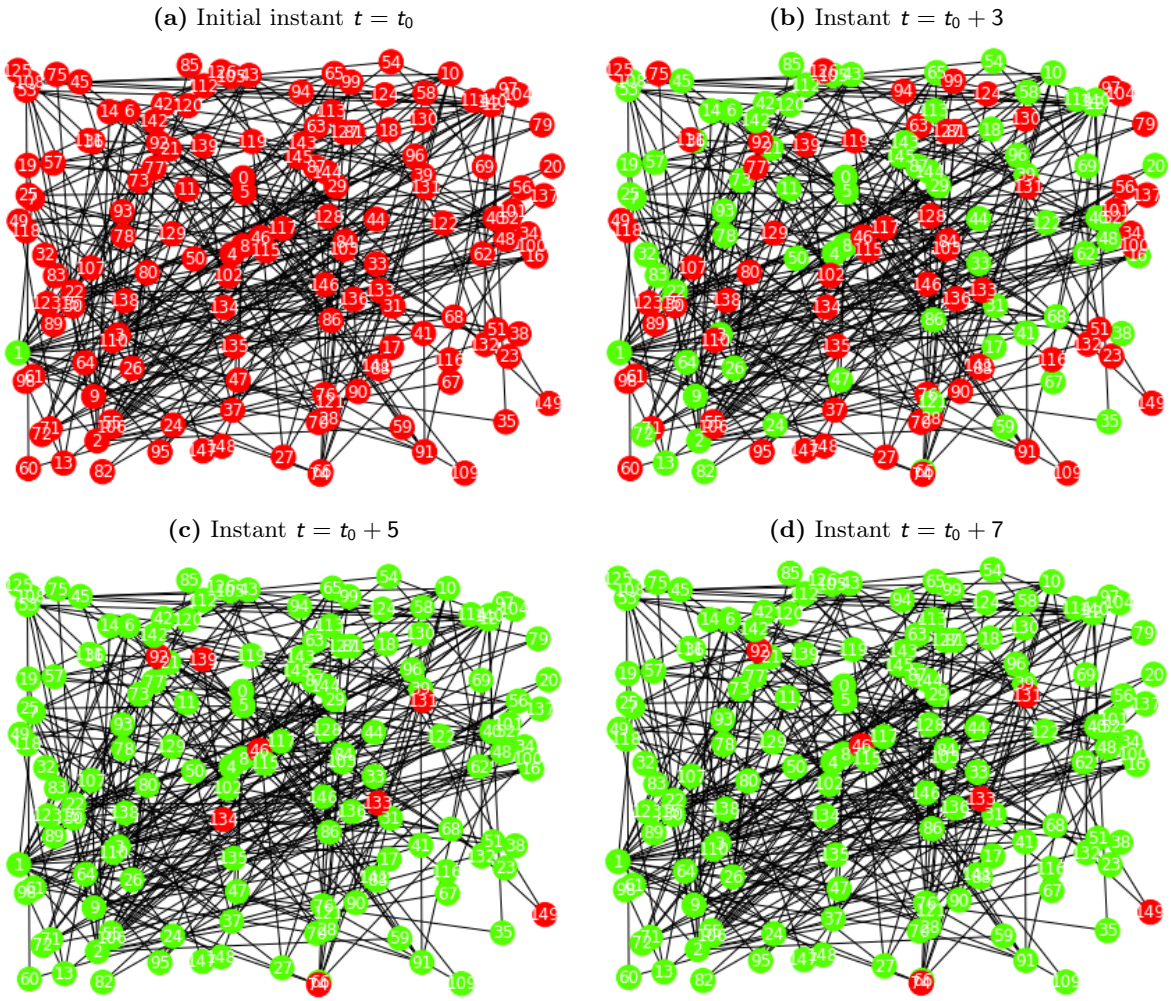
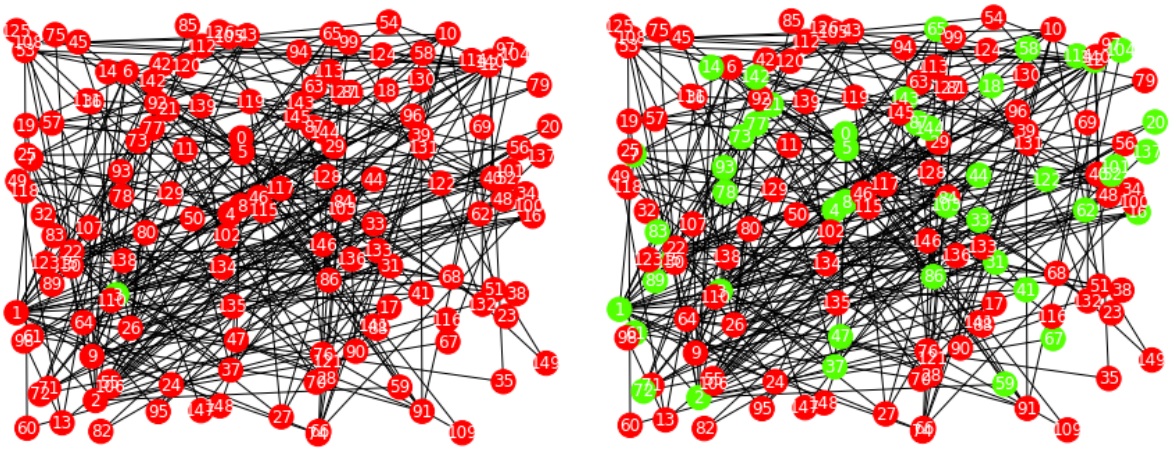


Figura 4.3 Information propagation for the weighted LTM over the Barabási-Albert random network $G_{(n,n_0,m)}$, with $n = 150$, $n_0 = 1$, $m = 2$, $\theta = 0.4$ and seed with the highest EVC.

In Figure 4.4, we initiate the process with node x_3, which has the highest value for the degree and betweenness centrality. In this example, the information diffusion process terminated after 6 iterations, resulting in a total of 144 informed nodes. This represents that approximately 96% of the social network's nodes are influenced. A comparison with the previous example reveals that both seed sets give similar results.

(a) Initial instant $t = t_0$

(b) Instant $t = t_0 + 2$



(c) Instant $t = t_0 + 4$

(d) Instant $t = t_0 + 6$

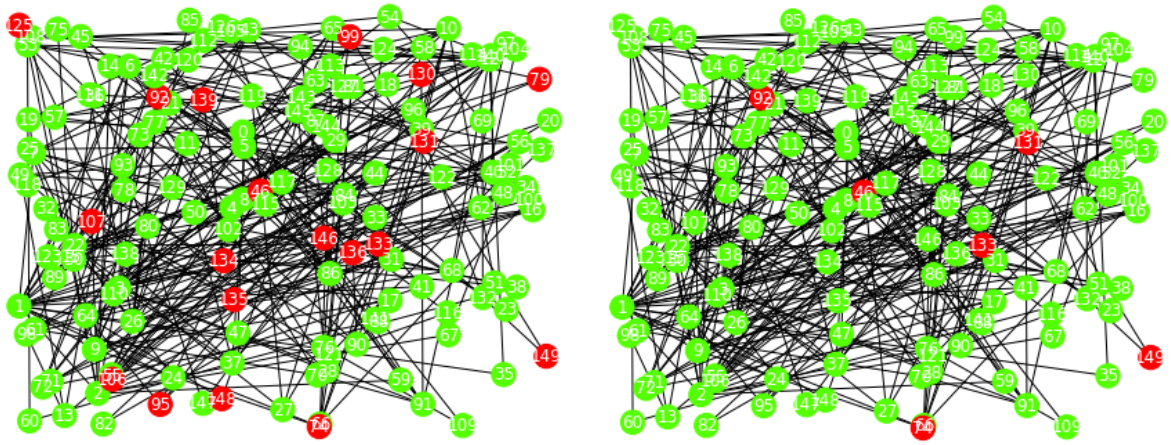


Figura 4.4 Information propagation for the weighted LTM over the Barabási-Albert random network $G_{(n,n_0,m)}$, with $n = 150$, $n_0 = 1$, $m = 2$, $\theta = 0.4$ and seed with the highest value for degree and betweenness centrality.

Figure 4.5 illustrates the scenario where the information propagation began with the node exhibiting the highest value of closeness centrality, that is, node x_5. It can be observed that this nodes gives the exact same result as the previous example, 144 influenced nodes after 6 iterations, i.e., once again 96% of the population is influenced.

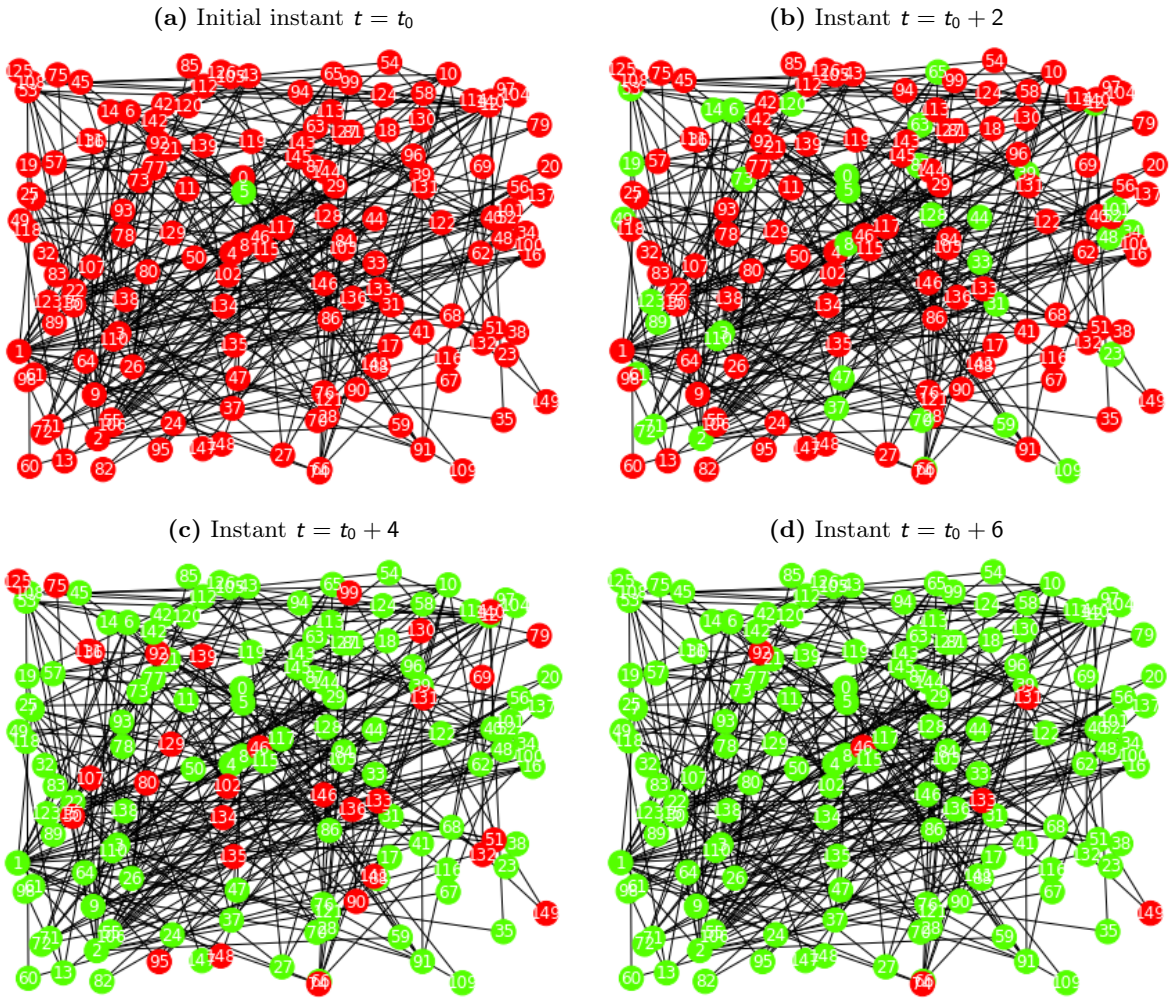


Figura 4.5 Information propagation for the weighted LTM over the Barabási-Albert random network $G_{(n,n_0,m)}$, with $n = 150$, $n_0 = 1$, $m = 2$, $\theta = 0.4$ and seed with the highest closeness centrality.

To conclude we can see that for these three different seed sets we obtain the same results, which means that the EVC measure is not a better choice to identify influencers. Figure 4.6 shows the growth of the number of influenced nodes throughout the iterations and we see that the increase in influenced nodes is very similar in the three examples. It therefore makes no difference which metric we use to select the influencers in this network.

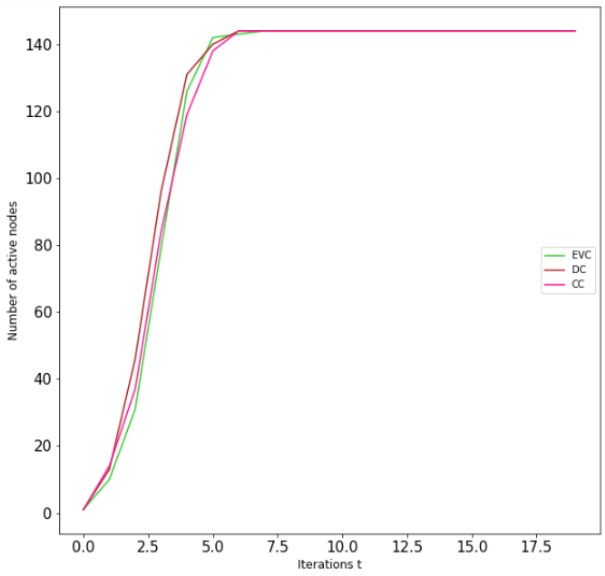


Figura 4.6 Growth of the number of active nodes of the Barabási-Albert random network $G_{(n,n_0,m)}$ with $n = 150$ nodes, $n_0 = 1$ and $m = 2$, for the EVC measure, degree and closeness centrality.

4.4.2 Influence Maximization Algorithms

As in the preceding section, we commence our process with a BA network featuring $n = 150$ individuals and $m = 2$, as illustrated in Figure 4.7. A descriptive summary of this network's characteristics is presented in Table 4.3.

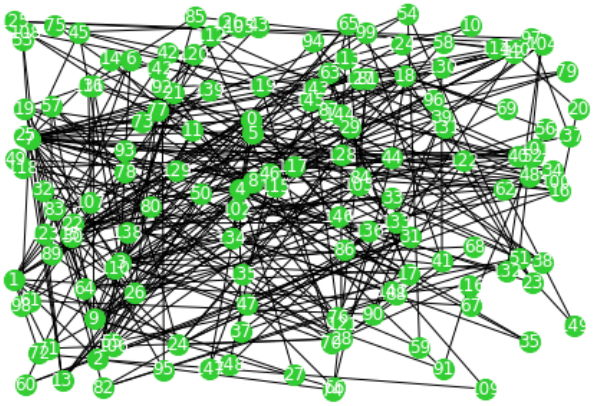


Figura 4.7 Barabási-Albert random network $G_{(n,n_0,m)}$ with $n = 150$ nodes, $n_0 = 1$ and $m = 2$.

Max degree	Min degree	Diameter	Average path length	Transitivity
25	2	6	3.37	0.044

Tabela 4.3 Descriptive characteristics of the Barabási-Albert random network $G_{(n,n_0,m)}$ with $n = 150$ nodes, $n_0 = 1$ and $m = 2$.

According to Table 4.3 this BA network has diameter $diam(G) = 6$. It can therefore be stated that a path of no more than 6 edges is sufficient to traverse between any two given nodes. The average path length indicates that it typically takes 3.37 edges to connect one node with another.

Figure 4.8 (a) illustrates the histogram of the BA network. As with the previous example, the majority of nodes possess a degree less than four, with a minimal number of nodes exhibiting high degree values. Figure 4.8 (b) depicts the proportion of nodes with a given degree. For instance, approximately 42% of the nodes have two neighbors, 22% have three and 16% four. This indicates that 80% of the nodes have degree of less than four.

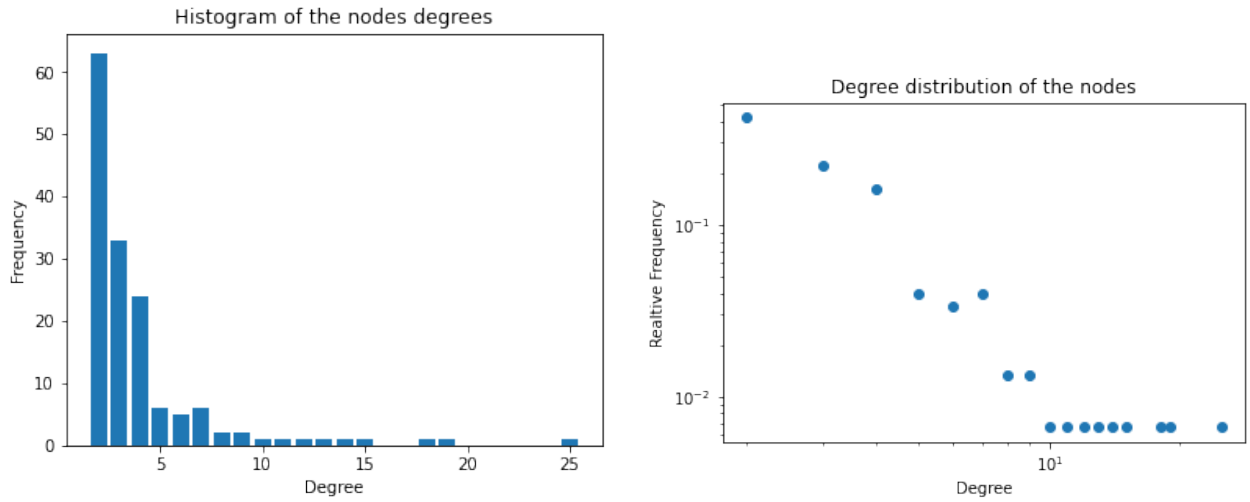


Figura 4.8 **(a)** Histogram of the node degrees of the Barabási-Albert random network $G_{(n,n_0,m)}$ with $n = 150$ nodes, $n_0 = 1$ and $m = 2$; **(b)** Degree distribution of the nodes of the Barabási-Albert random network $G_{(n,n_0,m)}$ with $n = 150$ nodes, $n_0 = 1$ and $m = 2$.

For this case we will use the unweighted centrality measures mentioned in Section 2.3, where Table 4.4 shows the nodes with the highest values of the centrality measures, as well as the nodes obtained with the greedy and CELF algorithms. For this network the algorithms took, approximately 1 minute and 21 seconds for the greedy and 1 minute and 41 seconds for the CELF, to select the best seed nodes.

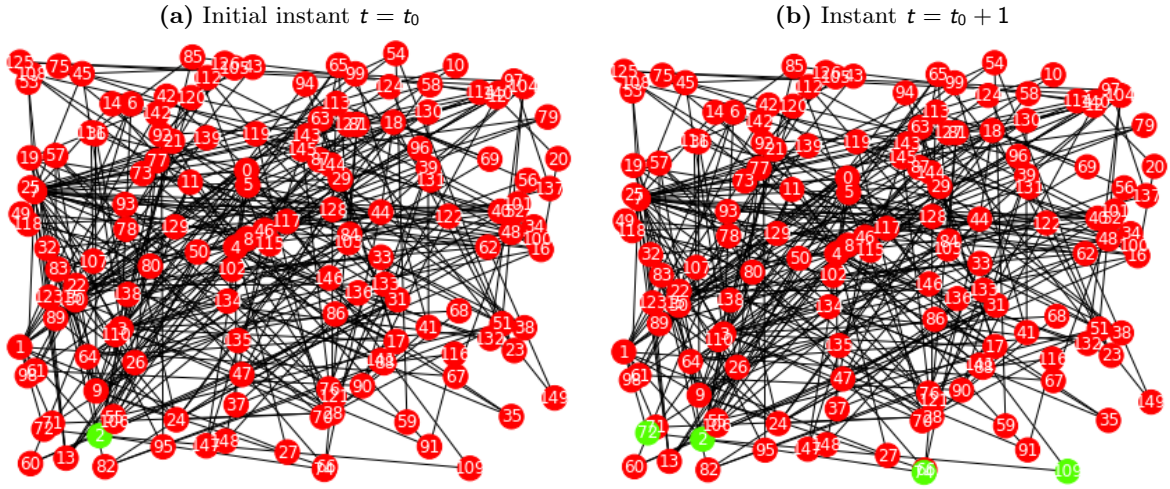
It can be observed that both algorithms got the same output, i.e., node x_2. As for the centrality measures, we can see that x_3 exhibits the highest values for degree, betweenness and eigenvector centrality.

Greedy	CELF	Degree centrality	Closeness centrality	Betweenness centrality	Eigenvector centrality
2	2	7	3	7	7

Tabela 4.4 Nodes with the highest centrality values for the Barabási-Albert random network $G_{(n,n_0,m)}$ with $n = 150$ nodes, $n_0 = 1$ and $m = 2$.

We have gathered all the elements necessary to start the propagation process with the unweighted LTM.

Figure 4.9 illustrates the propagation process when the seed set is the node x_2 (green color) which is the node obtained with the Influence Maximization algorithms. The nodes in red represent, once again, the individuals who are inactive. The process ceased after 1 iteration, with 4 nodes influenced, that means that, only 2% of the population was influenced. This seed set is not a good choice for a influencer.



(a) Initial instant $t = t_0$ **(b)** Instant $t = t_0 + 1$

Figura 4.9 Information propagation for the LTM over the Barabási-Albert random network $G_{(n,n_0,m)}$, with $n = 150$, $n_0 = 1$, $m = 2$, $\theta = 0.4$ and seed node obtained with the greedy and CELF algorithms.

In Figure 4.10, we initiate the process with the node x_7, which has the highest value for the degree, betweenness and eigenvector centrality. In this example, the information diffusion process terminated after 1 iteration, resulting in a total of 10 informed nodes. This represents approximately 7% of influenced users in the population. A comparison with the previous example reveals that with the same number of iterations we got 2.5 more influenced users, however it is still a very low amount of users in the active state.

(a) Initial instant $t = t_0$ (b) Instant $t = t_0 + 1$

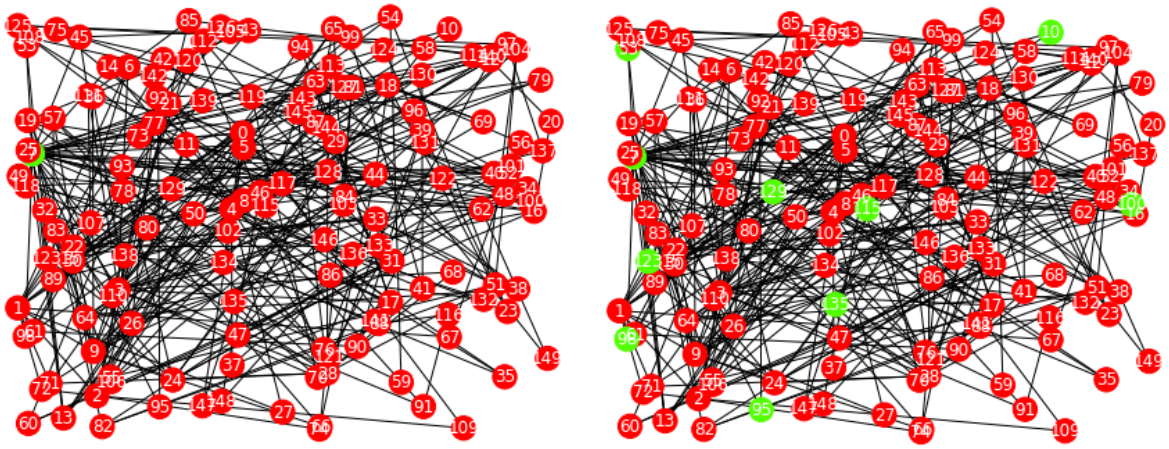
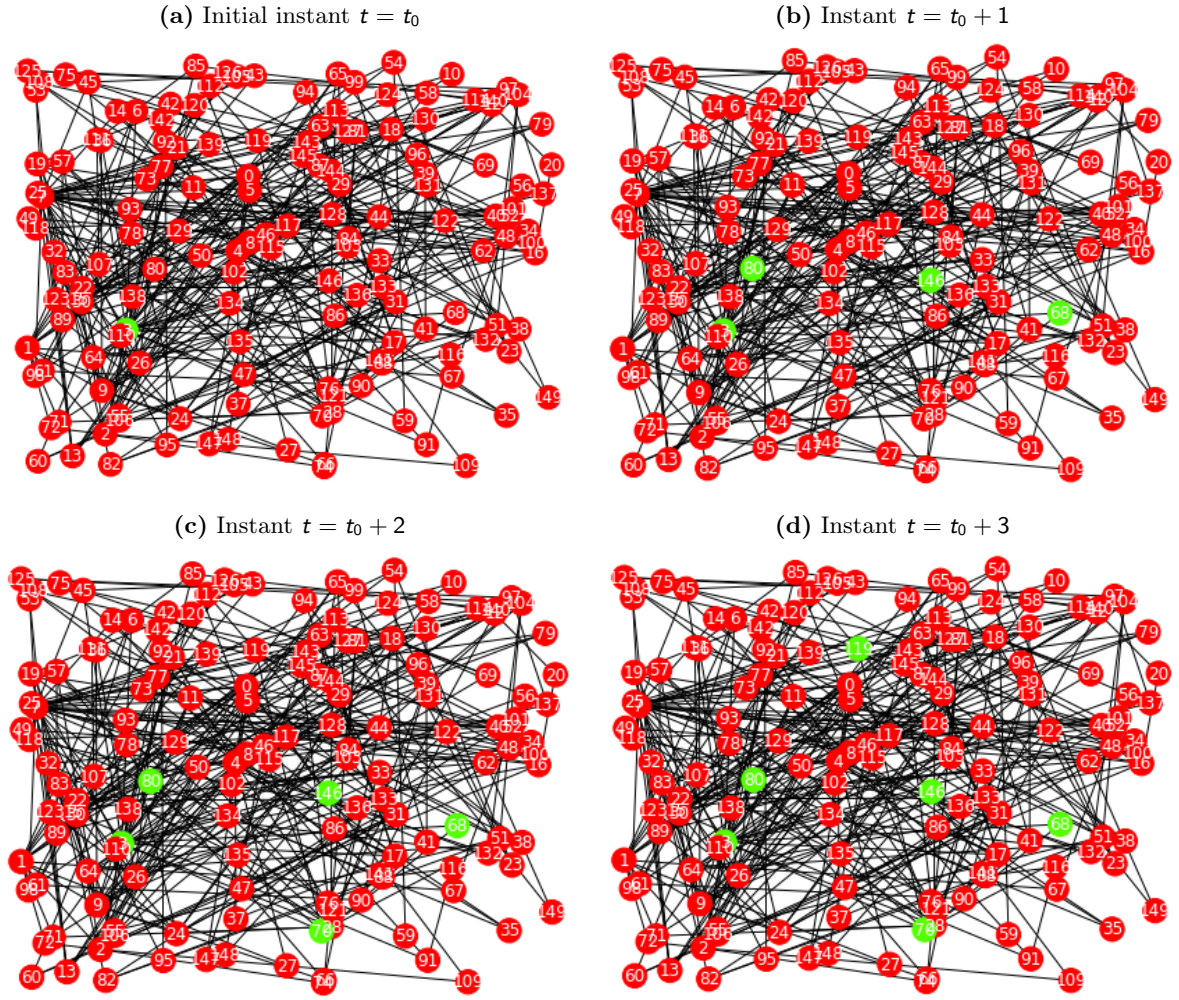


Figura 4.10 Information propagation for the LTM over the Barabási-Albert random network $G_{(n,n_0,m)}$, with $n = 150$, $n_0 = 1$, $m = 2$, $\theta = 0.4$ and seed node with the highest value of degree, betweenness and eigenvector centrality.

Figure 4.11 illustrates the scenario where the information propagation began with the node exhibiting the highest value of closeness centrality, that is, node x_3. For this node the process stopped after 3 iterations with a total of 6 nodes influenced, which represents 4% of the population.

In conclusion, it can be observed that for the threshold value $\theta = 0.4$, the three distinct seed sets yielded did not exhibit the characteristics of an influencer, as evidenced by the minimal percentage of users influenced across the three examples. Nevertheless, if we were to select one, it would be the node x_7, which is the node with the highest value of degree, betweenness and eigenvector centrality, as it achieved the greatest quantity of users influenced in the shortest amount of iterations.



(a) Initial instant $t = t_0$ **(b)** Instant $t = t_0 + 1$

(c) Instant $t = t_0 + 2$ **(d)** Instant $t = t_0 + 3$

Figura 4.11 Information propagation for the LTM over the Barabási-Albert random network $G_{(n,n_0,m)}$, with $n = 150$, $n_0 = 1$, $m = 2$, $\theta = 0.4$ and seed with the highest closeness centrality.

Capítulo 5

Conclusion

5.1 Conclusions and future works

In this master thesis, we started by introducing the Influence Maximization Problem and elucidate its most salient features. Along with it the definition of seed set and the its importance in this context. Furthermore, we illustrated the procedure described in (Rocha et al., 2023c) for the LTM in ER random networks.

Subsequently, two epidemic models, namely the Susceptible-Infected-Recovered (SIR) and the Susceptible-Infected-Susceptible (SIS) models, were presented, along with an exposition of their relationship with the information propagation process. In order to illustrate the probabilistic procedures of both models, the seed set was redefined, and the results obtained were presented. These results were derived from the references cited in the text, namely (Rocha et al., 2023a,b)

In the final section of this chapter, we observed the unweighted centrality measures and illustrated their application in social networks for the identification of influencers. To this end, we presented two examples: one pertaining to the Influence Maximization Problem, where the identified influencers were successful, and another based on the SIR model, which led us to conclude that these measures solely consider the position of the nodes on the network, without incorporating external factors.

In order to gain an understanding of how the propagation of various agents occurs, we explored the MAM, studied by Argaiz (Argaiz, 2015). The MAM is a model that was created, based on the interest that an individual has on the information that is being spread, and implemented on BA random networks $G_{(n,n_0,m)}$ (Barabási and Albert, 1999). The characteristics and particularities of this model were studied, and simulations were created to view the propagation process. In the case studies related to this model, the centrality measures were used initially. However, due to the varying nature of the networks, success was not always achieved with these measures. Consequently, the process was observed from the perspective of a hub, and it was concluded that hubs are a suitable representation for influencers..

The last chapter focus on the different methods of identifying the influencers, instead of doing a emphasis on the propagation models. In this chapter the identification of influencers

was conducted in three distinct ways. The first approach employed was weighted centrality measures, the second was the EVC measure, proposed by Wei *et al.* (Wei et al., 2013), which is a measure based on the theory of belief functions and then by Influence Maximization algorithms, namely the greedy algorithm in (Kempe et al., 2003) and the CELF algorithm by Leskovec *et al.* (Leskovec et al., 2007b). The simulations were divided into two distinct categories. The initial analysis demonstrated the distinction between the propagation process when the seed set was selected based on weighted centrality measures and when it was selected based on the EVC measure. Our examples indicated that there was not a significant difference between these two approaches. In this case, the BA random network model was employed to translate the social network, and the weighted LTM was utilized as the propagation model. The objective of the other case was to investigate the distinction between the influence maximization algorithms and the unweighted centrality measures. The selected network for this case was a BA random network, and the unweighted LTM was applied. The findings indicated that neither approach was effective in identifying the optimal influencers.

It would be interesting to attempt to simulate the aforementioned cases in the future, but in more complex networks, such as the Dorogovtsev-Mendes networks (Dorogovtsev and Mendes, 2003). An optimal progression for this work would be to apply the former tools in a real-life social network to gain insight into their efficacy.

Anexo A

Propagation Models

A.1 Galton-Watson Branching Model

The Galton-Watson Model is a simple stochastic model that is used to describe the growth of a given population over generations (Harris, 1963). These generations are periods of successive population growth, where each generation has a population of individuals and, each individual has the ability to generate offspring randomly.

The idea of this model is as follows:

- Initial generation (Generation 0): It is considered an initial number of individuals. Each of the individuals can generate a set of descendants at random, and the number of descendants is distributed according to a Poisson probability distribution.
- Next generations: Each individual of the next generations follows the same process, that is, each individual in generation n generates randomly a set of descendants, and the number of descendants follows the same probability distribution.

The probability distribution innate to the number of descendants is known as the "fertility law". If this distribution is a Poisson distribution with parameter λ, the Galton-Watson Branching model is called the "Poisson Branching Process". The expression for the probability distribution of the total number of descendants in n generations is given by:

$$P(N_n = k) = \frac{\lambda^k \cdot e^{-\lambda}}{k!}$$

where N_n is the total number of descendants in generation n and λ is the parameter of the Poisson distribution.

Therefore, the Galton-Watson model is used to understand stochastic population growth and is a fundamental tool in population theory and in stochastic process theory (Williams, 1991).

Anexo B

Statistical Distributions

B.1 Power-Law Distribution

The power law is a specific form of probability distribution that describes the relationship between two variables, indicating that one quantity is proportional to a negative power of the other. This distribution is characterized by heavy tails, that is, extreme events are more likely, than in symmetric distributions, such as the normal distribution.

The power law distribution, usually, is given by:

$$P(X \geq x) \propto x^{-\alpha}$$

where $P(X \geq x)$ is the probability of a random variable X being greater or equal to x and α represents the shape parameter, who is always a positive value.

The most important property of the power law distribution is that, it does not have a finite mean when $\alpha \leq 2$, and this means that the variance can be infinite, which will imply a high probability of occurrence of extreme events. The power law distribution is seen in phenomena where values are extremely large compared to most other values. This distribution is an important part of complex network theory and is often used to model the behaviour of complex systems and data distributions that exhibit a large degree of heterogeneity, see (Clauset et al., 2009).

B.2 Pareto's Distribution

The Pareto Distribution is a continuous probability distribution, often used to model phenomena where there is a large variation between parameters (Arnold, 2008). The probability density function of the Pareto Distribution is given by:

$$f(x; x_m; a) = \frac{a \cdot x_m^{\alpha}}{x^{\alpha-1}}, \text{ for } x \geq x_m, \ \alpha > 0$$

where x_m is the minimum value in the distribution support and α is the shape parameter.

This distribution is characterized by the existence of a long tail, which means that extremely large values are possible, although they become increasingly unlikely as we move away from the minimum value of x_m.

The fact that the Pareto distribution is a heavy-tailed distribution means that it describes rare and extreme events more frequently than more common distributions, such as the normal distribution. In practice, the Pareto distribution is often used to analyze data that exhibits large variation, especially when one is interested in the extreme values or tails of the distribution. The Pareto distribution is a particular case of the power law distribution, being characterized by its specific form and application in modelling heavy tails (Raftery, 1994).

Anexo C

Theory of Belief Functions

In this appendix, we give a overview of the theory of belief functions, where we detail its basic concepts (Jendoubi, 2016). First, we show some concepts that are used to model information and how to present the different pieces of information in a same universe. Next, we get to the information fusion and finally, the decision making step where we present some well known tools that are often used for decision making in the belief functions framework.

C.1 Information Modeling

In this section, we will present a series of fundamental functions that are essential for modeling information in order to facilitate its processing. First, we will introduce the basic belief assignment, which is also referred to as mass function. We will also present a number of its associated transformations.

C.1.1 Mass Function

To use the theory of belief functions, it is first necessary to define the frame of discernment, which is the set of all possible choices or decisions in the problem in question. Lets consider that $\Omega = \{C_1, C_2, ..., C_n\}$ is our frame of discernment where $C_i \cap C_j = \emptyset$, and $C_i, C_j \in \Omega$. The conjunction between elements of Ω is not allowed. The power set, 2^Ω is the set of all subset of Ω, which is defined by:

$$2^\Omega = \{\emptyset, \{C_1\}, \{C_2\}, \{C_1, C_2\}, ..., \{C_1, C_2, ..., C_n\}\}$$

The mass function or basic belief assigment (BBA), m^Ω, is given by:

$$2^\Omega \longrightarrow [0, 1]$$

$$A \mapsto m^\Omega(A)$$

The quantity $m^\Omega(A)$ is the mass value attributed to subset $A \subseteq \Omega$. The mass function has the following property

$$\sum_{A \subseteq \Omega} m^\Omega(A) = 1.$$

When $m^{\Omega}(A) > 0$, A is called the *focal element*. If $m^{\Omega}(\emptyset) = 0$, m^{Ω} is a normalized mass function. In many cases we may have $m^{\Omega}(\emptyset) \geq 0$. The mass given to the empty set is the mass value that is not given to any other subset. To this we call a fusion inconsistency value. It appears when we combine many pieces of information and its generally caused by the non idempotence of the combination rule and by the conflict between information sources, that is, by the contradiction degree between them. It can redistributed using the next transformation:

$$\begin{aligned} m^{\Omega}(A) &= \frac{m^{\Omega}(A)}{1 - m^{\Omega}(\emptyset)} \\ m^{\Omega}(\emptyset) &= 0 \end{aligned}$$

The mass value given to set Ω is the mass that can not be given to its subsets and it is called total ignorance. When we compare a BBA distribution with a probability distribution, the BBA allows a subset of Ω to be a focal element when there is a doubt about the decision, while the probability theory forces the equiprobability.

We have a simple mass function or simple BBA, when there are two focal elements, the first being a subset of Ω, $A \subseteq \Omega$ and the other being Ω (Shafer, 1976; Smets, 1990). Let $\alpha \in [0, 1]$, a simple BBA m^{Ω} is given by

$$m^{\Omega}(A) = \begin{cases} 1 - \alpha, & A \subseteq \Omega \\ \alpha, & A = \Omega \\ 0, & \text{otherwise} \end{cases} .$$

A consonant mass it is characterized by its nested focal elements , i.e., $A_1 \subseteq A_2 \subseteq ... \subseteq \Omega$.

C.1.2 Mass Transformations

The transformations of the mass function allow different ways of presenting the same information. The belief function bel^{Ω} represents the minimal amount of support given to subset A. The bel^{Ω} function is also a mapping from $2^{\Omega} \rightarrow [0, 1]$. The belief of A is defined by:

$$bel^{\Omega}(A) = \begin{cases} 0, & \text{se } A = \emptyset \\ \sum\limits_{\emptyset \neq B \subseteq A} m^{\Omega}(B), & \forall A \subseteq \emptyset, \ A \neq \emptyset \end{cases} .$$

The mass function m^{Ω} that produces bel^{Ω} can be retrieved by the following expression:

$$m^{\Omega}(A) = \sum_{B \subseteq A} (-1)^{|A|-|B|} bel^{\Omega}(B), \ \forall A \subseteq \Omega.$$

The plausibility function pl^{Ω} represents the maximum amount of support that can be given to subset A if other information becomes available (Shafer, 1976). It is also a mapping from

$2^{\Omega} \rightarrow [0,1]$ and given by:

$$\begin{aligned} pl^{\Omega}(A) &= bel^{\Omega}(\Omega) - bel^{\Omega}(\overline{A}), \ \ \forall A \subseteq \Omega \\ pl^{\Omega}(A) &= \sum_{B \cap A = \emptyset} m^{\Omega}(B), \forall A \subseteq \Omega. \end{aligned}$$

The mass function that produces pl^{Ω} can be retrieved through:

$$m^{\Omega}(A) = \sum_{B \subseteq A} (-1)^{|A|-|B|} pl^{\Omega}(\overline{B}), \ \ \forall A \subseteq \Omega.$$

C.1.3 From Probability to BBA

The transformation from probability to BBA is called consonant transformation or inverse pignistic transformation (Aregui and Denoeux, 2007, 2008). Let Pr^{Ω} be the probability distribution in Ω, to transform Pr^{Ω} in m^{Ω}, first we need to sort the probabilities given to the singletons of Ω:

$$Pr^{\Omega}(C_1) \geq Pr^{\Omega}(C_2) \geq ... \geq Pr^{\Omega}(C_n)$$

Then, we use the following expression to get the BBA:

$$\begin{aligned} m^{\Omega}(\{C_1, C_2, ..., C_n\}) &= n.Pr^{\Omega}(C_n) \\ m^{\Omega}(\{C_1, C_2, ..., C_{n-1}\}) &= (n-1).(Pr^{\Omega}(C_{n-1}) - Pr^{\Omega}(C_n)) \\ &... \\ m^{\Omega}(\{C_1, C_2\}) &= (2).(Pr^{\Omega}(C_2) - Pr^{\Omega}(C_3)) \\ m^{\Omega}(\{C_1\}) &= (1).(Pr^{\Omega}(C_1) - Pr^{\Omega}(C_2)). \end{aligned}$$

C.2 Information Fusion

The fusion of information grants the combination of many pieces of information derived from independent and different sources (Jendoubi, 2016). For the fusion of information we can use different combination rules, such as: Dempster's Rule (Dempster, 1967), the Conjunctive Combination Rule (Smets, 1990), Dubois and Prade's Rule (Dubois and Prade, 1988), Yager's Rule (Yager, 1987) and the PRC6 Rule (Martin and Osswald, 2006, 2007).

The Dempster Rule was the first one to be used for the combination of evidence. Considering two mass functions m_1^{Ω} and m_2^{Ω}, from two different sources, the combined BBA distrubution $m_{1\oplus 2}^{\Omega}$ is given by:

$$m_{1\oplus 2}^{\Omega}(A) = \begin{cases} \frac{\sum_{B \cap C = A} m_1^{\Omega}(B).m_2^{\Omega}(C)}{1 - \sum_{B \cap C = \emptyset} m_1^{\Omega}(B).m_2^{\Omega}(C)}, & \forall A \subseteq \Omega, A \neq \emptyset \\ \\ 0, \ \text{ se } A = \emptyset \end{cases}.$$

This BBA is normalized, that is, $m_{1\oplus 2}(\emptyset) = 0$.

The Conjuctive Rule of Combination (CRC) (Smets, 1990) also combines two mass functions m_1^Ω and m_2^Ω that come from distinct sources:

$$m_{1\otimes 2}^\Omega(A) = \sum_{B\cap C=A} m_1^\Omega(B).m_2^\Omega(C), \ \ \forall A \subseteq \Omega$$

and it is not normalized, i.e.

$$m_{1\otimes 2}^\Omega(A) = \sum_{B\cap C=A} m_1^\Omega(B).m_2^\Omega(C) \geq 0.$$

C.3 Decision Making

This phase can be achieved by using the pignistic probability BeP^Ω. It is a mapping from $\Omega \rightarrow [0,1]$ and it is calculated from the combined BBA m^Ω, i.e., given by,

$$BeP^\Omega(C_i) = \sum_{d_i\in A, \ \ A\in\Omega} \frac{m^\Omega(A)}{\#A.(1-m^\Omega(\emptyset))}$$

where $\#A$ is the cardinality of the subset A. The decision made according to the pignistic probability is the decision C_i, having the highest pignistic probability value (Smets, 2005).

Anexo D

Simulations Code

This appendix contains the entirety of the code utilized to generate the simulations presented throughout this work. The code associated with Chapter 3 will be presented initially, including the code for the MAM. The code for the MAM algorithm was derived from the final assignment for the Graphs and Networks course, authored by the students Beatriz Vieira, Cátia Pinto and Sofia Rocha. Subsequently, the algorithm was modified to align with the specific requirements of this project. In the end, the code associated with chapter 4 will be presented, which includes all the programming for the EVC measure and Influence Maximization Algorithms.

D.1 Viral Marketing

This section presents the code utilized throughout Chapter 3. The code utilized in this chapter was developed using the R programming language.

```
library(igraph)

# Network parameters
n <- 150
m <- 2

set.seed(230)
rede <- barabasi.game(n = n, m = m , directed = FALSE)
l <- layout_nicely(rede)
plot(rede, vertex.size = 8, layout=l, vertex.color = "lightgreen",
     edge.width = 1, edge.color = "black")

# Network characterization

# Diamter
diameter(rede, directed=F, weights=NA)
```

```r
# Node degrees
deg <- degree(rede, mode="all")
hist(deg, main="Histogram of the node degrees", xlab = "Degrees",
     breaks=seq(min(deg),max(deg),1), xlim = c(min(deg), max(deg)))

max(deg)
min(deg)

# Degree distribution
deg.dist <- degree_distribution(rede, cumulative=F, mode="all")
plot(x=0:max(deg), y=deg.dist, pch=19, cex=1.2, col="orange",
     xlab="Degree", ylab="Relative Frequency",
     main = "Degree distribution of the nodes")

# Hubs
hs <- hub_score(rede, weights=NA)$vector
plot(rede, vertex.size=hs*10, main="Hubs", layout = l, vertex.label = NA,
     edge.width = 1, edge.color = "black")

# Average path length
mean_distance(rede, directed=F)

# Transitivity
transitivity(rede, type="global")

# Centrality measures

# Degree centrality
degree.cent <- centr_degree(rede, mode = "all")
graus <- as.vector(degree.cent$res)

cg = c()
for(i in 1:length(graus)){
  max = max(graus)
  if(graus[i] == max){
    cg = c(cg, i)
  }
}
```

```
cat('The nodes with the highest degree centrality is', cg)

# Closeness centrality
closeness.cent <- closeness(rede, mode="all")
centralidades <- as.vector(closeness.cent)

cc = c()
for(i in 1:length(centralidades)){
  max = max(centralidades)
  if(centralidades[i] == max){
    cc = c(cc, i)
  }
}

cat('The nodes with the highest closeness centrality is', cc)

# Betweenness Centrality
between.centr <- betweenness(rede, directed = T, weights=NA)
intermadiacoes <- as.vector(between.centr)

bc = c()
for(i in 1:length(intermadiacoes)){
  max = max(intermadiacoes)
  if(intermadiacoes[i] == max){
    bc = c(bc, i)
  }
}

cat('The nodes with the highest betweenness centrality is', bc)

# Eigenvector centrality
eigen.centr <- eigen_centrality(rede, directed = F, weights=NA)
vps <- as.vector(eigen.centr$vector)

ec = c()
for(i in 1:length(vps)){
  max = max(vps)
  if(vps[i] == max){
    ec = c(ec, i)
  }
```

```
}

cat('The nodes with the highest eigenvector centrality is', ec)

### MAM model

# Parameters to be used in each propagation simulation
q <- 0.2
At <- 0.4
alpha <- 1.9
beta.s <- 10
beta.v <- 40

# Network parameters needed for propagation:
# maximum degree and network edges
kmax <-max(degree(rede))
arestas <- get.edgelist(rede)

# List for each node in the network with its neighbors
vizinhos <- vector("list", length = n)
for(no in 1:n){
  vizinhos[[no]] <- neighbors(rede, no)
}

# Giving affinity values to the nodes in the network, using a
# uniform distribution
set.seed(1)
u <- runif(n)
afinidade <- sample(n, (1-q)*n)

V(rede)$afinidade <- 0
V(rede)$afinidade[afinidade] <- u[afinidade]

# Function to determine the probability value of each node
# being chosen as a node to receive the message
Prob_Random <- function(no){
  prob_random <- 1 - (V(rede)$afinidade[vizinhos[[no]]] - At)
  prob_random[prob_random > 1] <- 0.9
  return(prob_random)
}
```

```
Prob_Max <- function(no){
  prob_max <- 1 - (V(rede)$afinidade[vizinhos[[no]]] - At)
  prob_max[prob_max > 1] <- 0.1
  return(prob_max)
}

# Function to determine the value of (r1)_n
Reco_Ns <- function(no){
  set.seed(1)
  r <- seq(1, (kmax), by = 1)
  prob_uni <- runif(n)

  somatorio.s <- sum(1/(beta.s + r^alpha))
  somatorio.v <- sum(1/(beta.v + r^alpha))

  h.s <- 1/somatorio.s
  h.v <- 1/somatorio.v

  value.s <- 0
  value.v <- 0
  sub_list.s <- list()
  sub_list.v <- list()
  for(i in r){
    value.s <- value.s + (h.s/(beta.s + i^alpha))
    sub_list.s <- append(sub_list.s, abs(value.s - prob_uni[no]))

    value.v <- value.v + (h.v/(beta.v + i^alpha))
    sub_list.v <- append(sub_list.v, abs(value.v - prob_uni[no]))
  }

  r.s <- which.min(sub_list.s)
  r.v <- which.min(sub_list.v)

  if(no == no_semente){
    return(r.s)
  } else{
    return(r.v)
  }
}
```

```r
# seeds: centrality measures
no_semente <- cg
no_semente <- cc
no_semente <- bc
no_semente <- ec

# seed: hubs
no_semente <- 4

nos_informados <- rep(F, n)

nos_informados[no_semente] <- TRUE
print(nos_informados)

# Message propagation process
set.seed(1)

V(rede)$color <- "lightgreen"
E(rede)$color <- "black"
V(rede)$color[no_semente] <- "#FF3333"
plot(rede, vertex.color = V(rede)$color, layout = l, vertex.size = 8,
     main = "Iteration 0", edge.width = 1)
nos_selecionados <- NULL

infetados <- length(no_semente)

j = 1

while(any(nos_informados)){

  for(i in seq_along(nos_informados)){
    if(nos_informados[i] == TRUE){
      no <- i
      break
    }
  }

  an <- V(rede)$afinidade[no]
  r1 <- Reco_Ns(no)
```

```
vizinhos_no <- vizinhos[[no]]

for(i in 1:n){
  vizinhos[[i]] <- vizinhos[[i]][vizinhos[[i]] != no]
}

rn <- round(abs(an - At) * r1)
rn[rn < 1] <- 1

if(rn >= length(vizinhos_no)){
  nos_selecionados <- vizinhos_no
} else{
  if(an < At){
    nos_selecionados <- sample(vizinhos_no, rn, prob = Prob_Random(no))
  }else{
    nos_selecionados <- sample(vizinhos_no, rn, prob = Prob_Max(no))
  }
}

for(nos in nos_selecionados){
  nos_informados[nos] <- (V(rede)$afinidade[nos] > At)
}

arestas <- all_simple_paths(rede, no, to = V(rede)[nos_selecionados],
                            cutoff = 1)

for(i in seq_along(arestas)){
  E(rede, path = arestas[[i]])$color <- "orange"
}

if(any(no_semente %in% nos_selecionados)){
  nos_selecionados <- nos_selecionados[!nos_selecionados %in% no_semente]
}

V(rede)$color[nos_selecionados] <- "orange"
V(rede)$color[no] <- "#FF3333"
set.seed(1)
plot(rede, vertex.color = V(rede)$color, edge.color = E(rede)$color,
     edge.width = 1, vertex.size = 8, layout = l,
     main = paste("Iteration", j))
```

```
  nos_informados[no] <- FALSE
  no <- NULL

  infetados <- infetados + length(nos_selecionados)
  print(paste("Nodes who receive the message in iteration ", j))
  print(infetados)
  j <- j+1
}
```

D.2 Influencers

This section presents the code utilized in Chapter 4, which encompasses the code for the EVC measure and the code for the Influence Maximization algorithms. The code was developed using Python.

D.2.1 Evidential Centrality Measure

```
import ndlib.models.ModelConfig as mc
import ndlib.models.epidemics as ep
import matplotlib.pyplot as plt
import networkx as nx
import numpy as np
import matplotlib as mpl
import pandas as pd
from operator import itemgetter

# Network
n = 150
m = 2
G = nx.barabasi_albert_graph(n, m, seed = 202)
nx.draw(G, pos=nx.random_layout(G, seed=1), with_labels = True,
        font_color='white', node_size=200, node_color='limegreen')

# Number of connected components
nx.number_connected_components(G)

# Weights of the network
# Generating the weights
np.random.seed(80)
```

```
pesos = []
for i in range(len(G.edges())):
    w = np.random.uniform(0,1)
    pesos.append(round(w, 4))

keys = G.edges()
values = pesos

weights = dict(zip(keys, values))

nx.set_edge_attributes(G, values = weights, name = 'weight')

nx.is_weighted(G)

# Network characteristics
def eccentricity(G, v=None, sp=None, weight=None):
order = G.order()
e = {}
for n in G.nbunch_iter(v):
    if sp is None:
        length = nx.shortest_path_length(G, source=n, weight=weight)

        L = len(length)
    else:
        try:
            length = sp[n]
            L = len(length)
        except TypeError as err:
            raise nx.NetworkXError('Format of "sp" is invalid.') from err
    if L != order:
        if G.is_directed():
            msg = (
                "Found infinite path length because the digraph is not"
                " strongly connected"
            )
        else:
            msg = "Found infinite path length because the graph is not"
            " connected"
        raise nx.NetworkXError(msg)
```

```
        e[n] = max(length.values())

    if v in G:
        return e[v]
    return e

def diameter(G, e=None, usebounds=False, weight=None):
    if usebounds is True and e is None and not G.is_directed():
        return _extrema_bounding(G, compute="diameter", weight=weight)
    if e is None:
        e = eccentricity(G, weight=weight)
    return max(e.values())

# Diameter
print('Diameter ', diameter(G, weight='weight'))

# Average path lenght
print('Average path length ',
      nx.average_shortest_path_length(G, weight = 'weight'))

# Transitivity
print('Transitivity ', nx.transitivity(G))

# Histogram of the node degrees
degree_sequence = sorted((d for n, d in G.degree()), reverse=True)
dmax = max(degree_sequence)

fig = plt.figure("Histogram of the nodes degrees", figsize=(10, 10))
axgrid = fig.add_gridspec(5, 4)

ax2 = fig.add_subplot(axgrid[3:, 2:])
ax2.bar(*np.unique(degree_sequence, return_counts=True))
ax2.set_title("Histogram of the nodes degrees")
ax2.set_xlabel("Degree")
ax2.set_ylabel("Frequency")

fig.tight_layout()
plt.show()
```

```
# Degree distribution
degree_freq = nx.degree_histogram(G)
degrees = range(len(degree_freq))

degrees_freq_relative = []
for i in range(len(degree_freq)):
    fr = degree_freq[i]/sum(degree_freq)
    degrees_freq_relative.append(fr)

plt.figure(figsize=(6, 4))
plt.loglog(degrees[2:], degrees_freq_relative[2:],'o')
plt.title('Degree distribution of the nodes')
plt.xlabel('Degree')
plt.ylabel('Realtive Frequency')

# Calculations for the EVC
# Degrees
graus = []
for i in G.degree():
    graus.append(i[1])

kmax = max(graus)
kmin = min(graus)

print('Maximum degree = ', kmax)
print('Minimum degree = ', kmin)

# Strength
strength = []
for i in G.degree(weight = 'weight'):
    strength.append(round(i[1], 4))

wmax = max(strength)
wmin = min(strength)

print('Maximum strength = ', wmax)
print('Minimum strength = ', wmin)

# Miu and epsilon
np.random.seed(1)
```

```
miu = round(np.random.uniform(0,1), 4)
epsilon = round(np.random.uniform(0,1), 4)

print('miu = ', miu)
print('epsilon = ', epsilon)
print()

#Sigma and delta
sigma = round(kmax - kmin + (2*miu), 4)
delta = round(wmax - wmin + (2*epsilon), 4)

print('sigma = ', sigma)
print('delta = ', delta)

# theta = {empty, h, l, theta}
# BPA functions
def Md_i(no):
    mdi_h = round(abs(graus[no] - kmin)/sigma, 4)
    mdi_l = round(abs(graus[no] - kmax)/sigma, 4)
    mdi_theta = round(1 - (mdi_h + mdi_l), 4)
    return (mdi_h, mdi_l, mdi_theta)

def Mw_i(no):
    mwi_h = round(abs(strength[no] - wmin)/delta, 4)
    mwi_l = round(abs(strength[no] - wmax)/delta, 4)
    mwi_theta = round(1 - (mwi_h + mwi_l), 4)
    return (mwi_h, mwi_l, mwi_theta)

Md_nos = []
Mw_nos = []

for i in G.nodes():
    x = Md_i(i)
    Md_nos.append(x)
    y = Mw_i(i)
    Mw_nos.append(y)

# Functions given by applying the Dempster-Shafer rule, for h, l and theta
def M_i(no):
    k_i = (Md_nos[no][0]*Mw_nos[no][1]) + (Md_nos[no][1]*Mw_nos[no][0])
```

```
    mi_h = ((Md_nos[no][0]*Mw_nos[no][0]) + (Md_nos[no][0]*Mw_nos[no][2]) +
            (Mw_nos[no][0]*Md_nos[no][2]))/(1-k_i)
    mi_l = ((Md_nos[no][1]*Mw_nos[no][1]) + (Md_nos[no][1]* Mw_nos[no][2]) +
            (Mw_nos[no][1] *Md_nos[no][2]))/(1-k_i)
    mi_theta = (Md_nos[no][2]* Mw_nos[no][2])/(1-k_i)
    return (round(mi_h,4), round(mi_l, 4), round(mi_theta, 4))

M_no = []
for i in G.nodes():
    a = M_i(i)
    M_no.append(a)

# M_i function for h and l
Mi_h = []
Mi_l = []
for no in G.nodes():
    x = M_no[no][0] + 1/(2* M_no[no][2])
    Mi_h.append(round(x,4))
    y = M_no[no][1] + 1/(2* M_no[no][2])
    Mi_l.append(round(y,4))

# EVC measure
evcs = []

for i in G.nodes():
    evc = Mi_h[i] - Mi_l[i]
    evcs.append(round(evc,4))

# Centrality measures
# EVC
sementes = []

for i in range(len(evcs)):
    if evcs[i] == max(evcs):
        print('The most influencial node is '+ str(i) +
              ' with EVC = ' + str(max(evcs)))
        sementes.append(i)

dg = G.degree(weight = 'weight')
cg = list(dg)
```

```
print('The node with highest degree centrality is ' +
      str(max(cg, key=itemgetter(1))[0]) + ' with value ' +
      str(max(cg, key=itemgetter(1))[1]))

sementes.append(max(cg, key=itemgetter(1))[0])

# Closeness centrality
# Dijkstra algorithm
pred = []
dist = []
for i in G.nodes():
    pred_i, dist_i = nx.dijkstra_predecessor_and_distance(G, i,
                                                          weight = 'weight')
    pred.append(list(sorted(pred_i.items())))
    dist.append(list(sorted(dist_i.items())))

# Weighted closeness centrality
cc = []

for i in range(len(dist)):
    somatorio_i = 0
    for j in range(len(dist[0])):
        somatorio_i = somatorio_i + dist[i][j][1]
    cc_i = 1/somatorio_i
    cc.append(cc_i)

for i in range(len(cc)):
    if cc[i] == max(cc):
        print('The most influencial node is '+ str(i)
              + ' with closeness centrality = ' + str(max(cc)))
        sementes.append(i)

# Betweenness centrality
btc = nx.betweenness_centrality(G, weight = 'weight')
ci = list(btc.values())

for i in range(len(ci)):
    if ci[i] == max(ci):
        print('The most influencial node is '+ str(i) +
```

```
                ' with betweenness centrality = ' + str(max(ci)))
        sementes.append(i)

# Eigenvector centrality
eig = nx.eigenvector_centrality(G, weight = 'weight', max_iter = 400)
cvp = list(eig.values())

for i in range(len(cvp)):
    if cvp[i] == max(cvp):
        print('The most influencial is '+ str(i) +
              ' with eigenvector centrality = ' + str(max(cvp)))
        sementes.append(i)

# Weighted LT model
r = []

G = nx.barabasi_albert_graph(n, m, seed = 202)
nx.is_weighted(G)

# Model selection
model = ep.GeneralThresholdModel(G)

# Model Configuration
config = mc.Configuration()
config.add_model_initial_configuration("Infected", [sementes[2]])

# Setting node and edges parameters
threshold = 0.4
weight = pesos
if isinstance(G, nx.Graph):
    nodes = G.nodes
    edges = G.edges

arestas = list(edges)
for i in nodes:
    config.add_node_configuration("threshold", i, threshold)
for e in range(len(arestas)):
    config.add_edge_configuration("weight", arestas[e], weight[e])

model.set_initial_status(config)
```

```
# Simulation execution
its = 20
iterations = model.iteration_bunch(its)

lista = []
linha = []
for j in range(its):
    y = iterations[j]['status']
    for k in range(len(G.nodes)):
        if j == 0 :
            linha.append(y[k])
lista.append(linha)


nova_linha = []
for s in range(1, its):
    y = iterations[s]['status']
    v1 = list(y.keys())
    v2 = list(y.values())
    nova_linha = lista[s-1].copy()
    for d in range(len(v1)):
        nova_linha[v1[d]] = v2[d]
    lista.append(nova_linha)

lista_vertices = []
for i in G.nodes:
    lista_vertices.append(i)

dic = []
for j in range(len(lista)):
    dict_from_list = {}
    for k in range(len(G.nodes)):
        dict_from_list[lista_vertices[k]] = lista[j][k]

    dic.append(dict_from_list)
r.append([dic])

val_map = dic[0]
norm = mpl.colors.Normalize(vmin=0, vmax=1, clip=True)
```

```
mapper = mpl.cm.ScalarMappable(norm=norm, cmap=mpl.cm.prism)

fig, axs = plt.subplots(ncols=2, figsize=(10, 6),
           gridspec_kw={'width_ratios': [5, 1]})
my_pos = nx.random_layout(G, seed = 1)
nx.draw(G , nodelist = val_map,
        node_color=[mapper.to_rgba(i) for i in val_map.values()],
        with_labels=True, font_color='white', ax = axs[0], pos = my_pos)
plt.title('Iteration ' +  str(0))
axs[1].axis('off')
plt.show()

for k in range(1, its):
    if dic[k]==dic[k-1]:
        break
    else:
        val_map = dic[k]
        norm = mpl.colors.Normalize(vmin=0, vmax=1, clip=True)
        mapper = mpl.cm.ScalarMappable(norm=norm, cmap=mpl.cm.prism)

        fig, axs = plt.subplots(ncols=2, figsize=(10, 6),
                                gridspec_kw={'width_ratios': [5, 1]})
        my_pos = nx.random_layout(G, seed = 1)
        nx.draw(G , nodelist = val_map,
                node_color=[mapper.to_rgba(i) for i in val_map.values()],
                with_labels=True, font_color='white', ax = axs[0],
                pos = my_pos)
        plt.title('Iteration' +  str(k))
        axs[1].axis('off')
        plt.show()

# Grafphic of the growth
pop = []

for w in range(len(r)):
    dic = r[w][0]
    lista = []
    for i in range(len(dic)):
        na = 0
        ni = 0
```

```
            dic_aux = list(dic[i].values())
            for j in range(len(dic[i])):
                if dic_aux[j] == 1:
                    na = na + 1

            lista.append([na])
        pop.append(lista)

    plt.figure(figsize=(10, 10))
    for i in range(len(pop)):

        colors=['limegreen', 'firebrick', 'deeppink', 'deepskyblue',
                'mediumorchid']

        print(pop[i])

        plt.plot(pop[i], color = colors[i])

        plt.xlabel("Iterations t", fontsize = 12)
        plt.ylabel("Number of active nodes", fontsize = 12)
        plt.legend(["EVC", "DC", "CC", "BC", "EC"], loc =5, fontsize = 10)
        plt.rcParams['xtick.labelsize'] = 15
        plt.rcParams['ytick.labelsize'] = 15

    plt.show()
```

D.2.2 Influence Maximization Algorithms

```
from random import uniform, seed
import time
from igraph import *

import ndlib.models.ModelConfig as mc
import ndlib.models.epidemics as ep
import matplotlib.pyplot as plt
import networkx as nx
import numpy as np
import matplotlib as mpl
import pandas as pd
from operator import itemgetter
```

```
# Algorithms
def IC(g, S, p = 0.5, mc = 1000):
"""
Input: graph, seed set, propagation probability and the number of
        Monte-Carlo simulations
Output: average number of nodes influenced by the seed set
"""

#Loop over the Monte-Carlo simulations
spread = []
for i in range(mc):

    #Simulate propagation process
    new_active, A = S[:], S[:]
    while new_active:

        #For each newly activated node, find its neighbors
        #that become activated
        new_ones = []
        for node in new_active:

            #Determine neigbors that become infected
            np.random.seed(i)
            success = np.random.uniform(0,1,
                                        len(g.neighbors(node, mode = "out"))) < p
            new_ones += list(np.extract(success, g.neighbors(node, mode = "out")))

        new_active = list(set(new_ones) -  set(A))

        #Add newly activated nodes to the set of activated nodes
        A += new_active

    spread.append(len(A))

return(np.mean(spread))


def greedy(g, k, p = 0.1, mc = 1000):
"""
Input: graph, number of seed nodes
```

```
    Output: optimal seed set, resulting propagation and time for each iteration
    """

    S, spread, timelapse, start_time = [], [], [], time.time()

    #Find k nodes with latgest marginal gain
    for _ in range(k):

        #Loop over nodes that are not yet in seed set to find biggest marginal gain
        best_spread = 0
        for j in set(range(g.vcount())) - set(S):

            #Get the spread
            s = IC(g, S + [j], p, mc)

            #Update the winning node and spread so far
            if s > best_spread:
                best_spread, node = s, j

        #Add the selected node to the seed set
        S.append(node)

        #Add estimated spread and elapsed time
        spread.append(best_spread)
        timelapse.append(time.time() - start_time)

    return(S, spread, timelapse)

def celf(g, k, p = 0.1, mc = 1000):
    """
    Input: graph, number of seed nodes
    Output:optimal seed set, resulting propagation, time for each iteration
    """

    #Calculate the first iteration sorted list
    start_time = time.time()
    marg_gain = [IC(g, [node], p, mc) for node in range(g.vcount())]

    #Create the sorted list of nodes and their marginal gain
    Q = sorted(zip(range(g.vcount()), marg_gain), key = lambda x: x[1], reverse = True)
```

```
#Select the first node and remove from cadidate list
S, spread, SPREAD = [Q[0][0]], Q[0][1], [Q[0][1]]
Q, LOOKUPS, timelapse = Q[1:], [g.vcount()], [time.time() - start_time]

#Find the next k-1 nodes using the list-sorting procedure
for _ in range(k-1):
    check, node_lookup = False, 0

    while not check:

        #Count the number of times the spread is computed
        node_lookup +=1

        #Recalculate spread of top node
        current = Q[0][0]

        #Evaluate the spread function and store the marginal gain in the list
        Q[0] = (current, IC(g, S + [current], p, mc) - spread)

        #Re-sort the list
        Q = sorted(Q, key = lambda x: x[1], reverse = True)

        #Check if previous top node stayed on top after the sort
        check = (Q[0][0] == current)

    #Select the next node
    spread += Q[0][1]
    S.append(Q[0][0])
    SPREAD.append(spread)
    LOOKUPS.append(node_lookup)
    timelapse.append(time.time() - start_time)

    #Remove the selected node from the list
    Q = Q[1:]

return(S, SPREAD, timelapse, LOOKUPS)

#Network and model
#Generate graph
```

```
seed(316)
G = Graph.Barabasi(n = 150, m = 2)

G.es["color"], G.vs["color"], G.vs["label"] = "#B3CDE3", "#FBB4AE", ""
plot(G, bbox = (300, 300), margin = 11, layout = G.layout("kk"))

#Finding seed nodes with the algorithms
celf_output = celf(G, 1, p = 0.2, mc = 1000)
greedy_output = greedy(G, 1, p = 0.2, mc = 1000)

print("celf output:   " + str(celf_output[0]))
print("greedy output: " + str(greedy_output[0]))

gr = greedy_output[0]
c = celf_output[0]

print(celf_output[2])
print(greedy_output[2])

# Network and its characteristics
A = G.get_edgelist()
g = nx.Graph(A)
nx.draw(g, pos=nx.random_layout(g, seed=1), with_labels = True,
        font_color='white', node_size=200, node_color='limegreen')

nx.is_weighted(g)

def eccentricity(G, v=None, sp=None, weight=None):
    order = G.order()
    e = {}
    for n in G.nbunch_iter(v):
        if sp is None:
            length = nx.shortest_path_length(G, source=n, weight=weight)

            L = len(length)
        else:
            try:
                length = sp[n]
                L = len(length)
            except TypeError as err:
```

```
                raise nx.NetworkXError('Format of "sp" is invalid.') from err
            if L != order:
                if G.is_directed():
                    msg = (
                        "Found infinite path length because the digraph is not"
                        " strongly connected"
                    )
                else:
                    msg = "Found infinite path length because the graph is not"
                          " connected"
                raise nx.NetworkXError(msg)

            e[n] = max(length.values())

        if v in G:
            return e[v]
        return e

    def diameter(G, e=None, usebounds=False, weight=None):
        if usebounds is True and e is None and not G.is_directed():
            return _extrema_bounding(G, compute="diameter", weight=weight)
        if e is None:
            e = eccentricity(G, weight=weight)
        return max(e.values())

    #Diamter
    print('Diameter ', diameter(g))

    #Average path lenght
    print('Average path length ', nx.average_shortest_path_length(g))

    # Transitivity
    print('Transitivity ', nx.transitivity(g))

    #Maximum and minimum
    graus = []
    for i in g.degree():
        graus.append(i[1])
```

```
print('Maximum degree = ', max(graus))
print('Minimum degree = ', min(graus))

#Node degree histogram
degree_sequence = sorted((d for n, d in g.degree()), reverse=True)
dmax = max(degree_sequence)

fig = plt.figure("Histogram of the nodes degrees", figsize=(10, 10))
axgrid = fig.add_gridspec(5, 4)

ax2 = fig.add_subplot(axgrid[3:, 2:])
ax2.bar(*np.unique(degree_sequence, return_counts=True))
ax2.set_title("Histogram of the nodes degrees")
ax2.set_xlabel("Degree")
ax2.set_ylabel("Frequency")

fig.tight_layout()
plt.show()

# Degree distribution
degree_freq = nx.degree_histogram(g)
degrees = range(len(degree_freq))

degrees_freq_relative = []
for i in range(len(degree_freq)):
    fr = degree_freq[i]/sum(degree_freq)
    degrees_freq_relative.append(fr)

plt.figure(figsize=(6, 4))
plt.loglog(degrees[2:], degrees_freq_relative[2:],'o')
plt.title('Degree distribution of the nodes')
plt.xlabel('Degree')
plt.ylabel('Realtive Frequency')

#Finding seeds with centrality measures
def MostCentralNodes(n_nodes, c_measure):
    list_cm = []

    if c_measure == 'b':
        for i in nx.betweenness_centrality(g).values():
```

```
            list_cm.append(round(i,4))

    elif c_measure == 'c':
        for i in nx.closeness_centrality(g).values():
            list_cm.append(round(i,4))

    elif c_measure == 'd':
        for i in nx.degree_centrality(g).values():
            list_cm.append(round(i,4))

    elif c_measure == 'e':
        for i in nx.eigenvector_centrality(g).values():
            list_cm.append(round(i,4))

    max_cm = []
    centrality_measure = []

    while len(max_cm) < n_nodes:

        max_value = max(list_cm)

        for i in range(len(list_cm)):

            if list_cm[i] == max_value:

                max_cm.append(i)
                centrality_measure.append(list_cm[i])
                list_cm[i] = -1
                break

    if n_nodes == 1:
        print('The most central node according to the chosen centrality
              measure is', str(max_cm),
              '\nwith value =', str(centrality_measure))

    else:
        print('The most central nodes according to the chosen centrality
              measure are', str(max_cm),
              '\nwith values =', str(centrality_measure))
```

```
    return max_cm

dc = MostCentralNodes(1, 'd')
cc = MostCentralNodes(1, 'c')
bc = MostCentralNodes(1, 'b')
ec = MostCentralNodes(1, 'e')

#Propagation model
r = []

def model(central_nodes,G,n):
    if central_nodes == dc:
        print('Centrality measure - degree')
    elif central_nodes == cc:
        print('Centrality measure - closeness')
    elif central_nodes == bc:
        print('Centrality measure - betweenness')
    elif central_nodes == ec:
        print('Centrality measure - eigenvector')
    elif central_nodes == gr:
        print('Greedy')
    elif central_nodes == c:
        print('Celf')

    # Model selection
    model = ep.ThresholdModel(G, seed = 1)

    # Model Configuration
    config = mc.Configuration()
    infected_nodes = central_nodes
    config.add_model_initial_configuration("Infected", central_nodes)

    # Setting node parameters
    threshold = 0.2
    for i in G.nodes():
        config.add_node_configuration("threshold", i, threshold)

    model.set_initial_status(config)
```

```
# Simulation execution
its = n
iterations = model.iteration_bunch(its)

lista = []
linha = []
for j in range(its):
    y = iterations[j]['status']
    for k in range(len(G.nodes)):
        if j == 0 :
            linha.append(y[k])
lista.append(linha)



nova_linha = []
for s in range(1, its):
    y = iterations[s]['status']
    v1 = list(y.keys())
    v2 = list(y.values())
    nova_linha = lista[s-1].copy()
    for d in range(len(v1)):
        nova_linha[v1[d]] = v2[d]
    lista.append(nova_linha)

lista_vertices = []
for i in G.nodes:
    lista_vertices.append(i)

dic = []
for j in range(len(lista)):
    dict_from_list = {}
    for k in range(len(G.nodes)):
        dict_from_list[lista_vertices[k]] = lista[j][k]

    dic.append(dict_from_list)
r.append([dic])

val_map = dic[0]
norm = mpl.colors.Normalize(vmin=0, vmax=1, clip=True)
mapper = mpl.cm.ScalarMappable(norm=norm, cmap=mpl.cm.prism)
```

```
    fig, axs = plt.subplots(ncols=2, figsize=(10, 6),
                            gridspec_kw={'width_ratios': [5, 1]})
    my_pos = nx.random_layout(G, seed = 1)
    nx.draw(G , nodelist = val_map,
            node_color=[mapper.to_rgba(i) for i in val_map.values()],
            with_labels=True, font_color='white', ax = axs[0], pos = my_pos)
    plt.title('Iteration ' +  str(0))
    axs[1].axis('off')
    plt.show()

    for k in range(1, its):
        if dic[k]==dic[k-1]:
            break
        else:
            val_map = dic[k]
            norm = mpl.colors.Normalize(vmin=0, vmax=1, clip=True)
            mapper = mpl.cm.ScalarMappable(norm=norm, cmap=mpl.cm.prism)

            fig, axs = plt.subplots(ncols=2, figsize=(10, 6),
                                    gridspec_kw={'width_ratios': [5, 1]})
            my_pos = nx.random_layout(G, seed = 1)
            nx.draw(G , nodelist = val_map,
                    node_color=[mapper.to_rgba(i) for i in val_map.values()],
                    with_labels=True, font_color='white', ax = axs[0],
                    pos = my_pos)
            plt.title('Iteração ' +  str(k))
            axs[1].axis('off')
    return plt.show()

model_greedy = model(gr,g,20)
model_dc = model(dc,g,20)
model_cc = model(cc,g,20)
```

Bibliografia

(2005). Womma: Word of mouth 101: An introduction to word of mouth marketing. Tech. rep., Word of Mouth Marketing Association, 333 W. North Avenue, #500, Chicago, IL 60610.

Ahmed, S. and Ezeife, C. I. (2013). Discovering influential nodes from trust network. In *Proceedings of the 28th Annual ACM Symposium on Applied Computing*, SAC '13, page 121–128. Association for Computing Machinery.

Aiello, W., Chung, F., and Lu, L. (2000). A random graph model for massive graphs. In *Proceedings of the Thirty-Second Annual ACM Symposium on Theory of Computing*, STOC '00, page 171–180. Association for Computing Machinery.

Allen, L. J. S. (2008). An Introduction to Stochastic Epidemic Models. In Brauer, F., Driessche, P., and Wu, J., editors, *Mathematical Epidemiology*, Lecture Notes in Mathematics, page 81–130. Springer, Berlin, Heidelberg.

Anderson, R. M. and May, R. M. (1979). Population biology of infectious diseases: Part I. *Nature*, 280:361–367.

Aregui, A. and Denoeux, T. (2007). Consonant Belief Function Induced by a Confidence Set of Pignistic Probabilities. In Mellouli, K., editor, *Symbolic and Quantitative Approaches to Reasoning with Uncertainty*, volume 4724 of *ECSQARU 2007*, Berlin, Heidelberg. Springer.

Aregui, A. and Denoeux, T. (2008). Constructing consonant belief functions from sample data using confidence sets of pignistic probabilities. *Int. J. Approx. Reason.*, 49:575–594.

Argaiz, J. L. I. (2015). *The Dynamic of Viral, The Dynamics of Viral Information Diffusion in Online Social Networks*. PhD thesis, Universidad Carlos III, Madrid.

Arnold, B. C. (2008). Pareto and Generalized Pareto Distributions. In Chotikapanich, D., editor, *Modeling Income Distributions and Lorenz Curves. Economic Studies in Equality, Social Exclusion and Well-Being*, volume 5. Springer, New York, NY.

Barabási, A. L. (2005). The origin of bursts and heavy tails in human dynamics. *Nature*, 435:207–211.

Barabási, A.-L. (2016). *Network Science*. Cambridge University Press.

Barabási, A. L. and Albert, R. (1999). Emergence of Scaling in Random Networks. *Science*, 286:509–512.

Barrat, A., Barthelemy, M., and Vespignani, A. (2008). *Dynamical Processes on Complex Networks*. Cambridge University Press.

Berry, J. and Keller, E. (2003). *The Influentials: One American In Ten Tells The Other Nine How To Vote, Where To Eat, And What To Buy*. Free Press.

Boase, J. and Wellman, B. (2001). A Plague of Viruses: Biological, Computer and Marketing. *Curr. Sociol.*, 49(6):39–55.

Bollobás, B. (1985). *Random Graphs*. Academic Press.

Bollobás, B. (2001). The Evolution of Random Graphs, The Giant Component. In *Random Graphs*, Cambridge Studies in Advanced Mathematics, pages 130–159. Cambridge University Press.

Bollobás, B. and Riordan, O. (2002). A polynomial of graphs on surfaces. *Math Ann*, 323:81–96.

Bond, M. and Harrigan, N. (2011). *The SAGE Handbook of Social Network Analysis*. Sage Publications Ltd.

Bozorgi, A., Haghighi, H., Zahedi, M. S., and Rezvani, M. (2016). INCIM: A community-based algorithm for influence maximization problem under the linear threshold model. *Inf. Process. Manag.*, 52:1188–1199.

Bruning, P. F., Alge, B. J., and Lin, H. C. (2020). Social networks and social media: Understanding and managing influence vulnerability in a connected society. *Bus. Horiz.*, 63:749–761.

Bruyn, A. D. and Lillien, G. L. (2008). A multi-stage model of word-of-mouth through viral marketing. *Int. J. Res. Mark.*, 25(3):151–163.

Castellano, C., Fortunato, S., and Loreto, V. (2009). Statistical physics of social dynamics. *Rev. Mod. Phys.*, 81:591–646.

Chakrabarti, D., Wang, Y., Wang, C., Leskovec, J., and Faloutsos, C. (2008). Epidemic thresholds in real networks. *ACM Trans. Inf. Syst. Secur.*, 10(4):1–26.

Chen, W., Wang, Y., and Yang, S. (2009). Efficient influence maximization in social networks. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '09, page 199–208. Association for Computing Machinery.

Chen, W., Yuan, Y., and Zhang, L. (2010). Scalable influence maximization in social networks under the linear threshold model. In *2010 IEEE International Conference on Data Mining*, pages 88–97.

Chen, Y., Peng, W. C., and Lee, S. Y. (2012). Efficient algorithms for influence maximization in social networks. *Knowl Inf Syst*, 33:577–601.

Choudhury, M. D., Lin, Y. R., Sundaram, H., Candan, K. S., Xie, L., and Kelliher, A. (2010). How Does the Data Sampling Strategy Impact the Discovery of Information Discusion in Social Media? In *Proceedings of the Fourth International Conference on Weblogs and Social Media, ICWSM 2010*, pages 34–41, Washington, DC, USA.

Chung, F. and Lu, L. (2002). Connected Components in Random Graphs with Given Expected Degree Sequences. *Ann. Comb.*, 6:125–145.

Clauset, A., Shalizi, C. R., and Newman, M. E. (2009). Power-Law Distributions in Empirical Data. *SIAM Review*, 51(4):661–703.

Cohen, R. and Havlin, S. (2003). Scale-Free Networks Are Ultrasmall. *Phys. Rev. Lett.*, 90:058701.

Cohen, R., Havlin, S., and ben Avraham, D. (2002). Structural properties of scale-free networks. In Bornholdt, S. and Schuster, H. G., editors, *Handbook of Graphs and Networks: From the Genome to the Internet*. Wiley-VCH.

Daley, D. J. and Gani, J. (1999). *Epidemic Modelling: An Introduction*. Cambridge University Press.

Delamater, P. L., Street, E. J., Leslie, T. F., Yang, Y. T., and Jacobsen, K. H. (2019). Complexity of the Basic Reproduction Number (R_0). *Emerg Infect Dis.*, 25(1):1–4.

Dellarocas, C. and Narayan, R. (2006). A Statistical Measure of a Population's Propensity to Engage in Post-purchase Online Word-of-Mouth. *Stat. Sci.*, 21(2):227–285.

Dempster, A. P. (1967). Upper and Lower Probabilities Induced by a Multivalued Mapping. *Ann. Math. Statist.*, 38(2):325–339.

Devroye, L. (1986). *Non-Uniform Random Variate Generation*. Springer New York, NY.

Diekmann, O. and Heesterbeek, J. A. P. (2000). *Mathematical Epidemiology of Infectious Diseases: Model Building, Analysis, and Interpretation*. John Wiley & Sons.

Dijkstra, E. (1959). A note on two problems in connexion with graphs. *Numer. Math.*, 1:269–271.

Domingos, P. and Richardson, M. (2001). Mining the network value of customers. In *Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '01, pages 57–66.

Dorogovtsev, S. and Mendes, J. F. F. (2002). Evolution of networks. *Adv. Phys.*, 51:1079–1187.

Dorogovtsev, S. N., Goltsev, A. V., and Mendes, J. F. F. (2002). Pseudofractal scale-free web. *Phys. Rev. E*, 65:066122.

Dorogovtsev, S. N. and Mendes, J. (2003). *Evolution of Networks: From Biological Nets to the Internet and WWW*. Oxford University Press.

Dubois, D. and Prade, H. (1988). Representation and combination of uncertainty with belief functions and possibility measures. *Comput. Intell.*, 4:244–264.

Dye, R. (2000). The buzz on buzz. *Harv. Bus. Rev.*, 78(6):139–146.

Ebel, H., Mielsch, L. I., and Bornholdt, S. (2002). Scale-free topology of e-mail networks. *Phys. Rev. E*, 66:035103.

Eckler, P. and Rodgers, S. (2010). Viral Marketing on the Internet. *Wiley International Encyclopedia of Marketing*.

Erdös, P. and Rényi, A. (1959). On Random Graphs. *Publ. Math. Debrecen*, 6:290–297.

Erdös, P. and Rényi, A. (1960). On the Evolution of Random Graphs. *Publ. Math. Inst. Hungar. Acad. Sci.*, 5:17–61.

Godin, S. (2001). *Unleashing the Ideavirus*. Hyperion Books.

Goldenberg, J., Libai, B., and Muller, E. (2001). Talk of the Network: A Complex Systems Look at the Underlying Process of Word-of-Mouth. *Mark. Lett.*, 12(3):211–223.

Gomez-Rodriguez, M., Balduzzi, D., and Schölkopf, B. (2011). Uncovering the Temporal Dynamics of Diffusion Networks. In *28th International Conference on Machine Learning (ICML)*, ICML'11, pages 561–568.

Goyal, A., Bonchi, F., and Lakshmanan, L. V. S. (2012). A data-based approach to social influence maximization. *Proc. VLDB Endow.*, 5(1):73–84.

Granovetter, M. (1978). Threshold Models of Collective Behavior. *Am. J. Sociol.*, 83(6):1420–1443.

Gruhl, D., Guha, R., Liben-Nowell, D., and Tomkins, A. (2004). Information diffusion through blogspace. In *Proceedings of the 13th International Conference on World Wide Web*, WWW '04, page 491–501. Association for Computing Machinery.

Guille, A., Hacid, H., Favre, C., and Zighed, D. A. (2013). Information diffusion in online social networks: a survey. *SIGMOD Rec.*, 42(2):17–28.

Hafiene, N., Karoui, W., and Romdhane, L. B. (2019). nfluential Nodes Detection in Dynamic Social Networks. In Abramowicz, W. and Corchuelo, R., editors, *Business Information Systems. BIS 2019*, volume 354 of *Lecture Notes in Business Information Processing*. Springer, Cham.

Hao, F., Zhu, C., Chen, M., Yang, L., and Pei, Z. (2011). Influence Strength Aware Diffusion Models for Dynamic Influence Maximization in Social Networks. In *2011 International Conference on Internet of Things and 4th International Conference on Cyber, Physical and Social Computing*, pages 317–322.

Harris, T. E. (1963). *The theory of branching processes*, volume 6. Springer, Berlin.

Heesterbeek, J. A. P. (2002). A Brief History of R_0 and a Recipe for its Calculation. *Acta Biotheor.*, 50:189–204.

Hofstad, R. (2016). *Random Graphs and Complex Networks*, volume 43 of *Cambridge Series in Statistical and Probabilistic Mathematics*. Cambridge University Press.

Holme, P. and Kim, B. J. (2002). Growing scale-free networks with tunable clustering. *Phys. Rev. E*, 65:026107.

Ibarra, H. and Andrews, S. B. (1993). Power, Social Influence, and Sense Making: Effects of Network Centrality and Proximity on Employee Perceptions. *Adm. Sci. Q*, 38(2):277–303.

Jendoubi, S. (2016). *Influencers Characterization in a Social Network for Viral Marketing Perspectives*. PhD thesis, Université de Rennes, France.

Johnson, N. L., Kemp, A. W., and Kotz, S. (1992). *Univariate Discrete Distributions*. John Wiley & Sons.

Jurvetson, S. (2000). What exactly is viral marketing? *Red Herring*, 78:110–112.

Kempe, D., Kleinberg, J., and Tardos, E. (2003). Maximizing the spread of influence through a social network. In *Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '03, pages 137–146.

Kempe, D., Kleinberg, J., and Tardos, E. (2015). Maximizing the Spread of Influence through a Social Network. *Theory Comput.*, 11(4):105–147.

Kermack, W. O. and McKendrick, A. G. (1927). A contribution to the mathematical theory of epidemics. *Proc. R. Soc. Lond. A*, 115:700–721.

Kimura, M. and Saito, K. (2006). Tractable Models for Information Diffusion in Social Networks. In Fürnkranz, J., Scheffer, T., and Spiliopoulou, M., editors, *Knowledge Discovery in Databases: PKDD 2006*, volume 4213 of *PKDD 2006. Lecture Notes in Computer Science*, page 259–271. Springer, Berlin, Heidelberg.

Kiss, C. and Bichler, M. (2008). Identification of influencers — Measuring influence in customer networks. *Decis. Support Syst.*, 46:233–253.

Kiss, I. Z., Miller, J. C., and Simon, P. L. (2017). *Mathematics of Epidemics on Networks, From Exact to Approximate Models*. Springer Cham.

Kitsak, M., Gallos, L. K., Havlin, S., Liljeros, F., Muchnik, L., Stanley, H. E., and Makse, H. A. (2010). Identification of influential spreaders in complex networks. *Nat. Phys*, 6:888–893.

Kumar, V., Petersen, J. A., and Leone, R. P. (2007). How valuable is word of mouth? *Harv. Bus. Rev.*, 85(10):139–166.

Lerman, K. and Galstyan, A. (2008). Analysis of social voting patterns on digg. In *Proceedings of the First Workshop on Online Social Networks*, WOSN '08, pages 7–12. ACM.

Leskovec, J., Adamic, L. A., and Huberman, B. A. (2007a). The dynamics of viral marketing. *ACM Trans. Web*, 1(1).

Leskovec, J., Krause, A., Guestrin, C., Faloutsos, C., VanBriesen, J., and Glance, N. (2007b). Cost-effective outbreak detection in networks. In *Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '07, page 420–429. Association for Computing Machinery.

Li, W., Zhong, K., Wang, J., and Chen, D. (2021). A dynamic algorithm based on cohesive entropy for influence maximization in social networks. *Expert Syst. Appl.*, 169:114207.

Li, Y., Fan, J., Wang, Y., and Tan, K. (2018). Influence Maximization on Social Graphs: A Survey. *IEEE Trans. Knowl. Data Eng.*, 30(10):1852–1872.

Macdonald, B., Shakarian, P., Howard, N., and Moores, G. (2012). Spreaders in the Network SIR Model: An Empirical Study. *arXiv*.

Martin, A. and Osswald, C. (2006). Human expert fusion for image classification. *Int. J. Inf. Secur.*, 20:122–143.

Martin, A. and Osswald, C. (2007). Toward a combination rule to deal with partial conflict and specificity in belief functions theory. In *2007 10th International Conference on Information Fusion*, pages 1–8.

Menczer, F., Fortunato, S., and Davis, C. A. (2020). *A First Course in Network Science*. Cambridge University Press.

Modzelewski, M. F. (2000). Finding a Cure for Viral Marketing Ills. *DM News*, 13:1.

Montgomery, A. L. (2001). Applying Quantitative Marketing Techniques to the Internet. *Interfaces*, 31(2):90–108.

Myers, S. A., Zhu, C., and Leskovec, J. (2012). Information diffusion and external influence in networks. In *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '12, page 33–41. Association for Computing Machinery.

Nemhauser, G. L., Wolsey, L. A., and Fisher, M. L. (1978). An analysis of approximations for maximizing submodular set functions — I. *Math. Program.*, 14:265–294.

Newman, M. (2010). *Networks: An Introduction*. Oxford University Press.

Newman, M. E. J. (2001). The structure of scientific collaboration networks. *Proc. Natl. Acad. Sci. U.S.A*, 98(2):404–409.

Okabe, Y. and Shudo, A. (2021). Microscopic Numerical Simulations of Epidemic Models on Networks. *Mathematics*, 9(9):932.

Opsahl, T., Agneessens, F., and Skvoretz, J. (2010). Node centrality in weighted networks: Generalizing degree and shortest paths. *Soc. Netw.*, 32:245–251.

Pastor-Satorras, R., Castellano, C., Mieghem, P. V., and Vespignani, A. (2015). Epidemic processes in complex networks. *Rev. Mod. Phys.*, 87:925–979.

Phelps, J., Lewis, R., Mobilio, L., Perry, D., and Raman, N. (2004). Viral Marketing or Electronic Word-of-Mouth Advertising: Examining Consumer Responses and Motivations to Pass Along Email. *J. Advert. Res.*, 44(4):333–348.

Pitkow, J. (1997). Summary of WWW characterizations. In *Proceedings of the Seventh World Wide Web Conference (WWW7)*.

Raftery, A. E. (1994). Pareto Distributions. *Stat. Sci.*, 9(4):410–424.

Reichstein, T. and Brusch, I. (2019). The decision-making process in viral marketing — A review and suggestions for further research. *Psychol. Mark.*, 36:1062–1081.

Rocha, J. L. and Caneco, A. (2013). Mutual information rate and topological order in networks. *Int. J. Nonlinear Sci*, 4:553–562.

Rocha, J. L. and Carvalho, S. (2021). Information transmission and synchronizability in complete networks of systems with linear dynamics. *Math. Comput. Simul.*, 182:340–352.

Rocha, J. L. and Carvalho, S. (2023). Complete dynamical networks: Synchronization, information transmission and topological order. *J. Discontinuity Nonlinearity Complex.*, 12:99–109.

Rocha, J. L., Carvalho, S., and Coimbra, B. (2023a). Probabilistic Procedures for SIR and SIS Epidemic Dynamics on Erdös-Rényi Contact Networks. *AppliedMath*, 3(4):828–850.

Rocha, J. L., Carvalho, S., and Coimbra, B. (2023b). SIR and SIS Epidemic Dynamics in Random Contact Networks. In *CHAOS*.

Rocha, J. L., Carvalho, S., Coimbra, B., Henriques, I., and Pereira, J. (2023c). Influence Maximization Dynamics and Topological Order on Erdös-Rényi Networks. *Mathematics*, 11(15):3299.

Rocha, J. L., Grácio, C., Fernandes, S., and Caneco, A. (2015). Spectral and dynamical invariants in a complete clustered network. *Appl. Math. Inf. Sci.*, 9:2367–2376.

Rodriguez, M. G., Leskovec, J., and Krause, A. (2010). Inferring networks of diffusion and influence. In *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '10, page 1019–1028.

Roelens, I., Baecke, P., and Benoit, D. F. (2016). Identifying influencers in a social network: The value of real referral data. *Decis. Support Syst.*, 91:25–36.

Rosen, E. (2000). *The Anatomy of Buzz: How to Create Word-of-Mouth Marketing*. Doubleday Business.

Ross, R. and Hudson, H. (1917). An application of the theory of probabilities to the study of a priori pathometry — Part III. *Proc. R. Soc. Lond. A*, 89:225–240.

Schmitt, P., Skiera, B., and den Bulte, C. V. (2011). Referral Programs and Customer Value. *J. Mark.*, 75(1):46–59.

Shafer, G. (1976). *A Mathematical Theory of Evidence*. Princeton University Press.

Shakarian, P., Bhatnagar, A., Aleali, A., Shaabani, E., and Guo, R. (2015a). The Independent Cascade and Linear Threshold Models. In *Diffusion in Social Networks*, SpringerBriefs in Computer Science, pages 35–48. Springer Cam.

Shakarian, P., Bhatnagar, A., Aleali, A., Shaabani, E., and Guo, R. (2015b). The SIR Model and Identification of Spreaders. In *Diffusion in Social Networks*, SpringerBriefs in Computer Science, page 3–18. Springer Cam.

Smets, P. (1990). The combination of evidence in the transferable belief model. *IEEE Trans. Pattern Anal. Mach. Intell.*, 12(5):447–458.

Smets, P. (2005). Decision making in the TBM: the necessity of the pignistic transformation. *Int. J. Approx. Reason.*, 38:133–147.

Stirzaker, D. (1999). *Probability and Random Variables: A Beginner's Guide*. Cambridge University Press.

Subramani, M. and Rajagopalan, B. (2003). Knowledge-sharing and influence in online social networks via viral marketing. *Commun. ACM*, 46(12):300–307.

Talukder, A., Alam, M. G. R., Tran, N. H., Niyato, D., Park, G. H., and Hong, C. S. (2019). Threshold Estimation Models for Linear Threshold Based Influential User Mining in Social Networks. *IEEE Access*, 7:105441–105461.

Teng, Y., Shi, Y., Tai, C., Yang, D., Lee, W., and Chen, M. (2021). Influence Maximization Based on Dynamic Personal Perception in Knowledge Graph. In *2021 IEEE 37th International Conference on Data Engineering (ICDE)*, pages 1488–1499.

Wang, Y., Wang, H., Li, J., and Gao, H. (2016). Efficient Influence Maximization in Weighted Independent Cascade model. In Navathe, S., Wu, W., Shekhar, S., Du, X., Wang, S., and Xiong, H., editors, *Database Systems for Advanced Applications*, volume 9643 of *DASFAA 2016. Lecture Notes in Computer Science*. Springer, Cham.

Watts, D. J. and Dodds, P. S. (2007). Influentials, networks and public opinion formation. *J. Consum. Res.*, 34:441–458.

Wei, D., Deng, X., Zhang, X., Deng, Y., and Mahadevan, S. (2013). Identifying influential nodes in weighted networks based on evidence theory. *Phys. A: Stat. Mech. Appl.*, 392:2564–2575.

Williams, D. (1991). *Probability with Martingales*. Cambridge University Press.

Yager, R. R. (1987). On the Dempster-Shafer framework and new combination rules. *Inf. Sci.*, 41(2):93–137.

Zhang, L. and Li, K. (2022). Influence Maximization Based on Snapshot Prediction in Dynamic Online Social Networks. *Mathematics*, 10(8):1341.