Politecnico di Milano

SCHOOL OF INDUSTRIAL AND INFORMATION ENGINEERING

Master of Science – Automation and Control Engineering

Opinion Manipulation in Social Networks

Manipolazione delle Opinioni nelle Reti Sociali

Supervisor

Prof. Alessandro Colombo

Co-Supervisors

Prof. Paolo Giuseppe Emilio Bolzern

Prof. Carlo Piccardi

Candidate

Luca Invernizzi – 913979

Academic Year 2019 – 2020

Sommario

Questa tesi si propone di indagare gli effetti di diverse strategie di manipolazione dell'opinione applicate a reti sociali sintetiche governate da un modello Markoviano multi-agente, i cui tassi di transizione individuali dipendono dalle specifiche opinioni assunte dai vicini di ciascun agente. Verranno mostrati i risultati di simulazioni Monte Carlo del sistema, al fine di esaminare le conseguenze dovute alle variazioni dei principali parametri del modello e di differenti strategie di attacco sull'opinione media della rete a regime. Verrà dimostrato che alcune strategie di attacco ottengono consistentemente risultati migliori rispetto alle altre, sebbene nessuna delle euristiche trattate per la selezione degli agenti più influenti è in grado di ottenere il risultato ottimale in ogni topologia di rete. Particolare attenzione verrà dedicata agli effetti introdotti sull'attacco dalla presenza nella rete di una struttura di comunità, dimostrando che perseguire una semplice strategia come concentrare l'attacco su una singola comunità non sempre costituisce il piano d'azione più efficace per manipolare l'opinione media di quella comunità. Infine, verrà analizzato come il gestore di un social network come Facebook o Twitter possa ripristinare un'opinione media neutrale attraverso l'esercizio di un'azione di filtraggio dei post condivisi tra gli utenti, e come valutare la gravità di un attacco dall'entità dello sforzo di filtraggio richiesto. Si vedrà che, agendo sulle "manopole di controllo centralizzate" utilizzate per esercitare l'azione di filtraggio, si intensifica la polarizzazione delle opinioni tra le diverse comunità.

Abstract

This thesis aims at investigating the effects of different opinion manipulation strategies applied to synthetic social networks. The opinion of each individual in the social network is modelled as a Markovian model, whose transition rates are dependent on the opinions of the individual's neighbours. Monte Carlo simulations are carried out in order to inspect the consequences induced by variations of the main model parameters and different attack strategies on the steady-state average opinion of the network. It will be proved that some strategies consistently achieve better results compared to the others, although none of the investigated heuristics for selecting the most influential agents is capable of attaining the optimal result for every network topology. Some attention is dedicated to the effects introduced on the opinion manipulation attack by the presence of a community structure in the network, showing that not always a straightforward strategy like concentrating the attack effort on a single community is the most effective plan of action for manipulating the average opinion of that community. Lastly, it is analysed how the platform manager of an online social network like Facebook or Twitter could restore a neutral average opinion in the network by exerting a content-dependent filtering action on the posts shared between users, and how to assess the severity of an opinion manipulation attack from the magnitude of the required filtering effort. It will be showed that acting on the "centralized control knobs" used to exert the filtering action intensifies the opinion polarization between different communities.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

With the advent of online social networks, everyone obtained the capability of reaching a very broad audience with just a few clicks. People could easily interact with long-lost friends and join international groups based on shared interests. Communications shifted from a one-to-one paradigm to a one-to-many paradigm, and the mechanisms for opinion formation and news propagation had to adapt accordingly. However, the sudden increase of connectedness between people all around the world came with some drawbacks: malicious actors now have powerful tools in their hands for targeting more susceptible people and manipulating the opinion dynamics of social networks, with the intent of pursuing either social, economical, or political interests.

In the last years, the most emblematic example of opinion manipulation was given by the Russian meddling in the U.S. 2016 elections. According to the *Report on the Investigation into Russian Interference in the 2016 Presidential Election*, Russia hindered the democratic process by making use of the *Internet Research Agency*, a "troll farm", i.e. a collection of bots and fake accounts operated by Russian operatives, with the objective of pushing perspectives and news stories that favored the Russian government, swayed political conversations, spread disinformation, and amplified political and social discord. Very large online social networks like *Facebook*, *Twitter* and *Reddit* were targeted by the Internet Research Agency in order to polarize a very broad audience. Twitter declared in 2018 [1] that it found over 3 800 accounts linked to Russian operatives in the investigations

following the 2016 elections. For comparison, the estimated number of Twitter users in the United States in 2016 was about 50 million. Notable examples are the now-terminated accounts *@ten_gop*, *@jenn_abrams* and *@pamela_moore13*, with 145 244, 70 926 and 70 732 followers at the time of their last tweet, respectively [2]. According to J. Albright [3], research director of the *Tow Center for Digital Journalism* at Columbia University, as few as six Facebook pages created by Russians were responsible for about 340 million "shares" and 19 million "interactions". In Albright's terminology, "shares" represent the *potential* number of people who may have seen the posts created by those pages, and it was estimated based on the number of followers of the accounts. The "interactions" were calculated as the sum of reactions, comments and shares of the posts published by the pages. A single anti-immigration post written by one of those pages was able to collect as high as 797 091 "interactions". The number disclosed by Facebook [4] as their estimation for the number of potential readers of posts generated by Russian operatives is not much lower than the one provided by Albright, at 126 million people who may have seen the posts during the 2016 U.S. elections. The political scientist K. H. Jamieson believe that it is "highly probable" that the combined effect of the Russian interferences was able to turn the outcome of the elections [5].

The objective of this thesis is to simulate opinion manipulation attacks on synthetic social networks governed by a suitable opinion dynamics stochastic model, in order to investigate the effects on the steady-state average opinion of the network caused by variations of the main model parameters and attack strategies. Then, it will be analysed how the network manager could restore a neutral average opinion by making use of content-filtering algorithms, and how to assess the severity of an opinion manipulation attack from the magnitude of the required filtering action.

1.1 Main contributions

The main contributions of this thesis are:

- The study of how the model parameters and the choice of the controlled agents influence the effects of an opinion manipulation attack;
- The development of a closed-loop control approach for assessing the severity

of an opinion manipulation attack, through the estimation of the magnitude of the control variable required for restoring the steady-state average opinion of the network;

- The study of how the presence of a community structure influences the effects of both the opinion manipulation attack and the rebalancing action exerted by the platform manager;

- The development of an efficient algorithm for calculating the contingency tables of the agents' opinions.

1.2 Document structure

The document is structured as follows:

- *Chapter 2* introduces the multi-agent Markovian model which will be used throughout this thesis as the model governing the opinion dynamics of the individuals belonging to the social network, describing its main characteristics and related results;

- *Chapter 3* covers the algorithm employed for the numerical simulations of the multi-agent Markovian model and the calculation of the contingency tables of the agents' opinions;

- *Chapter 4* describes the main topological features of social networks, presenting the two models for generating synthetic social networks which will be used in the simulations for representing the interactions between agents. Moreover, the chapter describes the algorithm used in this thesis for community identification, and the centrality indices employed in the selection of the most influential agents in a given network;

- *Chapter 5* covers opinion manipulation attacks, describing possible attack strategies, the parameters involved and the main effects of changing them, and some implications given by the presence of a community structure;

- *Chapter 6* describes a closed-loop control approach used for assessing the severity of an opinion manipulation attack and for restoring the average opinion of a network during the attack, while presenting some drawbacks in the rebalancing action implied by the presence of a community structure in the network;
- *Chapter 7* concludes the document summarizing the main results of the thesis and describing possible future developments.

Chapter 2

Models of Opinion Dynamics

This chapter describes the model governing the opinion dynamics of the individuals belonging to a social network. First, a very basic overview on models of opinion dynamics is presented, and then sections 2.2 and 2.3 elaborate further on the multi-agent Markovian model developed in [6, 7, 8], describing its main characteristics and related results.

2.1 Overview on models of opinion dynamics

In recent years, many dynamic models of opinion formation in social networks have been developed. According to [9], these models can be divided in two major classes: *macroscopic* models, which describe how the distribution of opinions in a network evolves over time, and *microscopic* models, which instead describe how the opinions of individual actors evolve. In this framework, each actor is represented as an *agent* and the social connections between actors are modelled by a network. An important classification in multi-agent models is whether the single opinion is represented as a real value (typically in the interval $[0, 1]$), or if it is described by a logical variable taking values in a discrete set.

One of the first models of opinion formation with real-valued opinions was developed by French [10] and then extended by DeGroot [11] in the so-called *iterative opinion pooling* model. This discrete-time model described a simple mechanism where each individual's opinion is influenced by the opinions of the neighbours, and

it demonstrated the capability of manifesting the social phenomenon of consensus. One important result of French's work is that it revealed a profound relationship between opinion formation and centrality measures on the underlying network. In [12], Abelson proposed a continuous-time counterpart of the French-DeGroot model, which was extended by Taylor [13] introducing *communication sources*, providing static opinions influencing agents' opinions, and *prejudices*, which are opinions formed by some external factors and which agents converge to in absence of interpersonal influences. A discrete-time counterpart of the Taylor model is the Friedkin-Johnsen model introduced in [14].

More recent works extended these models along various directions, including the introduction of gossip-based interactions, which remove the assumption of synchronicity in the change of opinions among the agents; bounded-confidence models accounting for homophily, i.e. the idea that similar individuals interact more often than dissimilar people; disagreement via cognitive dissonance, e.g. the *boomerang effect*, which states that an attempt to persuade a person sometimes shifts his/her opinion away from the persuader's opinion. More details about these models can be found in [15].

Real-valued deterministic models have two important limits: although well-suited to treat binary opinions, it is difficult to apply these models to multi-dimensional opinion spaces, and the assumption of a deterministic opinion dynamics may be an overly simplified description of real phenomena.

In the class of discrete opinion models, Markov chains are often adopted in order to describe the time evolution of opinions in a stochastic framework. Model based on Markov chains are very appealing in view of their flexibility, but they become intractable for modelling social networks as soon as the number of agents grows. [16] and [17] are examples of discrete-time Markovian models where specific assumptions on the interaction mechanism have been employed in order to overcome the intractability problem. In the former, the interaction occurs only between agents sharing the same opinion, while in the latter, at each time step each agent is affected just by a single individual randomly extracted from the set of his/her neighbours.

Bolzern, Colaneri and De Nicolao proposed in [6] a continuous-time multi-agent Markovian model, providing an exact analysis of the stochastic model without re-

sorting to a mean field approximation. According to their model, the opinion of each agent is modelled as a random variable taking values in a finite set, evolving as a Markov process with suitable transition rates affected by the neighbours' opinions. For a given agent, the probability of moving to a certain opinion increases proportionally to the number of the agent's neighbours who share that particular opinion. The model was then extended in [7] with the introduction of the concepts of *influenceability*, *trustiness* and *stochastic social power*, and in [8] with the analysis of the propagation of joint probabilities. The multi-agent Markovian model developed by Bolzern, Colaneri and De Nicolao is the model upon which this thesis is based, and it will be described in the next section.

2.2 Multi-agent Markovian model

A Markov chain is a stochastic process which describes a sequence of possible events where the probability of each event depends only on the current state of the system and not on its past states. This property is called *Markov property*.

In the multi-agent Markovian model, the opinion dynamics is described by a *continuous-time finite-state Markov chain*, where the transition rates of each agent are dependent on the opinion of his/her neighbours. The transition rates are the sum of two terms, which in this thesis are based on the following two assumptions:

1. The dynamics of the agents' opinion is influenced by the interactions with other agents due to a single social network, e.g. Facebook. In particular, one term of the transition rates accounts for the influence of an agent's neighbours and is proportional to the fraction of his/her neighbours that share a particular opinion;

2. Influences due to any source exogenous to the network under scrutiny (e.g. individual prejudice, information media, other social networks, etc.) can all be lumped into a single constant term.

The model

A matrix A is *Metzler* if its off-diagonal elements are non-negative, i.e. $a_{ij} \geq 0$, $\forall i \neq j$. Moreover, a $N \times N$ Metzler matrix A is *reducible* if there exists a permutation matrix P such that:

$$P'AP = \begin{bmatrix} A_{11} & A_{12} \\ 0 & A_{22} \end{bmatrix},$$

where A_{11} is a $k \times k$ matrix, $1 \leq k \leq N-1$. A Metzler matrix that is not reducible is called *irreducible*.

The interaction between the agents is described through a weighted graph $\mathcal{G} = (\mathcal{N}, \mathcal{E}, W)$, where $\mathcal{N} = \{1, ..., N\}$ is the set of nodes representing the agents, $\mathcal{E} \subseteq \mathcal{N} \times \mathcal{N}$ is the set of edges corresponding to reciprocal influences, and $W = [w_{rs}] \in \mathbb{R}^{N \times N}$ represents the interpersonal "trustiness", i.e. how much credit agent r gives to agent s and his/her opinion. In the sequel, it will be assumed that the graph $\mathcal{G}$ is connected. In the so-called *standard* model, the trustiness is uniform:

$$w_{rs} = \begin{cases} |\mathcal{N}^{[r]}|^{-1}, & (r,s) \in \mathcal{E} \\ 0, & \text{otherwise} \end{cases}$$

where $\mathcal{N}^{[r]} = \{s \in \mathcal{N} : (r,s) \in \mathcal{E}\}$ is the set of the neighbours of agent r, and $|\mathcal{N}^{[r]}|$ denotes its cardinality.

In the *degree-weighted normalized* model, the trustiness is dependent on the neighbours' popularity, quantified through their degree, and is given by:

$$w_{rs} = \begin{cases} |\mathcal{N}^{[s]}| \ / \ \sum_{i \in \mathcal{N}^{[r]}} |\mathcal{N}^{[i]}|, & (r,s) \in \mathcal{E} \\ 0, & \text{otherwise} \end{cases}$$

The system state evolves according to a *finite-state continuous-time Markov chain* with transition rate matrix:

$$\tilde{Q}^{[r]}(t) = Q^{[r]} + A^{[r]}(t). \tag{2.1}$$

$Q^{[r]} \in \mathbb{R}^{M \times M}$ is the transition rate matrix of agent r when isolated, modelling

all the sources of influence on the agent's opinion that are not attributed to the interaction with his/her neighbours in the network under scrutiny, according to the second assumption listed at the beginning of section 2.2. The elements $q_{ij}^{[r]} \geq 0$, $i \neq j$, are defined as:

$$\Pr\left(\sigma^{[r]}(t+dt) = j \mid \sigma^{[r]}(t) = i\right) = q_{ij}^{[r]}dt + o(dt), \quad i \neq j,$$

where $\sigma^{[r]}(t) \in \mathcal{M} = \{1, 2, ..., M\}$ is the opinion (i.e. state) of agent r at time t, while the symbol $\Pr(\mathcal{A}|\mathcal{B})$ denotes the conditional probability of the random event $\mathcal{A}$ given the event $\mathcal{B}$. The diagonal elements of $Q^{[r]}$ are defined as $q_{ii}^{[r]} = -\sum_{j=1, j\neq i}^{M} q_{ij}^{[r]}$, so that $Q^{[r]}$ is a Metzler matrix satisfying the equation $Q^{[r]}\mathbf{1}_M = 0$. The symbol $\mathbf{1}_M$ denotes the M-dimensional column vector with all elements equal to 1. Throughout the rest of the thesis, it will be assumed that $q_{ij}^{[r]} > 0$, $\forall i \neq j$, in order to guarantee irreducibility of $Q^{[r]}$, which is a necessary condition for the ergodicity of the Markov process.

In the case of *binary opinion*, i.e. $M = 2$, it is convenient to use the so-called (α, β)-parametrization introduced in [7], which defines:

$$\alpha^{[r]} = q_{12}^{[r]} + q_{21}^{[r]} \qquad \beta^{[r]} = \frac{q_{21}^{[r]}}{q_{12}^{[r]} + q_{21}^{[r]}}$$

so that:

$$Q^{[r]} = \alpha^{[r]} \begin{bmatrix} -(1-\beta^{[r]}) & 1-\beta^{[r]} \\ \beta^{[r]} & -\beta^{[r]} \end{bmatrix}.$$

$\beta^{[r]} \in (0, 1)$ acts as a *bias* parameter: as $\beta^{[r]}$ approaches 1, the agent opinion becomes more biased towards opinion $\sigma^{[r]} = 1$, and viceversa when $\beta^{[r]}$ approaches 0. The second parameter $\alpha^{[r]} > 0$ can be seen as a time-scale parameter measuring *volatility*, i.e. how "prone" is agent r to changing opinion.

The second term $A^{[r]}(t) \in \mathbb{R}^{M\times M}$ in equation (2.1) accounts for the influence of the neighbours of agent r at time t, imposing the transition rates to opinion j for agent r (excluding the individual contribution given by matrix $Q^{[r]}$) to be proportional to the number of his/her neighbours who share opinion j, weighted

through the respective trustiness. For $i \neq j$:

$$\begin{aligned} a_{ij}^{[r]}(t) &= \lambda_j \eta^{[r]} \sum_{s \in \mathcal{N}^{[r]}} w_{rs} \mathcal{I}_{\sigma^{[s]}(t)=j} \\ \mathcal{I}_{\sigma^{[s]}(t)=j} &= \begin{cases} 1, & \text{if } \sigma^{[s]}(t) = j \\ 0, & \text{otherwise} \end{cases} \end{aligned}$$

and the diagonal elements are defined as $a_{ii}^{[r]}(t) = -\sum_{j=1, j\neq i}^{M} a_{ij}^{[r]}(t)$, such that $A^{[r]}(t)\mathbf{1}_M = 0$.

The non-negative parameter $\eta^{[r]}$ represents the individual *influenceability* of agent r, i.e. how much he/she is susceptible to the opinion of his/her neighbours. An agent with $\eta^{[r]} = 0$ is not influenced by the others and is effectively isolated. As $\eta^{[r]}$ increases, agent r is more and more influenced by the opinion of others.

The non-negative parameter λ_j reflects the *interaction intensity* among agents regarding opinion j. In the context of this thesis, λ_j assumes values in $[0, 1]$ and it represents the fraction of messages in favour of opinion j made visible by the network, with respect to the total number of messages in favour of j. In online social networks, λ_j represents the effect of content-filtering algorithms applied by the network manager, which limit the fraction of posts in favour of opinion j to which users are exposed. The extreme case $\lambda_j = 0$ corresponds to a total censorship by the network manager of posts that are in favour of j, effectively stopping the spread of opinion j through mutual interactions between users. Conversely, $\lambda_j = 1$ represents an unfiltered circulation of posts in favour of opinion j. Moreover, the interaction intensity parameters acts as "centralized control knobs" which affect the network globally, in contrast with the localized nature of the influence given by the neighbours of an agent.

Due to its stochastic nature, the model can be conveniently analysed by means of Monte Carlo simulations based on the Gillespie algorithm illustrated in chapter 3.

2.3 Theoretical analysis of the model

When $M = 2$ (binary opinion), the state of each agent at time t is completely defined by a single scalar: the probability that the considered agent r has opinion 1 at time t, denoted with $z_r(t)$. It can be shown [7] that, if $\lambda_j = \lambda$, $\forall j \in \mathcal{M}$, then $z(t) = \begin{bmatrix} z_1(t) & \cdots & z_N(t) \end{bmatrix}'$ obeys:

$$\dot{z}(t) = -(F + \lambda HL)z(t) + g,$$

where $F = \text{diag}\{\alpha^{[r]}\}$, $g = \text{col}\{\alpha^{[r]}\beta^{[r]}\}$, $H = \text{diag}\{\eta^{[r]}\}$, and $L = I_N - W$. The symbol $\text{diag}\{v^{[r]}\}$ denotes the diagonal matrix with the scalar elements $v^{[r]}$, $r \in \{1, 2, ..., N\}$, on its diagonal, $\text{col}\{v^{[r]}\}$ indicates the column vector obtained by stacking the scalars $v^{[r]}$, and I_N is the $N \times N$ identity matrix.

For $t \to +\infty$, the solution of the differential equation converges asymptotically to:

$$\bar{z}(\lambda) = (I_N + \lambda F^{-1} HL)^{-1} \bar{z}(0),$$

where $\bar{z}(0) = \text{col}\{\beta^{[r]}\}$ is the vector of the probabilities when the agents are isolated ($\lambda = 0$).

From $z(t)$ it is straightforward to calculate the expected value at time t of the *vote share* $s_1(t)$, i.e. the fraction of agents sharing opinion 1:

$$E[s_1(t)] = E\left[\frac{n_1(t)}{N}\right] = \frac{1}{N}\mathbf{1}'_N z(t),$$

where $n_1(t)$ is the random variable describing the number of agents in opinion 1 at time t.

Given the value of λ, the expected value of $s_1(t)$ at steady-state is then calculated with:

$$E[\bar{s}_1(\lambda)] = \frac{1}{N}\mathbf{1}'_N \bar{z}(\lambda). \tag{2.2}$$

Stochastic social power

Equation (2.2) can be rewritten as follows:

$$E[\bar{s}_1(\lambda)] = \sum_{r=1}^{N} \psi_r(\lambda)\bar{z}_r(0),$$

where, denoting with $e_r(N)$ the r-th column of the $N \times N$ identity matrix I_N, the scalar:

$$\psi_r(\lambda) := \frac{1}{N}\mathbf{1}'_N(I_N + \lambda F^{-1}HL)^{-1}e_r(N)$$

can be interpreted as the *stochastic social power* [7] of agent r, representing the contribution of the isolated bias $\bar{z}_r(0) = \beta^{[r]}$ to the steady-state expected vote share, for a given value of λ.

In general, $\psi_r(\lambda)$ is a proper rational function of λ, and depends both on the graph topology and the ratios $\eta^{[r]}/\alpha^{[r]}$ of all the agents.

The stochastic social power essentially represents a measure of the influence of each agent on the average behaviour of the network. For this reason, it lends itself to be used as an index of centrality for assessing the importance of a given node in a social network.

Vote share variance

Assuming $M = 2$, let $v^r(t) := \mathcal{I}_{\sigma^{[r]}(t)=1}$ and $V(t) := E[v(t)v(t)']$. $V(t)$ is the correlation matrix of the random vector $v(t) = \mathrm{col}\{v^r(t)\}$. The diagonal element $V_{rr}(t)$ represents $E[v^r(t)v^r(t)] = E[v^r(t)]$, i.e. the probability that agent r has opinion 1 at time t, while the off-diagonal elements $V_{rs}(t)$ represent $E[v^r(t)v^s(t)]$, i.e. the probability that agents r and s share opinion 1 at time t.

Assuming $\lambda_j = \lambda$, $\forall j \in \mathcal{M}$, the time evolution of $V(t)$ is the solution of the following differential equation [8]:

$$\dot{V}(t) = \hat{F}(\lambda)V(t) + V(t)\hat{F}(\lambda)' + gz(t)' + z(t)g' + D(V(t)),$$

where $\hat{F}(\lambda) = -(F + \lambda HL)$, and F, H, L and g have the same definitions intro-

duced in section 2.2. The diagonal matrix $D(V(t))$ is given by:

$$\begin{aligned} D(V) &= \operatorname{diag}\left(\hat{F}(\lambda)\operatorname{diag}(V)+g\right) \\ &\quad -\operatorname{diag}\left(\operatorname{diag}\left(\hat{F}(\lambda)V+V\hat{F}(\lambda)+g(\operatorname{diag}(V))'+\operatorname{diag}(V)g'\right)\right). \end{aligned}$$

The notation used in the previous expression is the following: if v is a vector, $V=\operatorname{diag}(v)$ is a diagonal matrix with v on its diagonal, while if V is a square matrix, $v=\operatorname{diag}(V)$ is the column vector containing the diagonal elements of V.

The steady-state correlation matrix $\bar{V}(\lambda)$ can be computed as the solution of the equation:

$$\hat{F}(\lambda)\bar{V}(\lambda)+\bar{V}(\lambda)\hat{F}(\lambda)'+g\bar{z}(\lambda)'+\bar{z}(\lambda)g'+D(\bar{V}(\lambda))=0.$$

Numerical methods which can efficiently evaluate both the differential and algebraic equations just presented can be found in [8]. From the correlation matrix $V(t)$ it is possible to compute the second-order moment of the vote share:

$$E[s_1^2(t)]=\frac{1}{N^2}E[\mathbf{1}_N'v(t)v(t)'\mathbf{1}_N]=\frac{1}{N^2}\sum_r\sum_s V_{rs}(t),$$

and thus the variance:

$$\sigma_{s_1}^2(t)=E[s_1^2(t)]-(E[s_1(t)])^2=\frac{1}{N^2}\sum_r\sum_s\left(V_{rs}(t)-V_{rr}(t)V_{ss}(t)\right).$$

Due to the ergodicity of the process, the steady-state variance depends only on λ and can be calculated as:

$$\bar{\sigma}_{s_1}^2(\lambda)=\frac{1}{N^2}\sum_r\sum_s\left(\bar{V}_{rs}(\lambda)-\bar{V}_{rr}(\lambda)\bar{V}_{ss}(\lambda)\right).$$

All the theoretical results just presented are valid under the hypothesis of *unbiased interaction intensity parameters*, i.e. $\lambda_j=\lambda,\ \forall j\in\mathcal{M}$. At the time of writing, no such results exist for the case of biased interaction intensity parameters.

Chapter 3

Simulation Algorithm

Chapter 3 introduces Gillespie algorithm, a Monte Carlo method employed in this thesis for carrying out numerical simulations of the multi-agent Markovian model described in chapter 2. Then, section 3.2 provides some insights on the computation of the contingency tables required for the calculation of indices of association between agents.

3.1 Gillespie algorithm

Gillespie algorithm is an algorithm used to simulate stochastic processes that proceed by "jumps", i.e. discrete movements with random arrival times. It was developed by Daniel Gillespie in [18, 19] with the intent to numerically simulate systems of coupled chemical reactions. It is a *Monte Carlo method*, since it is based on generating random samples from a given probability distribution in order to determine the outcome of each state-changing event in the system to be simulated.

The algorithm is well-suited for the simulation of the model introduced in chapter 2, due to the fact that each agent is modelled as a Markov process and, in view of the *Markov property*, the permanence time of the agent in a given state follows an exponential distribution. In particular, the transition rate matrix of a

generic agent $r \in \mathcal{N}$ is $\tilde{Q}^{[r]}(t) = [\tilde{q}_{ij}^{[r]}(t)] \in \mathbb{R}^{M \times M}$, where:

$$\begin{aligned} \tilde{q}_{ij}^{[r]}(t) &= q_{ij}^{[r]} + \lambda_j \eta^{[r]} \sum_{s \in \mathcal{N}^{[r]}} w_{rs} \mathcal{I}_{\sigma^{[s]}(t)=j}, \qquad i \neq j \\ \tilde{q}_{ii}^{[r]}(t) &= -\sum_{j=1, j \neq i}^{M} \tilde{q}_{ij}^{[r]}(t) \end{aligned}$$

The definition of each term can be found in section 2.2. In this model, the time required to agent r for jumping away from state i is exponentially distributed with parameter $\mu^{[r]} = -\tilde{q}_{ii}^{[r]}$.

Algorithm outline

Let $P(t_{min}, s, j)dt_{min}$ be the probability at time t that the next change of opinion will occur in the differential time interval $(t + t_{min}, t + t_{min} + dt_{min})$, it will involve agent s, and the new opinion of agent s will be opinion j. The main steps of Gillespie algorithm applied to the multi-agent Markovian model are:

Algorithm 1 Gillespie algorithm

Input: The simulation end time T, the initial state $\sigma^{[r]}(0) \in \mathcal{M}$, $\forall r \in \mathcal{N}$

$t \leftarrow 0$
while $t < T$ **do**
  $\mu \leftarrow 0$
  for all $r \in \mathcal{N}$ **do**
    $i \leftarrow \sigma^{[r]}(t)$
    $\mu^{[r]} \leftarrow -\tilde{q}_{ii}^{[r]}$
    $\mu \leftarrow \mu + \mu^{[r]}$
  end for
  Extract from $P(t_{min}, s, j)$ the time interval t_{min} to the next opinion change, the agent s changing opinion, and the new opinion j of s.
  $t \leftarrow t + t_{min}$
  $\sigma^{[s]}(t) \leftarrow j$
end while

One big advantage of Gillespie algorithm when applied to the multi-agent Markovian model of chapter 2 lies in the fact that, in the hypothesis of keeping fixed the values of λ_j, $\forall j \in \mathcal{M}$, and of $\eta^{[r]}$, $\forall r \in \mathcal{N}$, state-changing events

have a "local" effect. In fact, whenever an agent changes opinion, only the agent's neighbours are affected and thus, in step 6, there is no need to recompute the values of $\mu^{[r]}$ for every single agent in the network.

Gillespie presented in [18] two equivalent procedures for carrying out the extractions from the joint probability density function $P(t_{min}, s, j)$ of step 9: the *direct method* and the *first-reaction method*.

The direct method

The joint distribution $P(t_{min}, s, j)$ can be rewritten as follows:

$$P(t_{min}, s, j) = P_1(t_{min})P_2(s, j \mid t_{min}),$$

where $P_1(t_{min})dt_{min}$ is the probability that the next opinion change will be in the interval $(t+t_{min}, t+t_{min}+dt_{min})$, irrespective of which agent changes opinion and his/her new opinion, while $P_2(s, j \mid t_{min})$ is the probability that the involved agent and opinion will be s and j, respectively, given that the opinion change occurs at time $t + t_{min}$. In the system under study, $P_2(s, j \mid t_{min})$ is actually independent from t_{min}.

The direct method consists in extracting three random numbers x_1, x_2, x_3 from the uniform distribution $\mathcal{U}(0, 1)$. Then, t_{min} is sampled from $P_1(t_{min})$, which is an exponential distribution of parameter μ:

$$\begin{aligned} P_1(t_{min}) &= \mu e^{-\mu t_{min}} \\ t_{min} &= \frac{1}{\mu} \ln\left(\frac{1}{x_1}\right) \end{aligned}$$

Given the independency from t_{min}, $P_2(s, j \mid t_{min})$ can be rewritten as:

$$P_2(s, j \mid t_{min}) = P_3(s)P_4(j \mid s).$$

Both P_3 and P_4 are categorical distribution depending on the state of the system:

$$\begin{aligned} P_3(s) &= \left[\frac{\mu^{[1]}}{\mu} \quad \dots \quad \frac{\mu^{[r]}}{\mu} \quad \dots \quad \frac{\mu^{[N]}}{\mu}\right] \\ P_4(j \mid s) &= \left[\frac{\tilde{q}^{[s]}_{i1}(t)}{\mu^{[s]}} \quad \dots \quad \frac{\tilde{q}^{[s]}_{i,i-1}(t)}{\mu^{[s]}} \quad \frac{\tilde{q}^{[s]}_{i,i+1}(t)}{\mu^{[s]}} \quad \dots \quad \frac{\tilde{q}^{[s]}_{iM}(t)}{\mu^{[s]}}\right], \quad i = \sigma^{[s]}(t) \end{aligned}$$

Thus, agent s can be determined as the integer for which:

$$\sum_{r=1}^{s-1} \mu^{[r]} < x_2 \mu \leq \sum_{r=1}^{s} \mu^{[r]},$$

while j is the integer for which:

$$\sum_{\substack{k=1 \\ k \neq i}}^{j-1} \tilde{q}^{[s]}_{ik}(t) < x_3 \mu^{[s]} \leq \sum_{\substack{k=1 \\ k \neq i}}^{j} \tilde{q}^{[s]}_{ik}(t),$$

where i was the opinion of agent s before the extraction.

Alternatively, the commuting agent s and his/her new opinion j can be determined with a single extraction from a categorical distribution with NM categories and suitable event probabilities. An example of appropriate distribution is:

$$\left[\frac{\mu^{[1]}_1}{\mu} \quad \dots \quad \frac{\mu^{[N]}_1}{\mu} \quad \dots \quad \frac{\mu^{[1]}_M}{\mu} \quad \dots \quad \frac{\mu^{[N]}_M}{\mu}\right],$$

where:

$$\mu^{[r]}_k = \begin{cases} \tilde{q}^{[r]}_{ik}(t), \ i = \sigma^{[r]}(t), & \text{if } \sigma^{[r]}(t) \neq k \\ 0, & \text{otherwise} \end{cases}$$

In this case, denoting with c the index of the extracted category, the new opinion and the commuting agent are given by:

$$j = \left\lceil \frac{c}{N} \right\rceil \qquad\qquad s = c - N(j-1)$$

The first-reaction method

The first-reaction method consists in the extraction of $N+1$ random numbers $x_1, ..., x_{N+1}$ from $\mathcal{U}(0,1)$, and the sampling from the exponential distributions

$P_r(t^{[r]}) = \mu^{[r]} e^{-\mu^{[r]} t^{[r]}}$, $r \in \mathcal{N}$, of the "tentative event times":

$$t^{[r]} = \frac{1}{\mu^{[r]}} \ln\left(\frac{1}{x_r}\right), \qquad r \in \mathcal{N},$$

which are used to determine:

$$t_{min} = \min\{t^{[r]}\}, \qquad r \in \mathcal{N}.$$

The commuting agent s is selected as the one with $t^{[s]} = t_{min}$, while his new opinion j is determined from x_{N+1} in the same way described in the direct method.

In general, the direct method is more computationally efficient with respect to the first-reaction method, because the latter requires the calculation of N logarithms, compared to the former in which only a single logarithm needs to be computed.

3.2 Computation of the contingency tables

The calculation of indices for assessing the association between agents requires the computation of the *contingency table* associated to each pair of agents in the network. If $\mathcal{M} = \{1, ..., M\}$ denotes the set of opinions available in the system, the contingency table $Y^{[rs]}$ of agents r and s is a $M \times M$ matrix listing the frequency distribution of the agents' opinions at the end of a system realization, weighted by the permanence time of the agents in the various opinions. The generic element $y_{ij}^{[rs]}$ of $Y^{[rs]}$ corresponds to the amount of time in which agent r was in opinion i while agent s was in opinion j. It holds true that:

$$\sum_{i}^{M} \sum_{j}^{M} y_{ij}^{[rs]} = T, \qquad \forall r, s \in \mathcal{N},$$

where T is the total simulation time, and $\mathcal{N} = \{1, ..., N\}$ is the set of all the agents in the network. In total, there are N^2 possible pairs of agents $(r, s) \in \mathcal{N}^2$, but only $\frac{N(N-1)}{2}$ contingency tables are actually needed, since the tables $Y^{[rr]}$, $r \in \mathcal{N}$, do not give any meaningful information, while each table $Y^{[rs]}$, $r \neq s$, is as informative

as $Y^{[sr]}$.

As described in section 3.1, the system evolves in time steps extracted from an exponential distribution dependent on the state and parameters of the system. In the sequel, the symbol t_k, $k \in \{0, ..., K\}$, will be used to indicate the time instant in which the k-th event in the system occurred. $k = 0$ and $k = K$ are the only events not associated to a change of opinion in any of the agents in the network, and the related time instants are set to $t_0 = 0$ and $t_K = T$. The time interval between two consecutive events will be denoted with $t_{min,k} = t_k - t_{k-1}$, $k \in \{1, ..., K\}$.

An inefficient way of computing the contingency tables would be, at every event $k > 0$, to update in each table $Y^{[rs]}$, $r < s$, $r, s \in \mathcal{N}$, the element $y_{ij}^{[rs]}$ to $y_{ij}^{[rs]} + t_{min,k}$, where $i = \sigma^{[r]}(t_{k-1})$, i.e. the opinion of agent r in the interval $[t_{k-1}, t_k)$, and $j = \sigma^{[s]}(t_{k-1})$. The low efficiency is given by the fact that, with this method, $\frac{N(N-1)}{2}$ memory accesses are required for each of the K events occurred during a realization of the system evolution.

A slightly more efficient way of proceeding makes use of the fact that, in view of how Gillespie algorithm works, at each time step no more and no less than one agent changes opinion. Let the scalar τ_r, $r \in \mathcal{N}$, denote the time elapsed since agent r last changed opinion. Algorithm 2 shows in pseudo-code form the main steps for calculating the contingency tables of the agents. As already noted, the generic table $Y^{[rs]}$ is as informative as $Y^{[sr]}$, and thus there is no need to compute both of them. To account for this, in steps 17 and 28 only table $Y^{[rs]}$ is updated if $r < s$ and only $Y^{[sr]}$ if $r > s$.

With this procedure, each of the first $K - 1$ events in the system requires $N - 1$ memory accesses to update the tables, plus the operations to be carried out on the scalars τ_r, $r \in \mathcal{N}$, which scale as $O(N)$. The last event (end of the simulation) requires $\frac{N(N-1)}{2}$ memory accesses as with the previous method. Thus, the complexity of this algorithm is $O(NK+N^2)$, whereas for the previous algorithm it was $O(N^2K)$.

Algorithm 2 Contingency tables computation

Input: The time instants t_k of the events in the system, $k \in \{1, ..., K\}$

for all $r \in \mathcal{N}$ **do**
  $\tau_r \leftarrow 0$
  for all $s \in \mathcal{N}$ **do**
    Initialize to 0 the table $Y^{[rs]}$
  end for
end for
$t_0 \leftarrow 0$
for $k = 1$ to $K - 1$ **do**
  $t_{min,k} \leftarrow t_k - t_{k-1}$
  for all $r \in \mathcal{N}$ **do**
    $\tau_r \leftarrow \tau_r + t_{min,k}$
  end for
  $r \leftarrow$ The agent who changed opinion at t_k
  $i \leftarrow \sigma^{[r]}(t_{k-1})$
  for all $s \in \mathcal{N} - \{r\}$ **do**
    $j \leftarrow \sigma^{[s]}(t_{k-1})$
    $y_{ij}^{[rs]} \leftarrow y_{ij}^{[rs]} + \min\{\tau_r, \tau_s\}$
  end for
  $\tau_r \leftarrow 0$
end for
$t_{min,K} \leftarrow T - t_{K-1}$
for all $r \in \mathcal{N}$ **do**
  $\tau_r \leftarrow \tau_r + t_{min,K}$
end for
for all $(r, s) \in \mathcal{N}^2$ **do**
  $i \leftarrow \sigma^{[r]}(T)$
  $j \leftarrow \sigma^{[s]}(T)$
  $y_{ij}^{[rs]} \leftarrow y_{ij}^{[rs]} + \min\{\tau_r, \tau_s\}$
end for

Chapter 4

Networks

Chapter 2 introduced the multi-agent Markovian model governing the opinion dynamics of the agents, which describes the interactions between them through an underlying network where nodes corresponded to agents and links corresponded to reciprocal influences between them. This chapter elaborates further on networks, describing the main features found in social networks and introducing the two models for generating synthetic social networks that have been used throughout the rest of this thesis. Then, in section 4.2, it is introduced the topic of community detection and Louvain algorithm, for judging in subsequent chapters the effect of the community structure on an opinion manipulation attack, and the effect of an attack on the opinion distribution of the communities. Another important topic is the identification of the most influential nodes in the network, which is treated in section 4.3 with the introduction of centrality indices.

Remarks on networks

A network is a collection of *nodes* connected by *links*. The topological structure of a network with N nodes can be represented with a $N \times N$ matrix called *adjacency matrix*, which has elements: $a_{ij} \neq 0$ if there is a link from node i to node j, and $a_{ij} = 0$ otherwise.

If all the links are bidirectional, meaning that for every link from node i to node j there exists a link from j to i, the network is said to be *undirected* and the adjacency matrix is symmetric, otherwise the network is said to be *directed*.

If links do not have weights (and there are no multi-links or self-loops), elements of the adjacency matrix can be either 1 (link present) or 0 (link absent), and the network is said to be *binary*. Otherwise, if there are weights attached to links, the network is said to be *weighted* and the generic element a_{ij} of the adjacency matrix is equal to the weight of the link from node i to node j.

4.1 Social networks models

A social network is a social structure where nodes represent people and links represent relationships between them, like "friend of", "colleague of", etc. In online social networks, users are related to profiles (nodes) and they can connect with each other by "sending friend requests". In many social networks, like *Facebook*, this type of relationship is symmetrical, meaning that if user i is linked to user j, then also user j is linked to user i.

Essential characteristics for social networks are believed to include [20, 21]:

- A broad degree distribution;
- An assortative mixing pattern;
- A high clustering coefficient;
- A short average path length;
- The presence of a community structure;
- A broad community size distribution.

The *degree* k_i of a node i is the number of links incident to i. The *degree distribution* p_k of a network is a probability distribution which provides the probability that a randomly chosen node of the network has degree k, i.e.:

$$p_k = \frac{N_k}{N},$$

where N is the total number of nodes in the network, and N_k is the number of them with degree equal to k. A broad degree distribution is a distribution where

the degree can span several orders of magnitude. An example of a broad degree distribution is the power-law distribution:

$$p_k \sim k^{-\gamma}, \qquad \gamma > 0.$$

A network with *assortative mixing* is a network where nodes tend to connect with other nodes with a similar degree, i.e. high-degree nodes mostly tend to connect to other high-degree nodes, and viceversa. Assortativity can be assessed through the *degree correlation function*:

$$k_{nn}(k) = \sum_{k'} k' \Pr(k'|k),$$

where $\Pr(k'|k)$ is the probability that a link of a node with degree k connects to a node with degree k'. Thus, $k_{nn}(k)$ is the average degree of the neighbours of all the nodes with degree k. In networks with assortative mixing, $k_{nn}(k)$ increases with k.

The *clustering coefficient* is a measure of the local link density in a network. The clustering coefficient C_i of node i is defined as:

$$C_i = \frac{2L_i}{k_i(k_i - 1)},$$

where L_i is the number of links between the neighbours of node i, and k_i is its degree. The measure can be extended to the whole network by taking the average of C_i over all nodes i, and it represents the probability that two neighbours of a randomly selected node link to each other.

A *path* is a sequence of nodes such that the elements of each pair of consecutive nodes are adjacent. The length of the shortest path between two nodes i and j, i.e. the one which has minimum length, is called the *distance* between node i and node j, and it is indicated with the symbol d_{ij}. The *average path length* d_{avg} is the average distance between all pairs of nodes in the network:

$$d_{avg} = \frac{1}{N(N-1)} \sum_{i \neq j} d_{ij}.$$

A network with "short" average path length indicates that distances are orders of magnitude smaller than the size of the network N, which typically means that d_{avg} scales as $\log(N)$.

In the context of social networks, a *community* is a locally dense connected subgraph of a network. The *size* of a community is the number of nodes which belong to that community. Similar to the degree distribution, a community size distribution is broad if it spans multiple orders of magnitude.

In order to carry out the simulations of the system under study in a setup similar to real social networks, two models for generating synthetic social networks which possess most of the previously listed characteristics has been chosen: Lancichinetti - Fortunato - Radicchi networks and Toivonen et al. networks.

Lancichinetti - Fortunato - Radicchi networks

Lancichinetti - Fortunato - Radicchi networks (which will be called "LFR networks" throughout the rest of the document) have been developed in [21] as a benchmark for community detection algorithms. LFR networks account for heterogeneity in degree distribution and, differently from many other social network models, they also come with a built-in community structure and a heterogeneous community size distribution.

The user selects the number of nodes N in the network, the average degree k_{avg} of the network, the exponents for the degree distribution and community size distribution power-laws γ and β, and the mixing coefficient μ.

The algorithm for generating LFR networks employs the *configuration model* [22] in order to construct networks with a pre-determined degree distribution. The procedure consists of two steps: First a degree extracted from the desired distribution is assigned to each node, represented as "half-links". In the case of LFR networks the degree distribution is a power-law with exponent γ. Then, a random pair of "half-links" is selected and the two nodes connected. This procedure is repeated until all "half-links" have been paired. The configuration model may introduce unwanted features like self-loops (links connecting a node to itself) and multi-links (multiple links between the same pair of nodes), which have to be handled appropriately. However, the expected number of self-loops and multi-links goes to 0 as

the network size N approaches ∞.

A LFR network is constructed through the following steps:

1. Each node degree is extracted from a power-law distribution with exponent γ, with the extremes of the distribution chosen in order to achieve the desired average degree k_{avg}. The nodes are then connected using the configuration model;

2. Each node shares a fraction $1-\mu$ of its links with the nodes belonging to its community, and a fraction μ of the links with the other nodes in the network;

3. The sizes of the communities are extracted from a power-law distribution with exponent β, with the additional constraint that the sum of all sizes must be equal to the the number N of nodes in the network;

4. Initially, all the nodes are not assigned to any community. In the first iteration, a node is assigned to a random community. If the community size exceeds the number of neighbours of the node inside the community, it enters the community, otherwise it remains unassigned. In successive iterations a unassigned node is again placed inside a random community, but this time, if the community is complete, a randomly selected node is kicked out of it. The procedure stops when there are no more unassigned nodes;

5. To enforce the condition given by the mixing coefficient μ, several degree-preserving rewiring steps are performed.

Toivonen et al. networks

Toivonen et al. networks have been developed in [20] with the objective to propose a simple model for generating undirected networks which reproduces all the main characteristics of social networks.

The algorithm consists of two growth processes: *random attachment* and *preferential attachment*. The local nature of the latter give rise to high clustering, assortativity and community structure.

The algorithm steps are:

1. Start with a "seed" network of N_0 nodes;

2. Pick on average $m_r \geq 1$ random vertices as "initial contacts" of the node to be added (random attachment);

3. Pick on average $m_s \geq 0$ neighbours of each initial contact as "secondary contacts" of the node to be added (preferential attachment);

4. Connect the new node to the initial and secondary contacts;

5. Repeat steps 2-4 until the network has grown to the desired size.

In steps 2 and 3, any non-negative distribution with expected values m_r and m_s can be used for selecting the number of initial and secondary contacts, respectively. However, if the distribution of the number of secondary contacts has a long tail, it may happen that the extracted number of secondary contacts is higher than the degree of the initial contact, biasing the distribution towards smaller degrees.

For the formation of an appreciable community structure, it is essential that the number of links to the neighbours of an initial contact varies, and that sometimes more than one initial contact are chosen in order for the node to be added to act as a "bridge" between communities.

4.2 Community detection

The community structure of a given network is often unknown a-priori. Given the topology of the network, being able to determine (if present) the underlying community structure can be helpful for devising specific opinion manipulation strategies based on that structure, and also to be able to compare the efficacy of an attack within the different communities.

Community detection algorithms can be divided in two main classes, based on whether the structure to be identified allows for the communities to overlap. Examples of algorithms for overlapping communities detection are the *clique percolation algorithm* [23] and the *link clustering algorithm* [24]. For simplicity, in this thesis only non-overlapping communities will be considered.

Two possible procedures for tackling the problem of non-overlapping community detection are *hierarchical clustering* and *modularity maximization*.

Hierarchical clustering is based on the definition of a *similarity matrix*, whose elements indicate the "distance" of any pair of nodes in the network, and then groups of nodes with high similarity are iteratively identified. *Agglomerative* algorithms, like Ravasz algorithm [25], construct communities by merging nodes with high similarity, while *divisive* algorithms, like Girvan-Newman algorithm [26], isolate communities by removing links with low similarity.

Modularity maximization is based on a quantity called *modularity*, which measures how much the link density of a given community deviates from the link density of the same set of nodes in a randomly rewired network. The underlying assumption is that randomly wired networks lack an inherent community structure. For a weighted network, the modularity M of a given partition can be calculated as follows:

$$M = \frac{1}{2m} \sum_{ij} \left(a_{ij} - \frac{k_i k_j}{2m} \right) \delta(c_i, c_j),$$

where m is the sum of the weights of all the links in the network, a_{ij} is the weight of the link between node i and node j, k_i and k_j are the sum of the weights of the links incident to nodes i and j, $\delta(\cdot, \cdot)$ is the *Kronecker delta function*, i.e. $\delta(a, b) = 1$ if $a = b$ and 0 otherwise, and c_i and c_j are the communities to which nodes i and j belong to.

Modularity maximization algorithms, like the greedy algorithm proposed by Newman [27] and Louvain algorithm [28], aim at obtaining a good community partition through an approximate maximization of the modularity M. Louvain algorithm has been adopted throughout this thesis for community detection, given its computational efficiency.

Louvain algorithm

Louvain algorithm is a modularity optimization algorithm proposed by Blondel et al. [28], and it consists of two iteratively repeated steps.

In the first step each node is assigned to a community of its own. Then, for each node i, it is evaluated the change in modularity ΔM of placing node i in the

community of one of its neighbours j. Node i will be moved in the community where the gain in modularity is the largest, but only if it is positive. This process is applied to all the nodes until no further improvements are possible.

The change in modularity ΔM of moving node i in community c is given by:

$$\Delta M = \left(\frac{\Sigma_{in} + k_{i,in}}{2m} - \left(\frac{\Sigma_{tot} + k_i}{2m} \right)^2 \right) - \left(\frac{\Sigma_{in}}{2m} - \left(\frac{\Sigma_{tot}}{2m} \right)^2 - \left(\frac{k_i}{2m} \right)^2 \right),$$

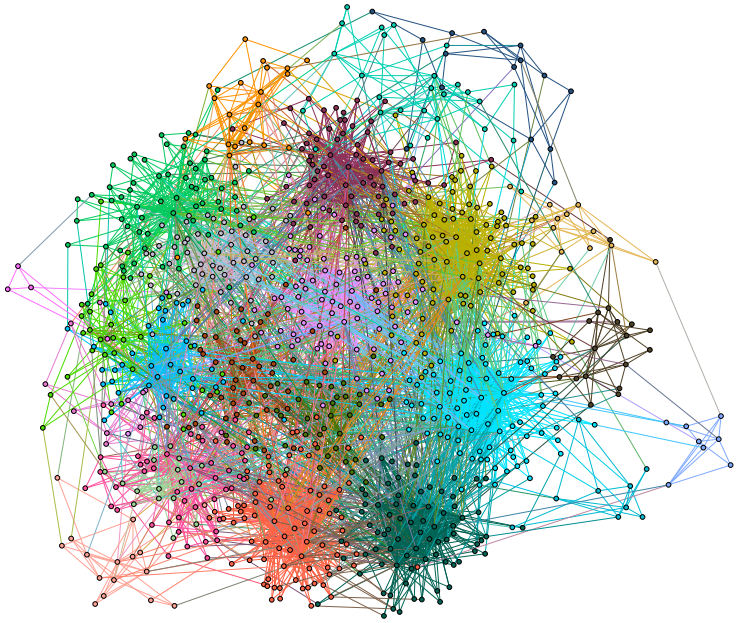
where Σ_{in} is the sum of the weights of the links inside c, Σ_{tot} is the sum of the weights of the links incident to nodes in c, k_i is the sum of the weights of the links incident to node i, $k_{i,in}$ is the sum of the weights of the links from i to nodes in c, and m is the sum of the weights of all the links in the network.

In the second step of the algorithm a new network is constructed, where the nodes correspond to the communities identified in the first step and the weights of the links between nodes are given by the sum of the weights of the links which connected the communities in the original network. Links that were internal to communities give rise to weighted self-loops. These two steps are iteratively repeated until there are no more improvements in modularity.

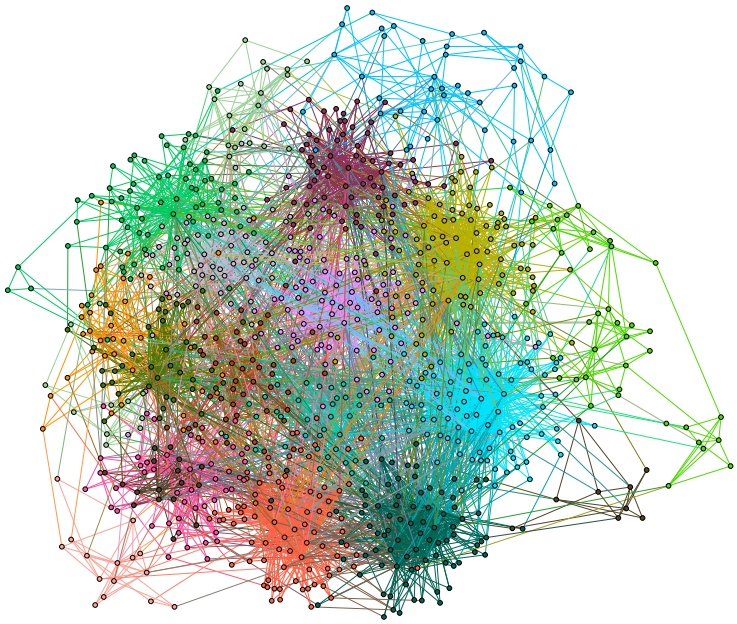
As an output example, Figure 4.1 shows the result of the application of Louvain algorithm to one of the LFR networks used in the simulations for this thesis, highlighting with different colors nodes belonging to different communities.

4.3 Centrality indices

Centrality indices are metrics used to measure the relative importance of the nodes belonging to a network. In the specific case of this thesis, centrality indices are employed with the objective to rank the agents based on their "influence" in the network, and then to select the top K of them to be *controlled*, i.e. to be polarized towards the specific opinion to be spread.



(a)



(b)

Figure 4.1: (a) *The built-in community structure of a LFR network.* (b) *The community structure of a LFR network identified using Louvain algorithm.* Nodes belonging to different communities are filled with different colors. The LFR network has 1000 nodes, average degree $k_{avg} = 10$, mixing coefficient $\mu = 0.25$, exponent of the degree distribution $\gamma = -2$, and exponent of the community size distribution $\beta = -1$. The modularity attained by Louvain algorithm in (b) is $M = 0.6738$.

Degree

The degree centrality C_{deg} of a node i is:

$$C_{deg}(i) = k_i,$$

where k_i is the degree of node i, i.e. the number of links incident to i.

Operatively, in the case of a binary and undirected network, the degree of node i can be calculated as follows:

$$k_i = \sum_{j=1}^{N} a_{ij},$$

where N is the network size, a_{ij} is the (i, j) element of the adjacency matrix A of the network, and $i, j \in \{1, ..., N\}$.

In the context of social networks, a high-degree node corresponds to a so-called "influencer", i.e. a person with many friends and acquaintances.

Strength

If the considered network is weighted, the generalization of the degree centrality is the strength centrality, which is the sum of the weights of all the links incident to a given node. Moreover, if the network is also directed, a distinction can be done between the in-strength, which is the sum of inward link weights, and the out-strength, which is the sum of outward link weights.

Betweenness

The betweenness centrality C_{betw} of a node i is:

$$C_{betw}(i) = \sum_{\substack{r \neq s \\ i \neq r \\ i \neq s}} \frac{\sigma_{rs}(i)}{\sigma_{rs,\text{tot}}},$$

where $\sigma_{rs,\text{tot}}$ is the total number of shortest paths from r to s, and $\sigma_{rs}(i)$ is the number of them which pass through i.

In undirected networks, a path σ is a sequence of nodes such that the elements

of each pair of consecutive nodes are adjacent. If the network is also binary, the length of the path is given by the number of edges belonging to the path. A shortest path between two nodes r and s is a path which starts in r, ends in s, and minimizes the path length.

The first formal description of the betweenness centrality is attributed to Freeman [29]. In the context of social networks, a node with high betweenness acts as a *bridge* for the flow of information between the people it connects.

Coreness

A node has coreness centrality equal to k (or, equivalently, k-coreness) if it belongs to a k-core of the network but not to a $(k+1)$-core.

A k-core is a maximal connected subnetwork in which all nodes have internal degree greater or equal than k.

The concept of k-core was first introduced by Seidman [30] in order to identify the most cohesive subsets in a social network. Further studies, such as [31], found that the most efficient "spreaders" are often those located in the core of the network.

Eigenvector centrality

The eigenvector centrality, often abbreviated with the acronym EVC, is a centrality measure where the score of a given node is proportional to the sum of the score of its neighbours, meaning that the importance of a node depends on how many and how important are the nodes it connects with.

The EVC γ_i of node i is thus given by:

$$\gamma_i = \frac{1}{\lambda} \sum_{r \in \mathcal{N}} a_{ir} \gamma_r,$$

where a_{ir} is the (i, r) element of the adjacency matrix A, $\mathcal{N}$ is the set of all the nodes in the network, and $\lambda \in \mathbb{R}$.

Letting $\gamma = \begin{bmatrix} \gamma_1 & \dots & \gamma_N \end{bmatrix}'$, $N = |\mathcal{N}|$, the previous equation can be rewritten in matrix form:

$$A\gamma = \lambda\gamma$$

If the network is connected, the only solution with $\gamma_i > 0$, $i \in \mathcal{N}$, is given by the eigenvector of the greatest eigenvalue λ.

PageRank, Google's famous web page ranking algorithm, is essentially similar to the eigenvector centrality.

Chapter 5

Manipulation of the Average Opinion

Chapter 5 illustrates the results obtained by carrying out opinion manipulation attacks on a social network modelled by the multi-agent Markovian model introduced in chapter 2 and simulated using Gillespie algorithm described in chapter 3. Section 5.1 introduces the topic by presenting the main features of an attack and the parameters of interest. The subsequent sections investigate the impact of varying the main parameters of an attack, such as the number of controlled agents, the agents selection criterion, etc. Lastly, section 5.6 describes briefly some peculiarities introduced by the presence of a community structure in the social network.

5.1 Introduction

In the context of this thesis, a *manipulation attack* carried out on a social network consists in the selection of a certain number of agents to be *controlled*, with the objective to shift the average opinion of the network towards a specific direction. In the sequel, it will be assumed that agents can only adopt opinions in the set $\mathcal{M} = \{1, 2\}$. Assuming that the opinion with index 1 is the one to be promoted, it is denoted with the symbol $s_1(t)$ the *vote share* at time t, i.e. the fraction of agents that have opinion 1 at time t with respect to the total number of agents in the network. Thus, the main objective of an attack is to manipulate the opinion

dynamics in order to maximize $E[\bar{s}_1(\lambda)] = \bar{\mu}_{s_1}$, where $\bar{s}_1(\lambda)$ is the steady-state value of the vote share as described in section 2.3. Of course, given a fixed amount of expendable *resources* (e.g. the number of controlled agents), the larger is $\bar{\mu}_{s_1}$ achieved by an attack, the better. The expected value of the vote share is not the only variable of interest: the dispersion of the vote share around its mean, quantified through the vote share variance $\sigma^2_{s_1}(t)$ and in particular its steady-state value $\bar{\sigma}^2_{s_1}(\lambda)$, plays a crucial role on being able to predict with a relative degree of confidence the actual outcome of a measure on the network, e.g. the result of an election in the hypothesis that opinion 1 and 2 represent preferences in a bipartisan election.

Prior to the attack, it is assumed that all the agents in the network are homogeneous with isolated transition rate matrix:

$$Q^{[r]} := \begin{bmatrix} -1 & 1 \\ 1 & -1 \end{bmatrix}, \qquad r \in \mathcal{N} = \{1, ..., N\},$$

and individual influenceability $\eta^{[r]} := \eta > 0$. Using the (α, β)-parametrization introduced in section 2.2, it is apparent that agents have a bias parameter $\beta^{[r]} = 0.5$, meaning that they are not biased towards any specific opinion, and volatility parameter $\alpha^{[r]} = 2$. The trustiness w_{rs} between the agents depends on whether the *standard* or the *degree-weighted normalized* model is adopted. Throughout the rest of the thesis the shorthands "std" and "dwn" will be used for referring to the two types of trustiness models. For the rest of the current chapter, it is also assumed that the interaction intensity parameters are unbiased, and there is no filtering action exerted by the network manager, i.e. $\lambda_j = 1$, $\forall j \in \mathcal{M}$.

The simulations of the attack have been carried out on two different models of synthetic social networks: *LFR* and *Toivonen et al.* networks, both introduced in section 4.1. All the synthetic networks employed in the simulations have been generated using the same set of parameters, chosen in order to have a similar average degree in both the network types:

- LFR networks: network size $N = 1000$, desired average degree $\bar{k}_{avg} = 10$, mixing coefficient $\mu = 0.25$, exponent of the degree distribution $\gamma = -2$, and exponent of the community size distribution $\beta = -1$;

- Toivonen et al. networks: network size $N = 1000$, number of initial contacts $k_1 \in \{1, 2, 3\}$ of a given node extracted from the discrete distribution $\begin{bmatrix} 0.7 & 0.2 & 0.1 \end{bmatrix}$ ($m_r = 1.4$ initial contacts, on average), and number of secondary contacts $k_2 \in \{0, 1, 2, 3, 4\}$ extracted from the discrete distribution $\begin{bmatrix} 0.13 & 0.14 & 0.13 & 0.2 & 0.4 \end{bmatrix}$ ($m_s = 2.6$ secondary contacts per initial contact, on average).

The LFR network employed in many of the simulations had actual average degree $k_{avg} = 9.66$, clustering coefficient $C = 0.3585$, average path length $d_{avg} = 3.2379$, and 18 communities with sizes ranging between 16 and 95 members, identified with modularity $M = 0.6738$ by using Louvain algorithm described in section 4.2. The Toivonen et al. network employed in many of the simulations had actual average degree $k_{avg} = 9.608$, clustering coefficient $C = 0.4832$, average path length $d_{avg} = 3.5207$, and 19 communities with sizes ranging between 31 and 97 members, identified with modularity $M = 0.5428$. For comparison, a 2009 snapshot [32] of a Facebook subnetwork composed by 63 392 nodes had average degree $k_{avg} = 12.886$, clustering coefficient $C = 0.2218$, average path length $d_{avg} = 4.3219$, and 63 communities with sizes ranging between 3 and 14 140 members, identified with modularity $M = 0.6361$. Figure 5.1 shows the cumulative degree distribution of each of the three networks.

The strategy employed in the opinion manipulation attack is characterized by two factors: the number K of controlled agents in the network, and the criterion used to select the K agents. A controlled agent has isolated transition rate matrix set to:

$$Q^{[r]} := \begin{bmatrix} -10^{-6} & 10^{-6} \\ 10^{6} & -10^{6} \end{bmatrix}, \qquad r \in \mathcal{C} \subset \mathcal{N},$$

where $\mathcal{C}$ denotes the set of the controlled agents. Those agents are highly polarized towards opinion 1, with bias parameter $\beta^{[r]} \approx 1$, and due to a very high volatility ($\alpha^{[r]} \approx 10^6$) they virtually never leave opinion 1, because even if they were to change opinion, they would immediately return back to opinion 1. Moreover, controlled agents have very low influenceability, e.g. $\eta^{[r]} = 10^{-6}$, $r \in \mathcal{C}$, so that they are basically uninfluenced by the neighbours.

Equation 2.2 allows to compute analytically the theoretical steady-state value

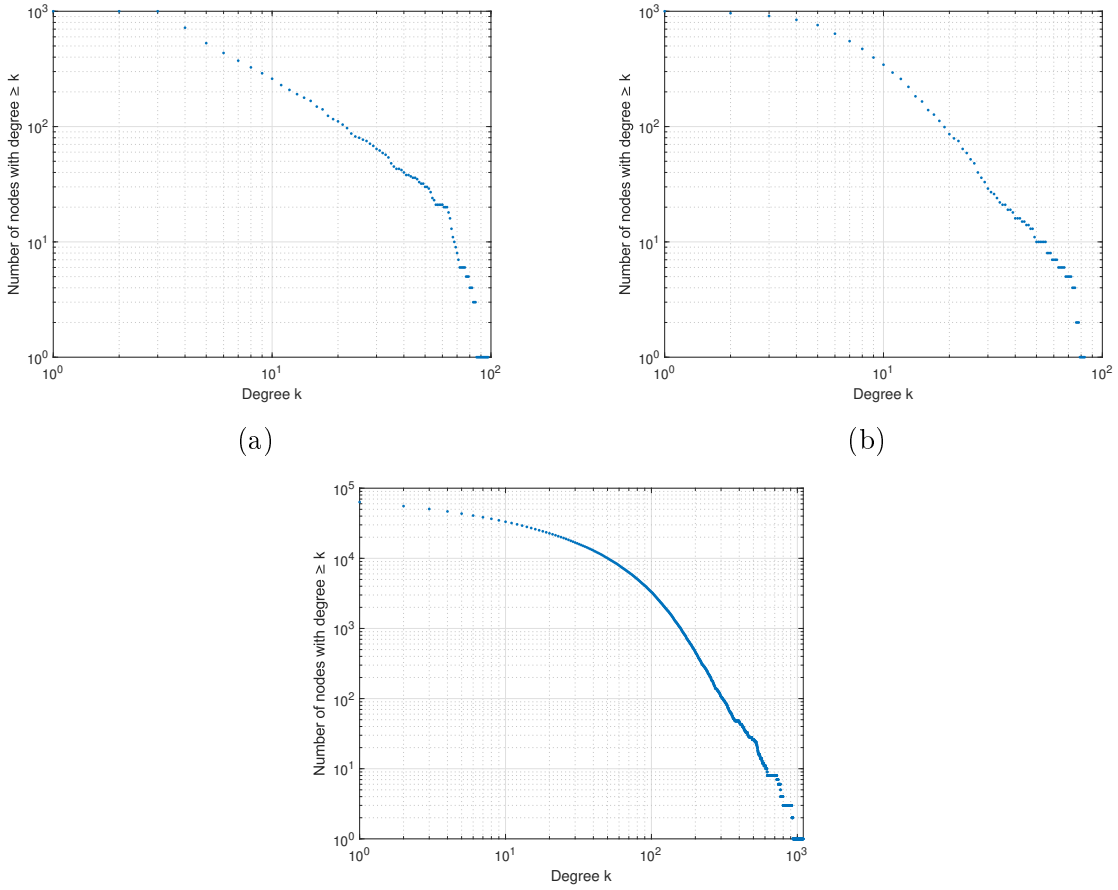


Figure 5.1: Cumulative degree distribution of a LFR network (a), a Toivonen et al. network (b), and a 2009 snapshot [32] of a Facebook subnetwork (c).

$\bar{\mu}_{s_1}$ of the mean vote share, given the model and attack parameters. Thus, denoting with $2^{\mathcal{N}}$ the power set of $\mathcal{N}$, i.e. the set of all the subsets of $\mathcal{N}$, it is possible to determine a function $f : 2^{\mathcal{N}} \to [0, 1]$ that relates the set of controlled agents $\mathcal{C} \subseteq \mathcal{N}$ to the value of $\bar{\mu}_{s_1}$ achieved by the attack. Since $2^{\mathcal{N}}$ is a finite set, for all $K \in \{1, ..., N\}$ it exists at least one optimal set $\mathcal{C}^o \in 2^{\mathcal{N}}$ with cardinality K that maximize $f(\mathcal{C})$. Let $\mathcal{A}$ be an algorithm that, given K, the model parameters, and the network topology, returns one optimal set $\mathcal{C}^o$ for the K controlled agents. Notice that $\mathcal{A}$ always exists, because it is always possible to find an optimal solution for the problem by trying all the combinations of K agents. However, the brute-force approach is computationally prohibitive, requiring in total $\binom{N}{K}$ trials (if $N = 1000$ and $K = 10$, $\binom{N}{K} \approx 2.63 \cdot 10^{23}$). In this thesis, the centrality indices described in section 4.3 will be employed as heuristics for approximating $\mathcal{A}$. All the agents in the network are sorted in decreasing order according to the chosen centrality index, and then the first K of them are selected as controlled agents.

5.2 Impact of the centrality index

In this section, six different attack strategies will be taken in consideration. Four strategies are based on standard centrality indices, considering the social network as binary and undirected: degree, betweenness, coreness, and eigenvector centrality. The remaining two strategies consider the network as weighted and directed, ordering the agents according to the importance given to them by their neighbours, quantified through the trustiness. The quantity $\sum_s w_{sr}$, $s \in \mathcal{N}^{[r]}$, corresponds to the in-strength of node r, i.e. the sum of the weights of the links pointing towards node r, where the weight of a generic link from r to s is given by the trustiness w_{rs}, and $\mathcal{N}^{[r]}$ denotes the set of the neighbours of agent r. Notice that, in general, $w_{rs} \neq w_{sr}$. Since there are two possible models for the trustiness between agents, the *standard* and the *degree-weighted normalized* trustiness models, there are also two different formulations of the strength that can be used as an index of centrality.

Figure 5.2 and Figure 5.3 show the results obtained varying the centrality index on a LFR network and a Toivonen et al. network, respectively. The remaining simulation parameters have been set to influenceability $\eta = 10$, degree-weighted

normalized trustiness model, and $K = 10$ controlled agents.

The theoretical steady-state mean $\bar{\mu}_{s_1}$ and standard deviation $\bar{\sigma}_{s_1}$ of the vote share $s_1(t)$ can be calculated as described in section 2.3. The two quantities can also be experimentally estimated from a long-enough realization of the process. In the sequel, the experimental mean and standard deviation of the vote share at steady-state will be indicated with the symbols $\hat{\mu}_{s_1}$ and $\hat{\sigma}_{s_1}$. The values $\bar{\mu}_{s_1}$, $\bar{\sigma}_{s_1}$, $\hat{\mu}_{s_1}$ and $\hat{\sigma}_{s_1}$ relative to the simulations of Figure 5.2 and Figure 5.3 can be found in Table 5.1, along with the aggregated stochastic social power ψ_{CA} of the controlled agents, calculated as:

$$\psi_{CA} = \sum_{r \in \mathcal{C}} \psi_r(\lambda),$$

where $\psi_r(\lambda)$ is the stochastic social power of agent r as described in section 2.3.

	$\hat{\mu}_{s_1}$	$\bar{\mu}_{s_1}$	$\hat{\sigma}_{s_1}$	$\bar{\sigma}_{s_1}$	ψ_{CA}
degree	0.7241	0.7261	0.0434	0.0442	0.4522
betweenness	0.7313	0.7333	0.0423	0.0424	0.4665
coreness	0.6307	0.6280	0.0604	0.0648	0.2560
eigenvector	0.6426	0.6434	0.0614	0.0628	0.2868
strength (std)	0.7244	0.7242	0.0448	0.0450	0.4484
strength (dwn)	0.7279	0.7289	0.0433	0.0437	0.4578

(a)

	$\hat{\mu}_{s_1}$	$\bar{\mu}_{s_1}$	$\hat{\sigma}_{s_1}$	$\bar{\sigma}_{s_1}$	ψ_{CA}
degree	0.7417	0.7427	0.0321	0.0313	0.4854
betweenness	0.7395	0.7425	0.0310	0.0314	0.4850
coreness	0.7220	0.7229	0.0378	0.0376	0.4457
eigenvector	0.7421	0.7427	0.0308	0.0313	0.4854
strength (std)	0.7440	0.7433	0.0316	0.0314	0.4866
strength (dwn)	0.7431	0.7424	0.0309	0.0315	0.4848

(b)

Table 5.1: Experimental mean $\hat{\mu}_{s_1}$ and standard deviation $\hat{\sigma}_{s_1}$ of the vote share and their theoretical steady-state values $\bar{\mu}_{s_1}$ and $\bar{\sigma}_{s_1}$, for different centrality indices, in a LFR network (a) and a Toivonen et al. network (b). The last column of the table contains the stochastic social power ψ_{CA} of the controlled agents.

In the LFR network of Figure 5.2, the betweenness centrality achieved the highest value for the mean vote share, both experimentally and theoretically, although

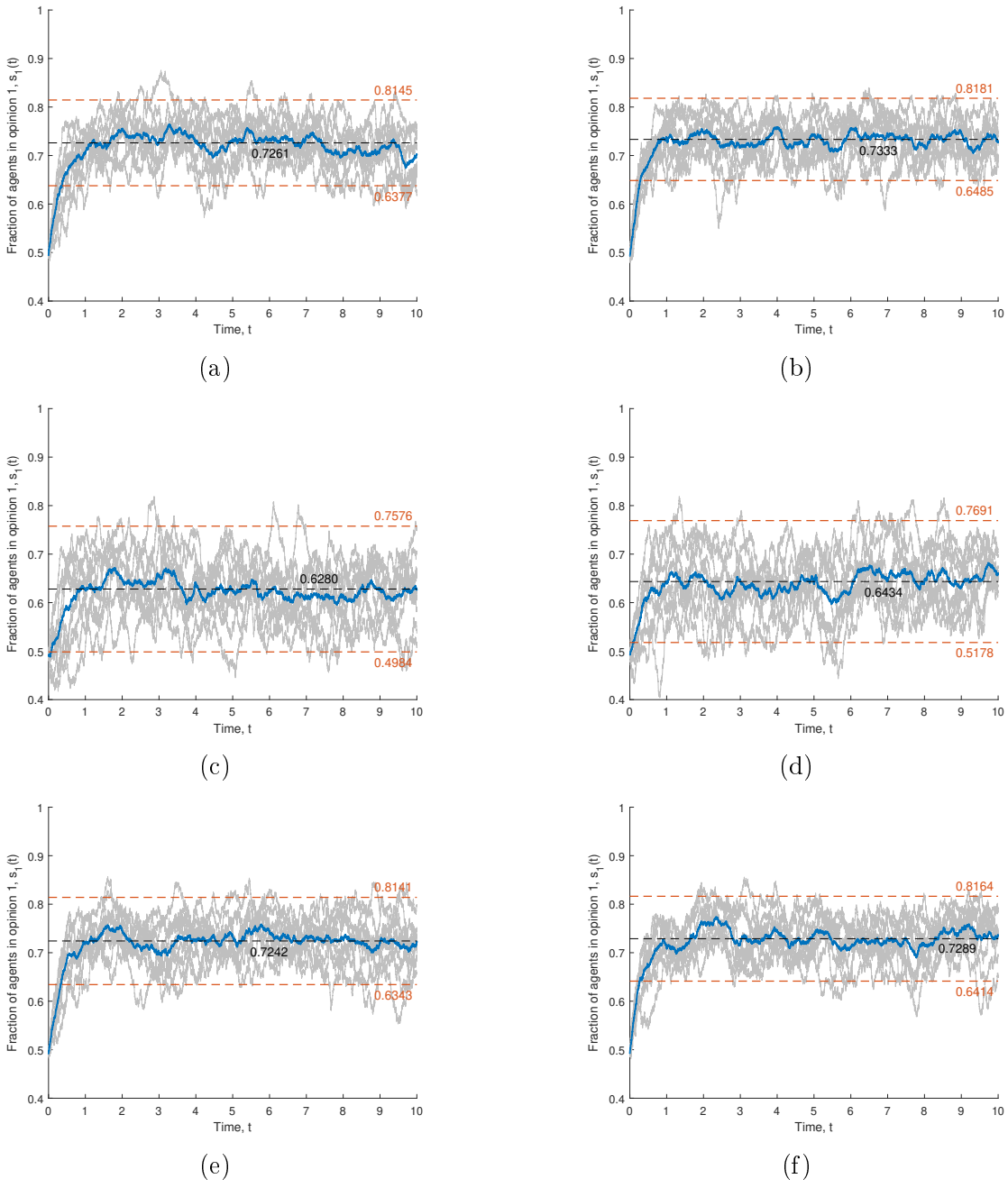


Figure 5.2: Time evolution of the vote share $s_1(t)$ in a LFR network using as centrality index: degree (a), betweenness (b), coreness (c), eigenvector centrality (d), strength based on the standard trustiness model (e), and strength based on the degree-weighted normalized trustiness model (f). The gray lines represent the ten single realizations. The blue line is the average of the realizations. The black dashed line corresponds to the theoretical mean vote share $\bar{\mu}_{s_1}$. The orange dashed lines are traced at $\pm 2\bar{\sigma}_{s_1}$.

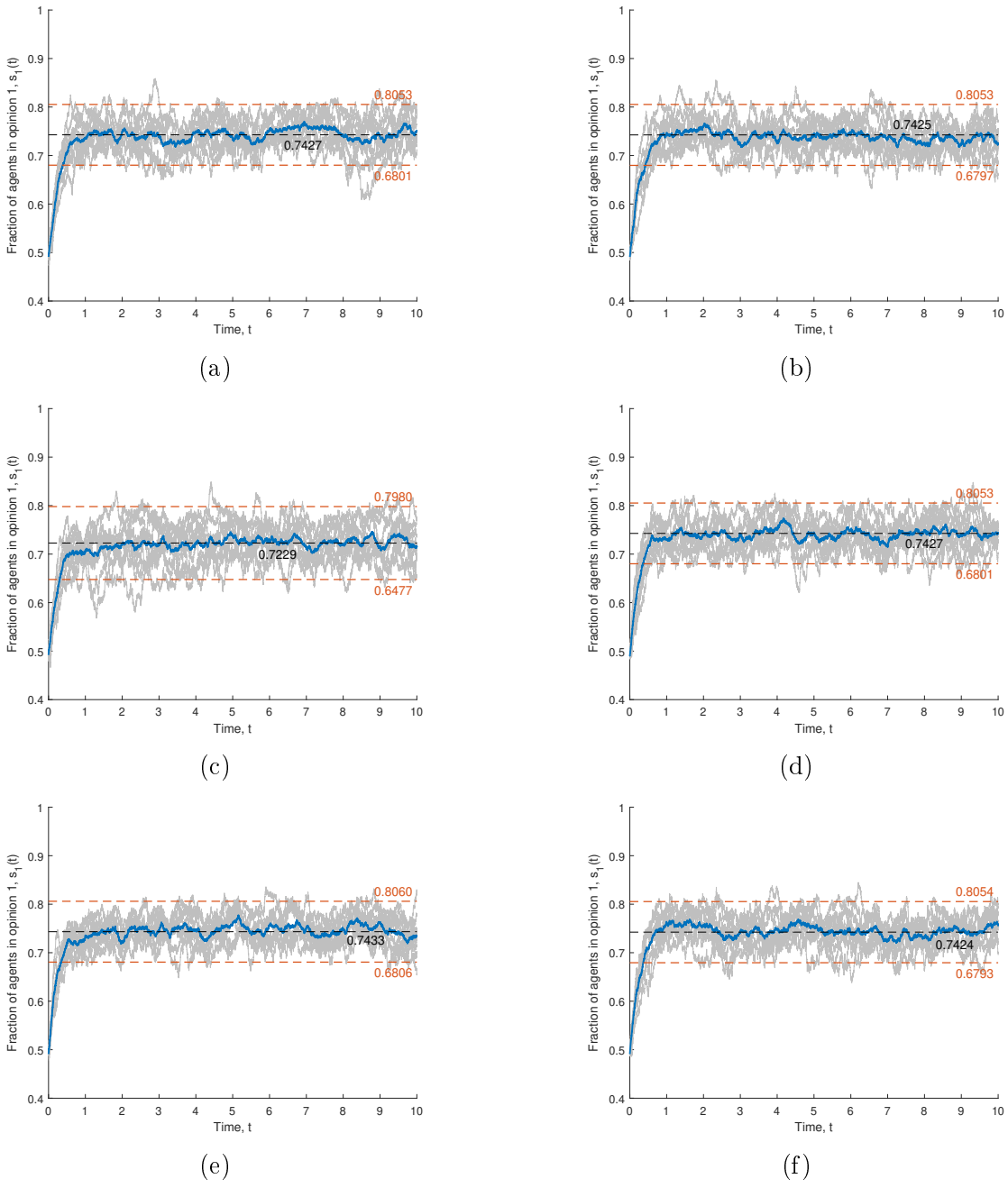


Figure 5.3: Time evolution of the vote share $s_1(t)$ in a Toivonen et al. network using as centrality index: degree (a), betweenness (b), coreness (c), eigenvector centrality (d), strength based on the standard trustiness model (e), and strength based on the degree-weighted normalized trustiness model (f). The gray lines represent the ten single realizations. The blue line is the average of the realizations. The black dashed line corresponds to the theoretical mean vote share $\bar{\mu}_{s_1}$. The orange dashed lines are traced at $\pm 2\bar{\sigma}_{s_1}$.

with a small margin over degree and the two strength centrality indices. In fact, further simulations with the same set of parameters but on different realizations of the underlying LFR network showed that the "best" centrality index varies depending on the network topology. Regardless, degree, betweenness and the two types of strength consistently got very high and very similar results in simulations on LFR networks, while coreness and eigenvector centrality consistently achieved lower results. On the Toivonen et al. network, the results were very close for the whole set of indices, and comparable to those obtained in the LFR counterpart. The best result for the network of Figure 5.3, experimentally and theoretically, came from the strength centrality based on the standard trustiness model, but simulations on different realizations of the network showed that also for the Toivonen et al. network the most effective index depends on the specific topology of the network, and it can be any of the investigated indices with the exception of the coreness centrality which consistently got slightly lower results. The fact that, in both LFR and Toivonen et al. networks, the most effective index depends on the network topology proves that none of the investigated centrality indices is equivalent for every topology to the algorithm $\mathcal{A}$ described in section 5.1.

Figures 5.4a-b display the correlation between the degree and betweenness centrality in the two networks employed in the simulations, which show that the two indices are highly correlated. Similar results can be obtained comparing degree and betweenness with the two types of strength, in both LFR and Toivonen et al. networks, explaining the similarity in the results for the four indices. However, degree, betweenness and strength seem to be less correlated with coreness and eigenvector centrality in LFR networks, as it is apparent from Figures 5.4c-d and Figures 5.4e-f. On the contrary, in Toivonen et al. networks, the eigenvector centrality is highly correlated with betweenness and the rest of the indices, while coreness has high correlation with betweenness only in the upper-right corner of the figure, obtaining results comparable to the other indices only when the number of controlled agents K is not very high. Lastly, as it can be noticed by looking at Figures 5.4c-d, it happens frequently that there are multiple agents who achieve the highest value of coreness, thus it is important to keep in mind that the actual performance of the coreness centrality index, in both LFR and Toivonen et al. networks, depends on the criterion used to select the K agents to control among

those who got the highest coreness score.

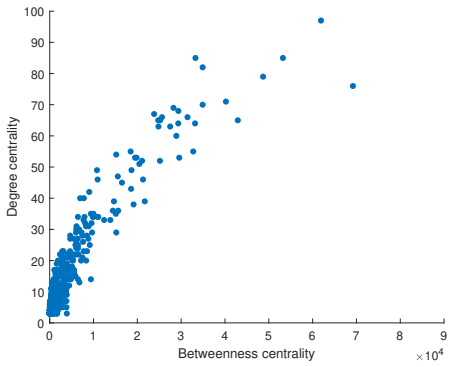
The results suggest that degree, betweenness and the two types of strength are able to grasp some features of the network that are important for the effectiveness of the attack, while eigenvector centrality in LFR networks and coreness centrality in both network topologies seem to pick on less effective network features, consequently obtaining lower results.

5.3 Impact of the number of controlled agents

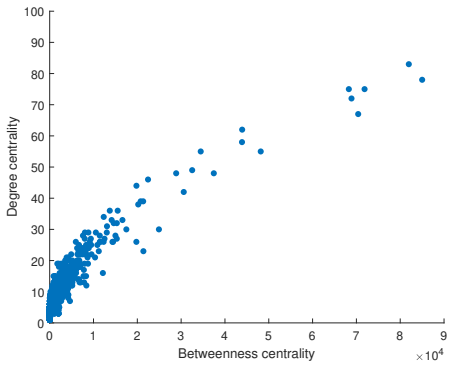
At a first glance, the effect on the attack of varying the number of controlled agents K is relatively straightforward: an increase of K determines a larger number of uncontrolled agents directly influenced via links connected to controlled agents, which ultimately increases the efficacy of the attack.

Figure 5.5 shows the time evolution of the vote share $s_1(t)$ on a LFR network with influenceability $\eta = 10$, degree-weighted normalized trustiness model and betweenness as centrality index, for $K = 10$ and $K = 100$ controlled agents. The same set of simulations but carried out on a Toivonen et al. network can be seen on Figure 5.6. The stochastic social power of the controlled agents and the experimental and theoretical values of the steady-state mean and standard deviation of $s_1(t)$ can be found in Table 5.2. In the LFR network, with $K = 100$, the vote share is about 90% in favour of opinion 1, much higher than its expected value of $\frac{N+K}{2N} = 0.55$ in case of non-interacting agents. From the point of view of the attacker, increasing K also has a beneficial effect on the dispersion of the vote share, which decreases as the number of controlled agents grow.

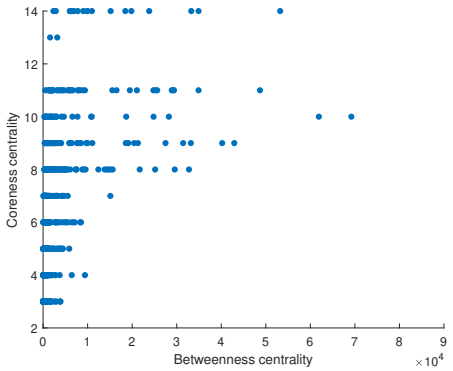
Table 5.3 lists the mean and standard deviation of the vote share resulted from simulations on LFR and Toivonen et al. networks with $K = 100$, $\eta = 10$, degree-weighted normalized trustiness model, and either betweenness or the two types of strength as centrality index. While with $K = 10$ the simulations on the Toivonen et al. network obtained results comparable to the LFR network counterpart (see Table 5.1), when the number of controlled agents is increased to 100 the latter network topology obtains noticeably higher results. It is still an open question the specific reasons why the LFR topology achieves better results for high values of K, but one possible explanation lies in the specific shape of the degree distribution



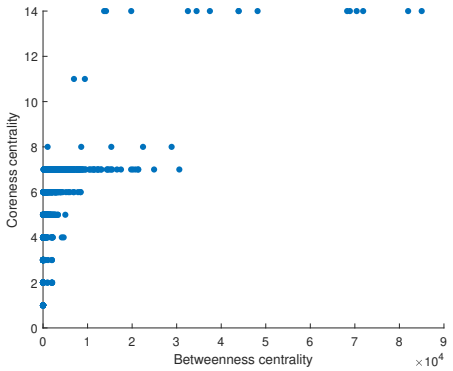
(a)



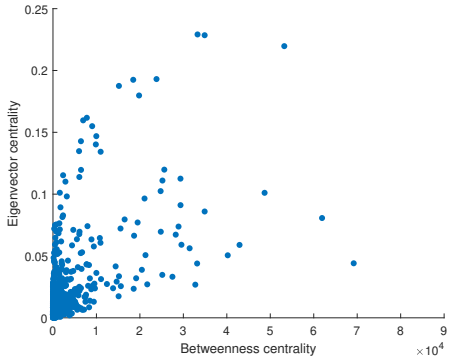
(b)



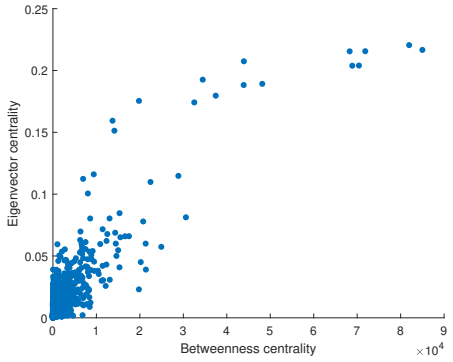
(c)



(d)

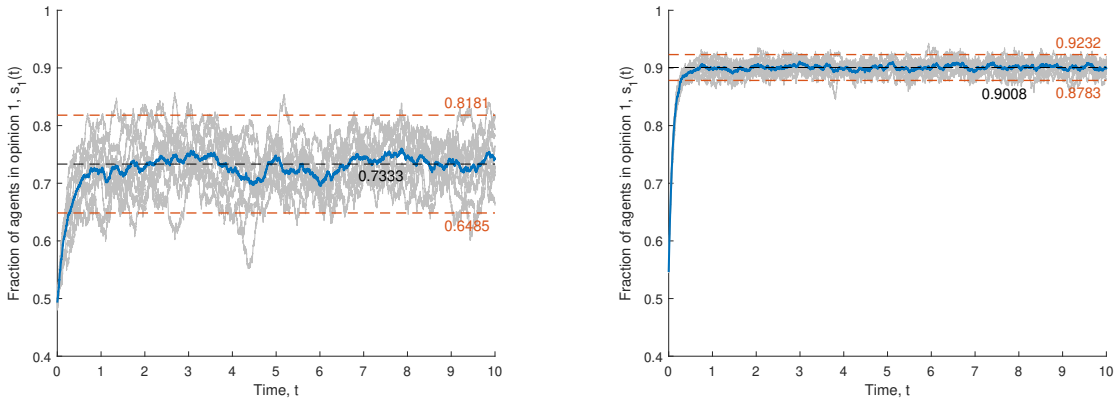


(e)



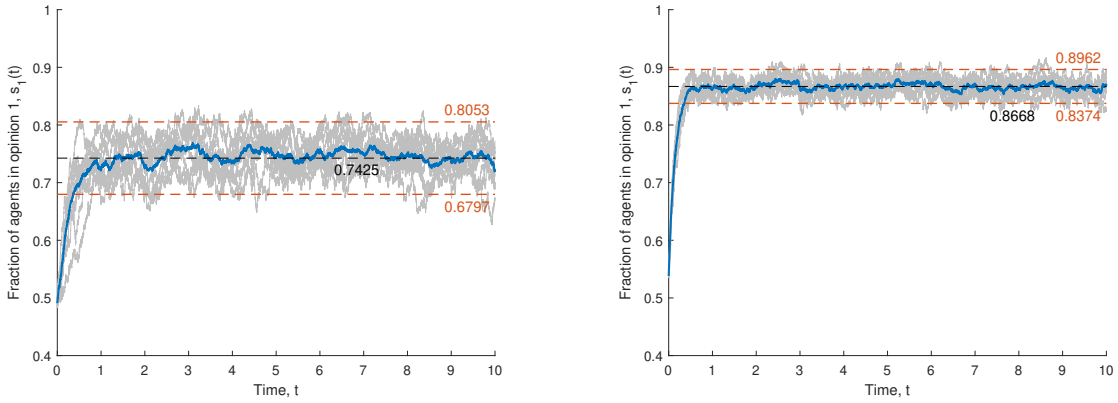
(f)

Figure 5.4: Scatter plots showing the correlation between degree centrality and betweenness centrality (a, b), between coreness centrality and betweenness centrality (c, d), and between eigenvector centrality and betweenness centrality (e, f). Figures on the left (a, c, e) are relative to a LFR network, while figures on the right (b, d, f) are relative to a Toivonen et al. network. Each figure contains all the N agents of the network.



(a) (b)

Figure 5.5: Time evolution of the vote share $s_1(t)$ in a LFR network for $K = 10$ (a) and $K = 100$ (b) controlled agents. The gray lines represent the ten single realizations. The blue line is the average of the realizations. The black dashed line corresponds to the theoretical mean vote share $\bar{\mu}_{s_1}$. The orange dashed lines are traced at $\pm 2\bar{\sigma}_{s_1}$.



(a) (b)

Figure 5.6: Time evolution of the vote share $s_1(t)$ in a Toivonen et al. network for $K = 10$ (a) and $K = 100$ (b) controlled agents. The gray lines represent the ten single realizations. The blue line is the average of the realizations. The black dashed line corresponds to the theoretical mean vote share $\bar{\mu}_{s_1}$. The orange dashed lines are traced at $\pm 2\bar{\sigma}_{s_1}$.

of the two investigated network topologies. For relatively low values of K, the average degree of the controlled agents is similar for the two types of networks, but as K grows the difference increases and the LFR network presents a higher average degree. Table 5.3 also shows that in the LFR network the two types of strength yield the best mean vote share, both experimentally and theoretically, while for $K = 10$ the most effective index on the same network was the betweenness centrality. Thus, the relative efficacy of a given centrality index, compared to the others, depends not only on the network topology as proved in section 5.2, but also on the number K of controlled agents.

It is interesting to note that the first few controlled agents are responsible for a significant portion of the overall effect on the network, while increasing K gives a diminishing return in the efficacy of the attack. Figure 5.7 shows the results of eighty realizations carried out with different sets of parameters. Each line consists of ten simulations, executed with $K \in \{10, 20, ..., 100\}$. With the most "extreme" set of parameters (Toivonen et al. network, degree-weighted normalized model, $\eta = 10$, degree centrality), when $K = 10$, $\hat{\mu}_{s_1}$ goes from the baseline value of 0.5 (i.e. the theoretical value when all agents are homogeneous and unbiased) up to 0.74, while it reaches $\hat{\mu}_{s_1} = 0.8649$ when K is increased to 100. The first 10 selected agents are thus responsible for about 66% of the overall increase of $\hat{\mu}_{s_1}$,

	$\hat{\mu}_{s_1}$	$\bar{\mu}_{s_1}$	$\hat{\sigma}_{s_1}$	$\bar{\sigma}_{s_1}$	ψ_{CA}
$K = 10$	0.7331	0.7333	0.0427	0.0424	0.4665
$K = 100$	0.9010	0.9008	0.0113	0.0112	0.8015

(a)

	$\hat{\mu}_{s_1}$	$\bar{\mu}_{s_1}$	$\hat{\sigma}_{s_1}$	$\bar{\sigma}_{s_1}$	ψ_{CA}
$K = 10$	0.7443	0.7425	0.0307	0.0314	0.4850
$K = 100$	0.8667	0.8668	0.0146	0.0147	0.7337

(b)

Table 5.2: Experimental mean $\hat{\mu}_{s_1}$ and standard deviation $\hat{\sigma}_{s_1}$ of the vote share and their theoretical steady-state values $\bar{\mu}_{s_1}$ and $\bar{\sigma}_{s_1}$, for $K = 10$ and $K = 100$ controlled agents, in a LFR network (a) and a Toivonen et al. network (b). The last column of the table contains the stochastic social power ψ_{CA} of the controlled agents.

	$\hat{\mu}_{s_1}$	$\bar{\mu}_{s_1}$	$\hat{\sigma}_{s_1}$	$\bar{\sigma}_{s_1}$	ψ_{CA}
betweenness	0.9004	0.9008	0.0111	0.0112	0.8015
strength (std)	0.9020	0.9027	0.0111	0.0109	0.8053
strength (dwn)	0.9033	0.9033	0.0109	0.0108	0.8066

(a)

	$\hat{\mu}_{s_1}$	$\bar{\mu}_{s_1}$	$\hat{\sigma}_{s_1}$	$\bar{\sigma}_{s_1}$	ψ_{CA}
betweenness	0.8666	0.8668	0.0144	0.0147	0.7337
strength (std)	0.8707	0.8705	0.0146	0.0142	0.7409
strength (dwn)	0.8728	0.8724	0.0142	0.0140	0.7449

(b)

Table 5.3: Experimental mean $\hat{\mu}_{s_1}$ and standard deviation $\hat{\sigma}_{s_1}$ of the vote share and their theoretical steady-state values $\bar{\mu}_{s_1}$ and $\bar{\sigma}_{s_1}$, for different centrality indices and with $K = 100$ controlled agents, in a LFR network (a) and a Toivonen et al. network (b). The last column of the table contains the stochastic social power ψ_{CA} of the controlled agents.

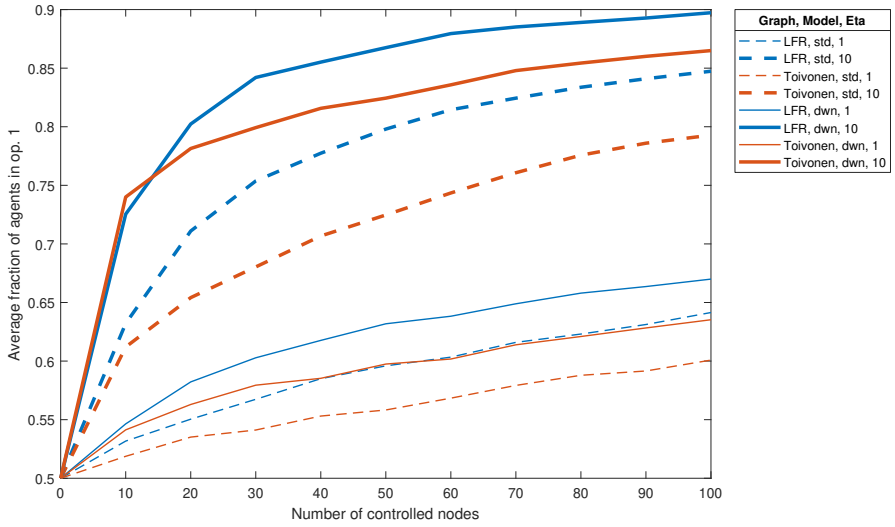


Figure 5.7: Average value $\hat{\mu}_{s_1}$ of the vote share for different values of K controlled agents. Each line corresponds to ten realizations executed with the same set of parameters. A blue (respectively, orange) line is used for the LFR (Toivonen et al.) network. A dashed (solid) line is used for the standard (degree-weighted normalized) trustiness model. A thick (thin) line is used for influenceability of the uncontrolled agents $\eta = 10$ ($\eta = 1$). The employed agent selection criterion is degree centrality for each of the simulations in the figure.

while constituting only 10% of all the controlled agents in the network and being responsible for about 26% of all the controlled links (a link is *controlled* if at least one of the nodes it connects to is itself controlled). These results suggest that, in general, the cost of opinion manipulation increases with the polarization of the network. Being able to control 10% of the agents in a network (e.g. $K = 100$ if $N = 1000$) is most-likely unfeasible in large scale systems such as *Facebook* or *Twitter*, but it is apparent that in order to have a sensible effect on the opinion distribution of the network it is not necessary to take control of such a large portion of the agents.

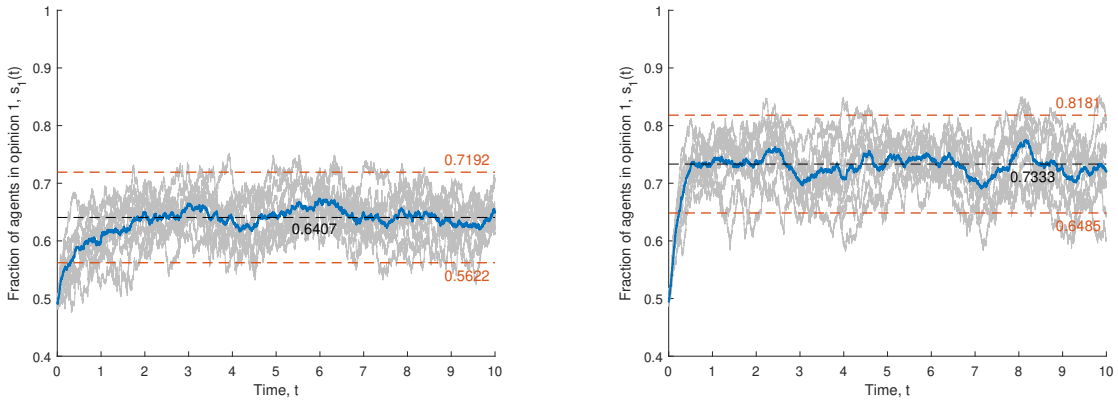
5.4 Impact of the agents' trustiness

In the standard trustiness model, from the point of view of a given agent, the same importance is assigned to each of the agent's neighbours. On the contrary, in the degree-weighted normalized trustiness model, an agent importance is proportional to his/her degree, i.e. how many people he/she connects with. For this reason, the degree-weighted normalized model synergize better than its counterpart with attacks based on either degree centrality or degree-correlated centrality indices.

Figure 5.8 and Figure 5.9 show the time evolution of the vote share $s_1(t)$ on a LFR and a Toivonen et al. network, respectively, with influenceability $\eta = 10$ and with $K = 10$ controlled agents selected according to betweenness centrality, in both the standard and degree-weighted normalized models. Table 5.4 lists the stochastic social power of the controlled agents in the two models, along with the mean and standard deviation of the vote share computed both from the simulations and from the theoretical analysis of section 2.3.

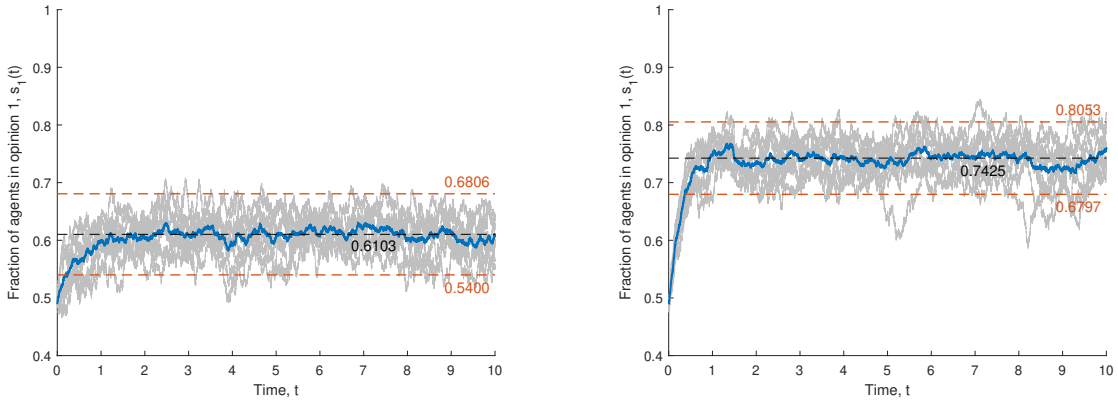
5.5 Impact of the agents' influenceability

Increasing the influenceability η of the uncontrolled agents intensify the susceptibility of the agents to the influence of the neighbours, making the social network weaker to opinion manipulation attacks. However, a greater influenceability makes the agents more prone to opinion changes, reducing the average time interval be-



(a) (b)

Figure 5.8: Time evolution of the vote share $s_1(t)$ in a LFR network for the *standard* trustiness model (a) and *degree-weighted normalized* trustiness model (b). The gray lines represent the ten single realizations. The blue line is the average of the realizations. The black dashed line corresponds to the theoretical mean vote share $\bar{\mu}_{s_1}$. The orange dashed lines are traced at $\pm 2\bar{\sigma}_{s_1}$.



(a) (b)

Figure 5.9: Time evolution of the vote share $s_1(t)$ in a Toivonen et al. network for the *standard* trustiness model (a) and *degree-weighted normalized* trustiness model (b). The gray lines represent the ten single realizations. The blue line is the average of the realizations. The black dashed line corresponds to the theoretical mean vote share $\bar{\mu}_{s_1}$. The orange dashed lines are traced at $\pm 2\bar{\sigma}_{s_1}$.

tween them and increasing the variance of the vote share, as it is apparent from Figure 5.10, relative to a LFR network with $K = 10$ controlled agents selected using betweenness as centrality index, and degree-weighted normalized trustiness model. The same happens in the Toivonen et al. network of Figure 5.11. As usual, Table 5.5 contains the values of the mean and standard deviation of the vote share $s_1(t)$ and the stochastic social power of the controlled agents. The results suggest that an opinion manipulation attack benefits from targeting networks and communities with people that are, in general, more susceptible to the influence of others.

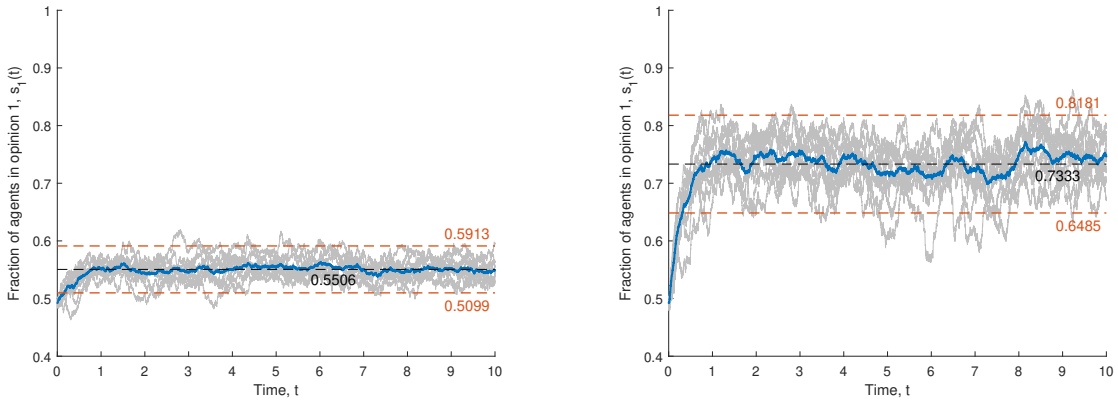
	$\hat{\mu}_{s_1}$	$\bar{\mu}_{s_1}$	$\hat{\sigma}_{s_1}$	$\bar{\sigma}_{s_1}$	ψ_{CA}
"std" model	0.6417	0.6407	0.0392	0.0393	0.2813
"dwn" model	0.7301	0.7333	0.0432	0.0424	0.4665

(a)

	$\hat{\mu}_{s_1}$	$\bar{\mu}_{s_1}$	$\hat{\sigma}_{s_1}$	$\bar{\sigma}_{s_1}$	ψ_{CA}
"std" model	0.6115	0.6103	0.0344	0.0352	0.2206
"dwn" model	0.7423	0.7425	0.0320	0.0314	0.4850

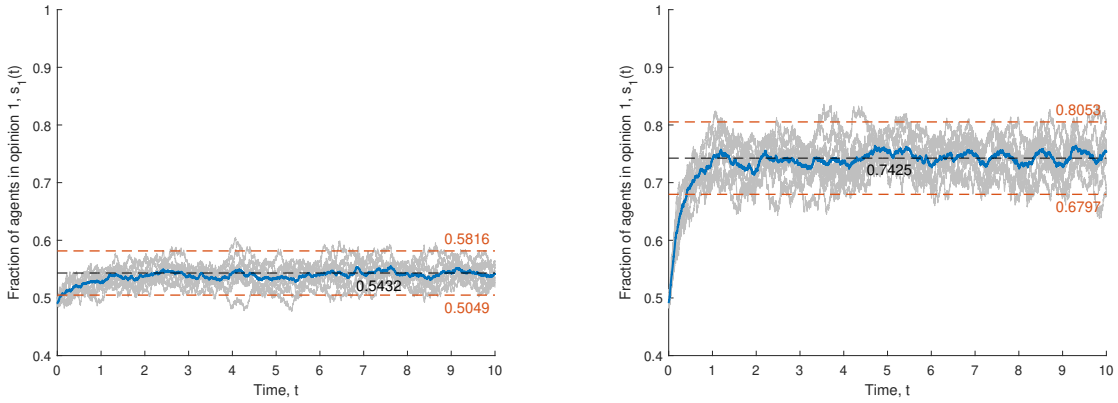
(b)

Table 5.4: Experimental mean $\hat{\mu}_{s_1}$ and standard deviation $\hat{\sigma}_{s_1}$ of the vote share and their theoretical steady-state values $\bar{\mu}_{s_1}$ and $\bar{\sigma}_{s_1}$, for the two different trustiness models, in a LFR network (a) and a Toivonen et al. network (b). The last column of the table contains the stochastic social power ψ_{CA} of the controlled agents.



(a) (b)

Figure 5.10: Time evolution of the vote share $s_1(t)$ in a LFR network for influenceability $\eta = 1$ (a) and $\eta = 10$ (b). The gray lines represent the ten single realizations. The blue line is the average of the realizations. The black dashed line corresponds to the theoretical mean vote share $\bar{\mu}_{s_1}$. The orange dashed lines are traced at $\pm 2\bar{\sigma}_{s_1}$.



(a) (b)

Figure 5.11: Time evolution of the vote share $s_1(t)$ in a Toivonen et al. network for influenceability $\eta = 1$ (a) and $\eta = 10$ (b). The gray lines represent the ten single realizations. The blue line is the average of the realizations. The black dashed line corresponds to the theoretical mean vote share $\bar{\mu}_{s_1}$. The orange dashed lines are traced at $\pm 2\bar{\sigma}_{s_1}$.

	$\hat{\mu}_{s_1}$	$\bar{\mu}_{s_1}$	$\hat{\sigma}_{s_1}$	$\bar{\sigma}_{s_1}$	ψ_{CA}
$\eta = 1$	0.5512	0.5506	0.0204	0.0203	0.1011
$\eta = 10$	0.7345	0.7333	0.0412	0.0424	0.4665

(a)

	$\hat{\mu}_{s_1}$	$\bar{\mu}_{s_1}$	$\hat{\sigma}_{s_1}$	$\bar{\sigma}_{s_1}$	ψ_{CA}
$\eta = 1$	0.5425	0.5432	0.0189	0.0192	0.0865
$\eta = 10$	0.7420	0.7425	0.0309	0.0314	0.4850

(b)

Table 5.5: Experimental mean $\hat{\mu}_{s_1}$ and standard deviation $\hat{\sigma}_{s_1}$ of the vote share and their theoretical steady-state values $\bar{\mu}_{s_1}$ and $\bar{\sigma}_{s_1}$, for influenceability $\eta = 1$ and $\eta = 10$, in a LFR network (a) and a Toivonen et al. network (b). The last column of the table contains the stochastic social power ψ_{CA} of the controlled agents.

5.6 Communities

Intra-community association

The generic element α_{rs} of an *agreement matrix* corresponds to the association between agents r and s, measured as the fraction of the total simulation time in which agents r and s had the same opinion. It can be easily obtained from the contingency table $Y^{[rs]}$ of agents r and s, by computing $\alpha_{rs} = \frac{1}{T}\sum_j y_{jj}^{[rs]}$, $j \in \mathcal{M}$, where T is the total simulation time. The calculation of the contingency tables of the agents has been described in section 3.2.

Figure 5.12 shows the agreement matrices of two different simulations carried out on a LFR network, for $K = 0$ and $K = 10$ controlled agents. The identification of the communities has been executed using Louvain algorithm, described in section 4.2. Without controlled agents ($K = 0$), people belonging to a given community present a slightly higher association, measured with the simple agreement index, with respect to the association measured between agents belonging to different communities. When the network is under attack, the association between agents belonging to communities containing at least one controlled agent noticeably increases, while agents belonging to the rest of the network have just a slight increase in association.

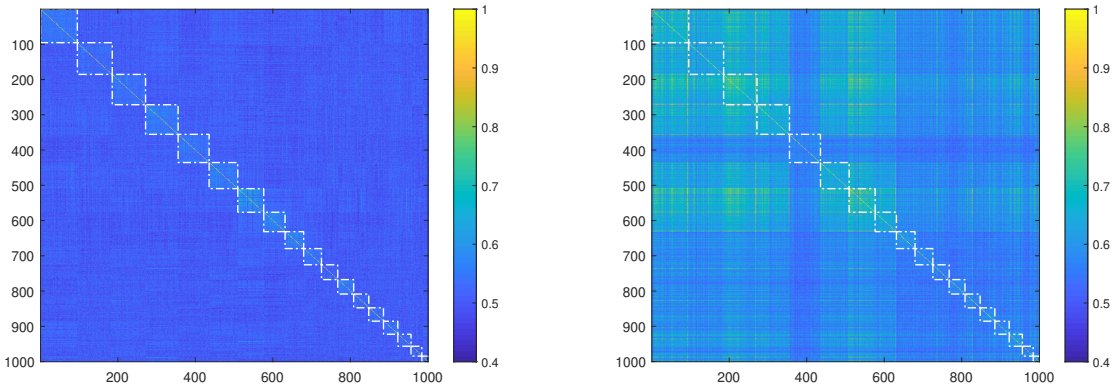
Attacks on networks without a community structure

Section 4.2 briefly mentioned the assumption that randomly wired networks lack a community structure. It is thus possible to apply a "degree-preserving randomization" procedure to a network in order to break its community structure while preserving its degree distribution. Simulations carried out on "rewired" LFR networks, i.e. LFR networks which underwent this procedure, showed slightly better results in most of the cases with respect to those performed on standard LFR networks and the same set of parameters. For example, attacks on a set of "rewired" LFR networks using betweenness centrality, influenceability $\eta = 10$, and degree-weighted normalized trustiness model, achieved on average $\hat{\mu}_{s_1} = 0.7554$ and $\hat{\sigma}_{s_1} = 0.0418$ for $K = 10$ controlled agents, while $\hat{\mu}_{s_1} = 0.9062$ and $\hat{\sigma}_{s_1} = 0.0103$ for $K = 100$ controlled agents. The average modularity of the community partition obtained

by applying Louvain algorithm to this set of networks is $M = 0.2753$. For comparison, simulations on the same set of parameters for a standard LFR network (see Table 5.2) resulted in $\bar{\mu}_{s_1} = 0.7333$ and $\bar{\sigma}_{s_1} = 0.0424$ for $K = 10$ controlled agents, and $\bar{\mu}_{s_1} = 0.9008$ and $\bar{\sigma}_{s_1} = 0.0112$ for $K = 100$ controlled agents. These results suggest that the presence of a community structure in the network may slightly reduce the efficacy of a given attack.

Targeting a single community

If the target of an attack is not the whole network but a single community, for a LFR network an effective strategy is to select all K agents to be controlled from the target community. Figure 5.13 shows the mean vote share of each community obtained from simulations carried out on a LFR network targeting the biggest community with $K = 10$ controlled agents, and from simulations of a "standard" attack with the same set of parameters ($K = 10$, $\eta = 10$, degree-weighted normalized model, and betweenness centrality) but with the controlled agents not



(a) (b)

Figure 5.12: Colormaps of the agreement matrices of a LFR network with $K = 0$ controlled agents (a) and $K = 10$ controlled agents (b). Rows and columns have been sorted in order to have agents grouped by communities identified with Louvain algorithm. The white dash-dotted lines help in highlighting them. The communities containing at least one controlled agent are the first eight starting from the upper-left corner, excluding the fifth one. The first community from the top contains two controlled agents, the third community contains three, the other communities contain one controlled agent each.

restrained in belonging to a specific community. The overall performance of the

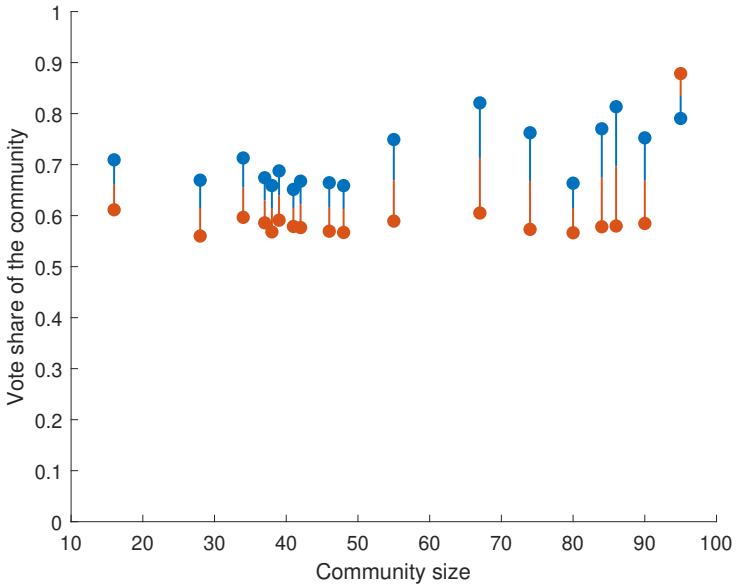


Figure 5.13: Scatter plot of the mean vote share of each community in a LFR network, versus the number of agents belonging to the community. Blue dots correspond to the mean vote share of the communities in a "standard" attack, i.e. an attack where the $K = 10$ controlled agents are not restrained in belonging to a specific community and are selected according to the highest values of betweenness centrality in the whole network. Orange dots correspond to the mean vote share of the communities in a "targeted" attack, i.e. an attack where the $K = 10$ controlled agents are the ones belonging to the target community and with the highest betweenness score. The target community is the rightmost one in the figure.

"targeted" attack was expectedly lower ($\hat{\mu}_{s_1} = 0.6086$, $\bar{\mu}_{s_1} = 0.6137$) compared to the "standard" attack, but with a mean vote share equal to 0.8785 for the agents inside the target community. The remaining 17 communities had an average mean vote share equal to 0.5812 and standard deviation of the mean vote share equal to 0.0142. The "standard" attack resulted in $\hat{\mu}_{s_1} = 0.7313$, $\bar{\mu}_{s_1} = 0.7333$, mean vote share of the biggest community equal to 0.7905, average mean vote share of the remaining communities equal to 0.7110, and standard deviation of the mean vote share of the remaining communities equal to 0.0564.

Interestingly, it seems that this strategy does not always work in the Toivonen et al. network with the usual network parameters. In fact, simulations carried out

on different realizations of the network showed contrasting outcomes. For example, these are the results of three sets of twenty simulations each (ten "targeted" attacks, ten "standard" attacks) carried out on three realizations of a Toivonen et al. network:

1. "Targeted" attack: mean vote share of the target community on average equal to 0.7499. "Standard" attack: mean vote share of the target community on average equal to 0.7821, and 1 of the 10 controlled agents belonged to the target community;

2. "Targeted" attack: mean vote share of the target community on average equal to 0.8072. "Standard" attack: mean vote share of the target community on average equal to 0.8062, and 9 of the 10 controlled agents belonged to the target community;

3. "Targeted" attack: mean vote share of the target community on average equal to 0.8106. "Standard" attack: mean vote share of the target community on average equal to 0.7633, and 5 of the 10 controlled agents belonged to the target community.

Chapter 6

Estimation of the Attack Severity

This chapter introduces a closed-loop control action that can be used by the network manager for counteracting the change in the average opinion of the network introduced by an opinion manipulation attack, and for assessing the severity of the attack through the measure of the interaction intensity parameter needed to restore a neutral average opinion. After the presentation in section 6.2 of some results relative to different values for the number of controlled agents and the centrality index employed, section 6.3 illustrates some of the effects on the communities introduced by the rebalancing action exerted by the network manager.

6.1 Introduction

From the point of view of the network manager, the "magnitude" of an opinion manipulation attack can be assessed by estimating the variation of the interaction intensity parameter λ_j, $j \in \mathcal{M}$, needed for rebalancing the average opinion of the network. In the specific case of $M = 2$ opinions, assuming that the network is manipulated as described in chapter 5 in order to foster opinion 1, the objective of the platform manager is to estimate how much λ_1 should be lowered for restoring, at steady-state, the mean of the vote share $s_1(t)$ to the value it had prior to the attack, i.e. $\mu_{s_1,ref} = 0.5$.

The stochastic process $s_1(t)$ with mean μ_{s_1} and variance $\sigma^2_{s_1}$ can be treated (at steady-state) as a constant signal with value μ_{s_1}, with a superimposed "distur-

bance" whose variance is equal to $\sigma_{s_1}^2$. Thus, the estimation of the desired value of λ_1 can be achieved by controlling this process with a closed-loop control system, suitably tuned so that it is slow enough to ignore the relatively faster dynamics of the "disturbance" while tracking the dynamics of μ_{s_1}.

6.2 Closed-loop control

Figure 6.1 shows the diagram of the closed-loop control system employed for the estimation of λ_1. The reference is given by the desired mean vote share $\mu_{s_1,ref}$, the output variable is the actual vote share s_1, and the control variable is the interaction intensity parameter λ_1.

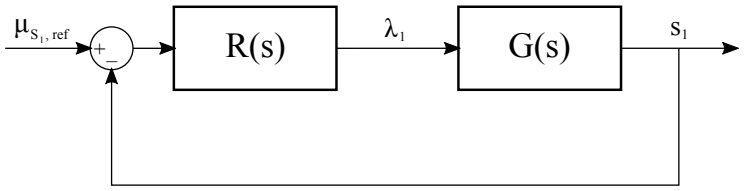


Figure 6.1: Closed-loop system used for estimating the steady-state value of λ_1 needed for rebalancing the average opinion in an attacked network.

The transfer function $G(s)$ from the interaction intensity parameter λ_1 to the vote share s_1 can be approximated as a first order system:

$$G(s) = \frac{\rho}{1 + s\tau}.$$

The gain ρ and the time constant τ of the first order approximation can be estimated through the open-loop response of the system to a step applied to the interaction parameter λ_1. The parameters of $G(s)$ employed in the simulations of this chapter are $\rho = 0.6767$ and $\tau = 0.1762$.

The controller $R(s)$ is a linear Proportional-Integral controller, with transfer function:

$$R(s) = k_p + \frac{k_i}{s}.$$

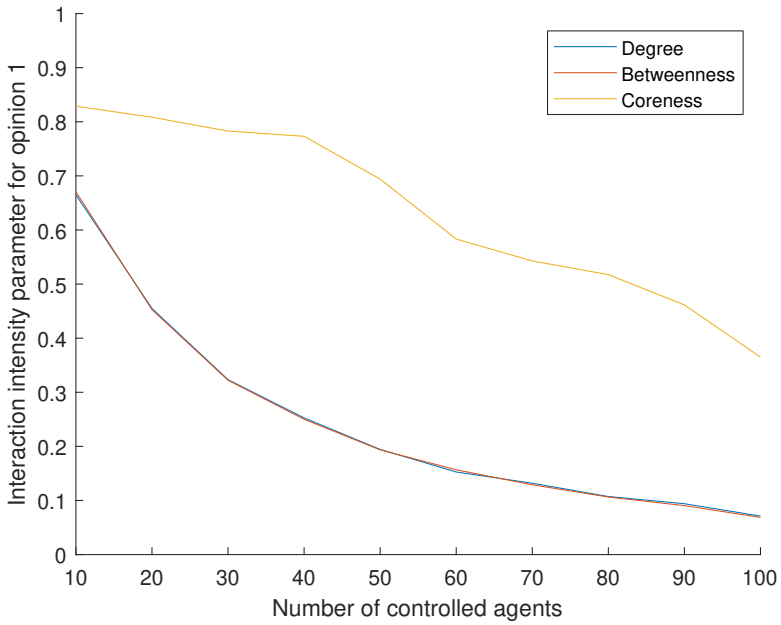
The integral action in the controller is useful for nullifying the tracking error at steady-state. The proportional and integral constants employed in the simulations

are $k_p = 0.0220$ and $k_i = 0.1249$, corresponding to a cut-off frequency of the closed-loop system equal to $\omega_c = 0.0845$ and phase margin $\varphi_m = \frac{\pi}{2}$.

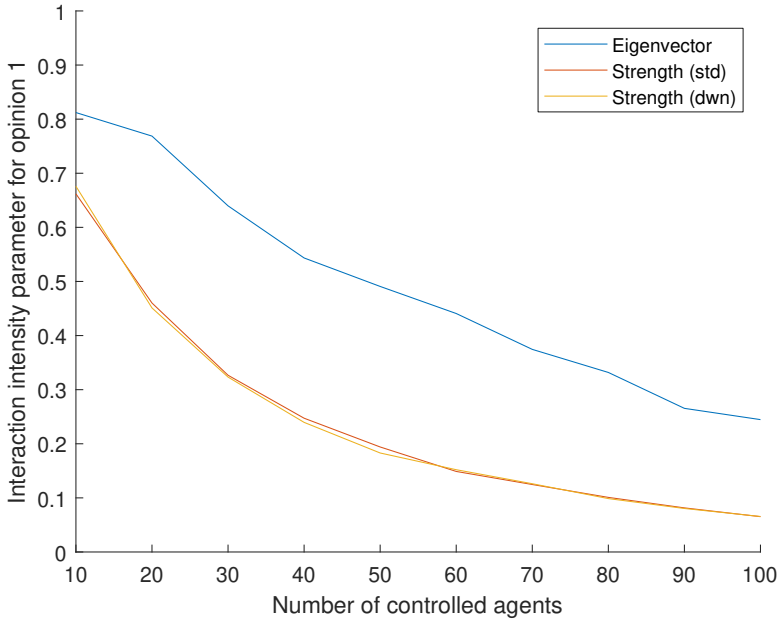
Although the control system has been designed in the continuous-time domain, Gillespie algorithm simulates the system with variable-length discrete-time steps. Due to this limitation, the control variable λ_1 cannot be varied continuously by the controller, and thus, in the following simulations, its value has been updated in correspondence of state-changing events (i.e. a change of opinion) occurred in the system. However, the error introduced by the approximation is acceptable in this setup because, due to the low speed of the control system, the change of the control variable λ_1 between two consecutive events is negligible.

Figure 6.2 and Figure 6.3 show the steady-state value of the interaction intensity parameter λ_1 needed for rebalancing the average opinion in a LFR network and a Toivonen et al. network, respectively, with influenceability $\eta = 10$, degree-weighted normalized trustiness model, and $K \in \{10, 20, ..., 100\}$ controlled agents selected according to the various centrality indices investigated in section 5.2. The interaction intensity parameter for opinion 2 is constant and set to $\lambda_2 = 1$. As an example, Figure 6.4 shows the time evolution of the vote share s_1 and the control variable λ_1 in a LFR network for the two extreme cases of $K = 10$ and $K = 100$ controlled agents selected using betweenness centrality.

The results are in agreement with the findings of chapter 5. Degree, Betweenness and the two types of strength are highly correlated in both the network topologies, and thus require very similar values of λ_1 for rebalancing the average opinion of the network. The noticeably higher values of λ_1 (meaning that a lower effort is required to the network manager for rebalancing the network) for coreness and eigenvector centrality demonstrate that the two indices cannot quite keep up with the other ones in the Toivonen et al. network, and the results for the LFR network are even worse. Lastly, the values of λ_1 for $K = 10$ controlled agents are very similar for the two network topologies, around $0.62 \div 0.68$ for degree and correlated indices, but as K increases the LFR network requires lower values of λ_1 compared to its counterpart, reaching $\lambda_1 \approx 0.06$ for $K = 100$ against $\lambda_1 \approx 0.22$ in the Toivonen et al. network. It is interesting to note that it is possible to counteract with a $\lambda_1 > 0$ even an attack with a mean vote share as high as 0.9, such as the one depicted in Figure 6.4. However, if the number of controlled agents

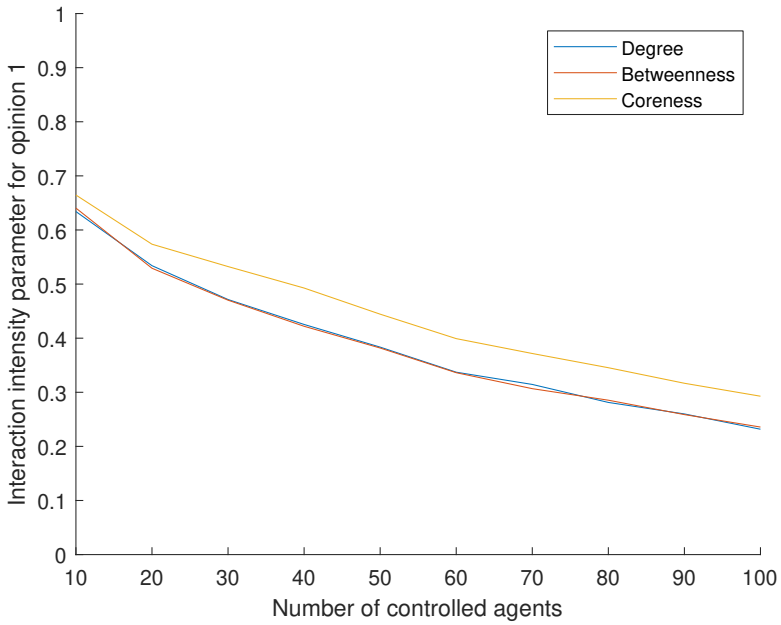


(a)

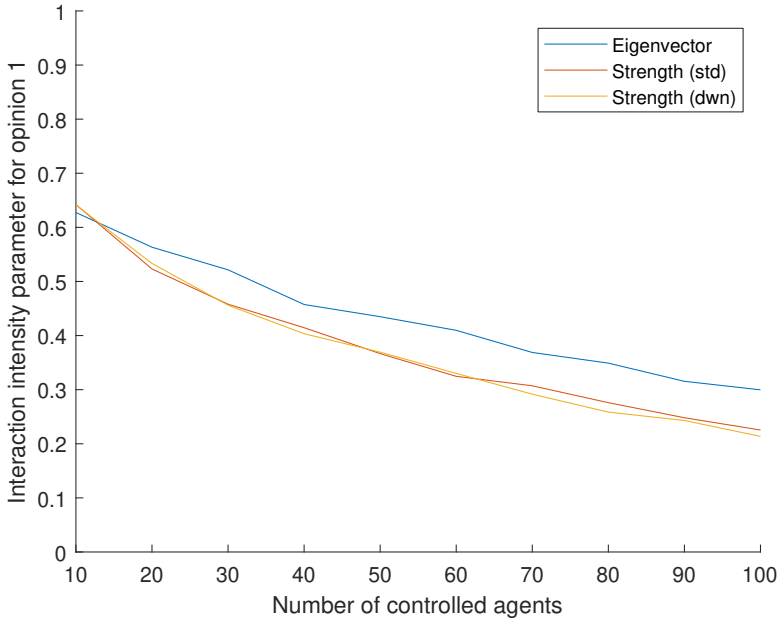


(b)

Figure 6.2: Steady-state value of the interaction intensity parameter λ_1 in a LFR network for $K \in \{10, 20, ..., 100\}$ controlled agents selected using as centrality index: degree, betweenness, and coreness in (a), eigenvector, and the two types of strength in (b).



(a)



(b)

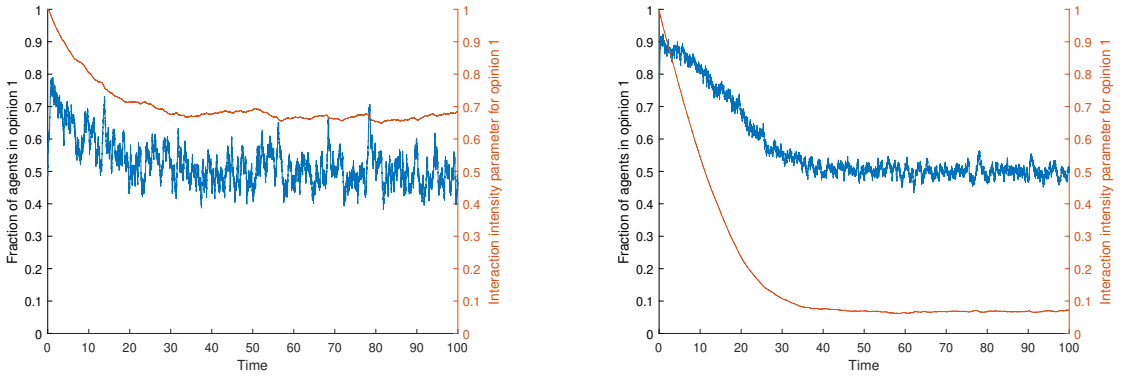
Figure 6.3: Steady-state value of the interaction intensity parameter λ_1 in a Toivonen et al. network for $K \in \{10, 20, ..., 100\}$ controlled agents selected using as centrality index: degree, betweenness, and coreness in (a), eigenvector, and the two types of strength in (b).

increases too much, the control variable λ_1 will saturate to zero, meaning that in some cases it may not be possible to rebalance the average opinion of the network by simply tuning the interaction intensity parameter λ_1.

6.3 Effect on communities

The previous section showed that, in most of the cases, it is feasible to rebalance the steady-state average opinion of a social network by suitably tuning the interaction intensity parameter λ_1. However, that does not happen for the average opinion of the single communities in the network. During an opinion manipulation attack, the communities containing the controlled agents are the ones which end up to be polarized the most towards opinion 1. Varying the interaction intensity parameter λ_1 however has a global effect on the opinion distribution of the network, affecting all the communities.

Figure 6.5 shows the steady-state value of the mean vote share of each community of a LFR network, before and after the rebalancing action applied by the network manager. The parameters of the simulations are: $K = 10$ controlled agents selected using betweenness centrality, influenceability $\eta = 10$, and degree-weighted normalized trustiness model. The identification of the communities is carried out using Louvain algorithm, described in section 4.2. As it can be seen



(a) (b)

Figure 6.4: Time evolution of the vote share s_1 (blue line) and the interaction intensity parameter λ_1 (orange line), in a LFR network with $K = 10$ (a) and $K = 100$ (b) controlled agents selected using betweenness centrality.

from the figure, after the rebalancing action the average opinion of the single communities is often far from the ideal value of 0.5. Two communities have a mean vote share greater than 0.65, while 8 out of the 18 total communities have a mean vote share in the range $0.37 \div 0.4$, meaning that they are polarized towards opinion 2, even if there is not a single agent in the whole network with an individual bias towards opinion 2. Simulations with different values of K show that the magnitude of this community polarization effect increases as the number of controlled agents grows.

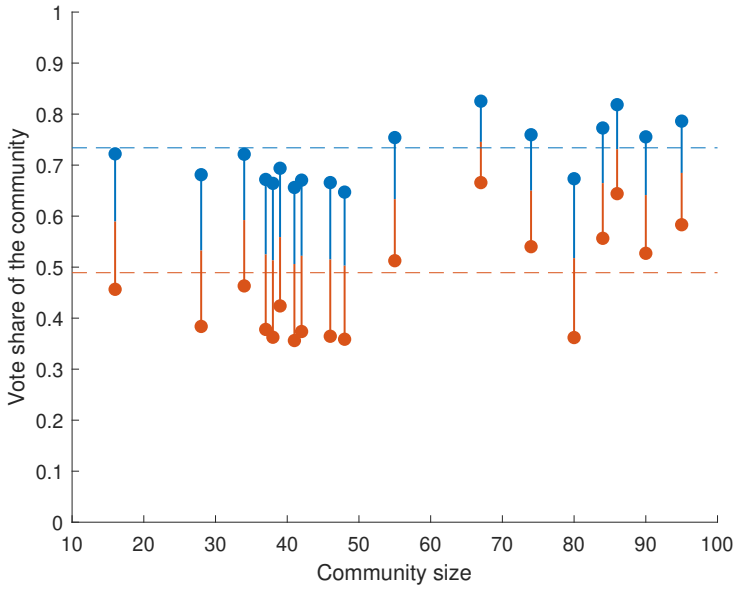


Figure 6.5: Scatter plot of the mean vote share of each community in a LFR network, versus the number of agents belonging to the community. Blue (respectively, orange) dots correspond to the mean vote share of the communities before (after) the rebalancing action exerted by the network manager. A dashed line of the corresponding color indicates the mean vote share of the overall network.

Figure 6.6 shows the agreement matrices, described in section 5.6, for the same LFR network used in the simulations of Figure 6.5, before the attack ($K = 0$), during the attack but before the rebalancing action ($K = 10$ and $\lambda_1 = 1$), and after the rebalancing action ($K = 10$ and λ_1 varied according to the control described in section 6.2). The third matrix of Figure 6.6 clearly shows the reduction in association, compared with the perfectly balanced case of the non-attacked network, between the communities containing controlled agents (i.e. the seven above the 0.5

orange line in Figure 6.5) and the rest of the communities.

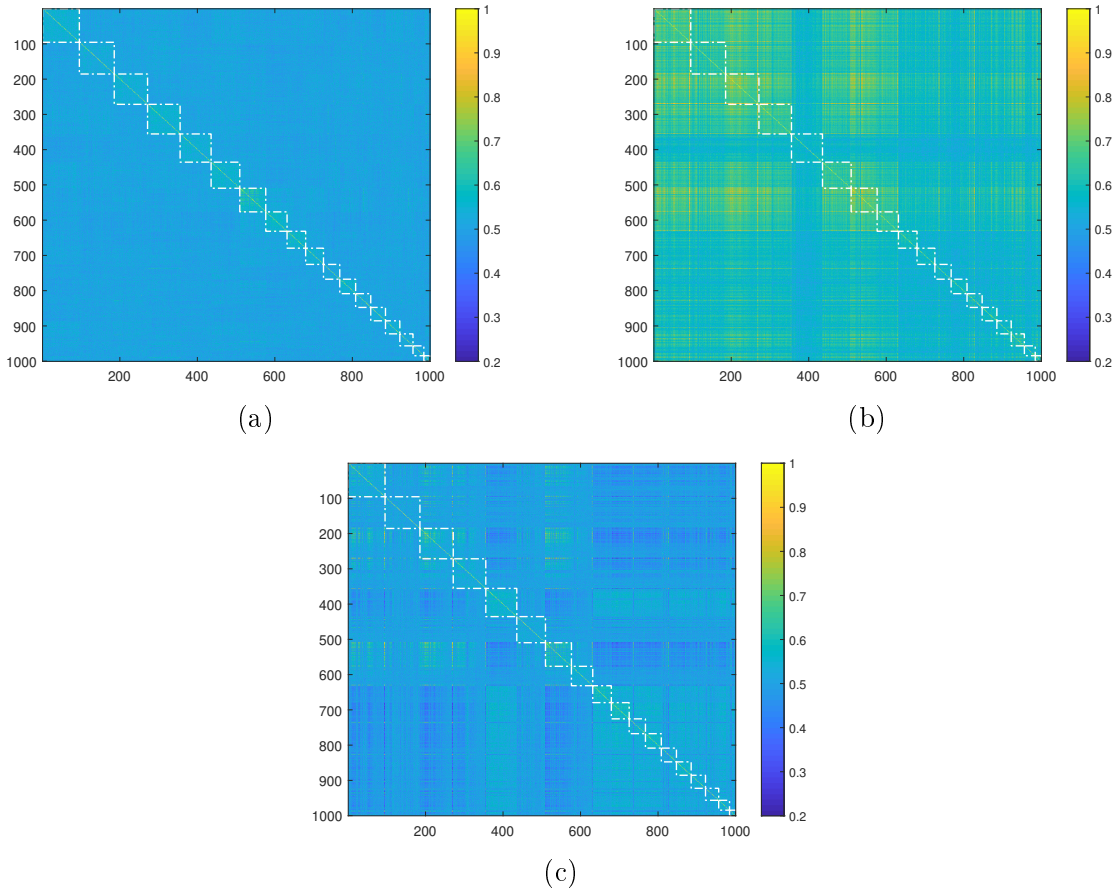


Figure 6.6: Colormaps of the agreement matrices of a LFR network before the attack (a), during the attack but before the rebalancing action (b), and after the rebalancing action (c). Rows and columns have been sorted in order to have agents grouped by communities identified with Louvain algorithm. The white dash-dotted lines help in highlighting them. The communities containing at least one controlled agent are the first eight starting from the upper-left corner, excluding the fifth one. The first community from the top contains two controlled agents, the third community contains three, the other communities contain one controlled agent each.

Chapter 7

Conclusions

The main results achieved in this thesis are:

- None of the investigated centrality indices is capable of achieving the optimal value of the steady-state mean vote share in every network topology. The simulations of section 5.2 showed that the most effective index depends on the specific network topology, thus excluding that one of them could be a perfect implementation of the optimal selection criterion. Nonetheless, centrality indices like degree, betweenness, and strength consistently proved to be very effective in the limited framework of the executed simulations, and demonstrated to be very correlated to one another in the investigated network topologies. The two types of strength employed in this thesis are both based on link-weights completely determined by the network topology, which in the case of online social networks is often very easily identifiable by an attacker through the use of suitable crawlers. The eigenvector centrality index achieved lower results in one of the two investigated network topologies, while the coreness centrality index could not stand the competition with the rest of the indices in both the topologies. The specific reasons why degree, betweenness and strength achieved the highest results, while coreness and eigenvector centrality proved to be less effective, are still an open question.
- A content-filtering action exerted by the platform manager could intensify the opinion polarization between the communities of a network. The simulations of section 6.2 proved that this filtering action is capable of restoring the

overall average opinion of the network even in the case of an attack where a significant portion of the network has been controlled. However, section 6.3 showed that this does not happen for the opinion distribution of the single communities, where the ones containing controlled agents remained more or less polarized towards the fostered opinion, while the rest of the communities shifted towards the opposite opinion. This phenomenon could explain the strong polarization that can be frequently observed in real online social networks. Moreover, the results raise concerns about the possibility that a platform manager could employ the content-filtering action for pushing specific political or economical agendas.

- Selecting all the controlled agents from a single community may not be the most effective strategy for manipulating the average opinion of that community. Simulations carried out in section 5.6 showed that, for some network topologies, "standard" attacks selecting the most influential agents in the whole network and not restrained in belonging to a specific community sometimes achieve a higher steady-state mean vote share for the target community, compared to "targeted" attacks selecting all the agents from the target community.

- A large part of the total effectiveness of an opinion manipulation attack is given by the first few selected nodes. Section 5.3 demonstrated that the number of controlled agents is a very important attack parameter, predictably showing an increase in efficacy of the attack as the number of stubborn spreaders grows. Moreover, the simulations proved that the first few agents with the highest centrality scores are responsible for a substantial portion of the total effectiveness of an attack, suggesting that in order to manipulate the average opinion of the agents it is not necessary to take control of a very large section of the network, and that the cost of the attack may increase with the polarization of the network. Lastly, the simulations showed that the relative efficacy of the centrality indices depends not only on the specific network topology, but also on the number of controlled agents.

- Opinion manipulation attacks benefit from targeting people more susceptible

to the influence of others, and from targeting networks where the "importance" assigned to a given person and his/her opinion depends on how popular he/she is. The simulations carried out in sections 5.4 and 5.5 showed that the effectiveness of an attack improves in networks where the trustiness between agents is proportional to their degree, and in networks where the uncontrolled agents have an higher influenceability. However, a greater influenceability also increases the variance of the vote share.

7.1 Possible future works

Some directions for future developments on the topics covered by this thesis could be:

- The validation of the presented results on real social network topologies;
- Further investigations on the dependence of the relative effectiveness of centrality indices with both the number of controlled agents and the network topology;
- The investigation of different criteria for selecting the agents to be controlled, or the employment of different centrality measures, such as the weighted versions of betweenness and eigenvector centrality;
- The development of strategies for restoring the steady-state average opinion of a network after an opinion manipulation attack which are capable of restoring a neutral opinion also within the communities;
- The development of theoretical analysis for the multi-agent Markovian model in the case of biased interaction intensity parameters;
- The extension of both theoretical and experimental results to the case of more than two possible opinions in the network;
- The development of methods for estimating in real social networks the model parameters, such as the individual prejudice matrix, the trustiness model, and the influenceability of the agents.

Bibliography

[1] Update on Twitter's review of the 2016 US election. *Twitter Blog*, https://blog.twitter.com/en_us/topics/company/2018/2016-election-update.html, 2018.

[2] https://russiatweets.com/author.

[3] J. Albright. *Tableau Public*, https://public.tableau.com/profile/d1gi#!/vizhome/FB4/TotalReachbyPage, 2017.

[4] D. Ingram. Facebook says 126 million Americans may have seen Russia-linked political posts. *Reuters*, https://reut.rs/2yZUXa0, 2017.

[5] K. Jamieson. Interview by Judy Woodruff. *PBS NewsHour*, https://www.pbs.org/newshour/show/why-this-author-says-its-highly-probable-russian-interference-swung-the-2016-election, 2018.

[6] P. Bolzern, P. Colaneri, G. De Nicolao. Opinion influence and evolution in social networks: a Markovian agents model. *Automatica, Vol. 100, 219:230*, 2019.

[7] P. Bolzern, P. Colaneri, G. De Nicolao. Opinion dynamics in social networks: The effect of centralized interaction tuning on emerging behaviours. *IEEE Transactions on Computational Social Systems, Vol. 7, No. 2, 362:372*, 2020.

[8] P. Bolzern, P. Colaneri, G. De Nicolao. Effect of social influence on a two party election: A Markovian multi-agent model. *Submitted to IEEE Transactions on Control of Network Systems*, 2021.

[9] A. Proskurnikov, R. Tempo. A tutorial on modeling and analysis of Dynamic Social Networks, part I. *Annual Reviews in Control, Vol. 43, 65:79*, 2017.

[10] J. French. A formal theory of social power. *Psychological Review, Vol. 63, No. 3, 181:194*, 1956.

[11] M. DeGroot. Reaching a consensus. *Journal of the American Statistical Association, Vol. 69, No. 345, 118:121*, 1974.

[12] R. Abelson. Mathematical models of the distribution of attitudes under controversy. *N. Frederiksen, H. Gulliksen (Eds.), Contributions to Mathematical Psychology. Holt, Rinehart & Winston, Inc. 142:160*, 1964.

[13] M. Taylor. Towards a mathematical theory of influence and attitude change. *Human Relations, Vol. 21, No. 2, 121:139*, 1968.

[14] N. Friedkin, E. Johnsen. Social influence and opinions. *Journal of Mathematical Sociology, Vol. 15, No. 3-4, 193:206*, 1990.

[15] A. Proskurnikov, R. Tempo. A tutorial on modeling and analysis of Dynamic Social Networks, part II. *Annual Reviews in Control, Vol. 45, 166:190*, 2018.

[16] S. Banisch, R. Lima, T. Araújo. Agent based models and opinion dynamics as Markov chains. *Social Networks, Vol. 34, No. 4, 549:561*, 2012.

[17] C. Asavathiratham, S. Roy, B. Lesieutre, G. Verghese. The influence model. *IEEE Control Systems, Vol. 21, No. 6, 52:64*, 2001.

[18] D. Gillespie. A general method for numerically simulating the stochastic time evolution of coupled chemical reactions. *Journal of Computational Physics, Vol. 22, No. 4, 403:434*, 1976.

[19] D. Gillespie. Exact Stochastic Simulation of Coupled Chemical Reactions. *The Journal of Physical Chemistry, Vol. 81, No. 25, 2340:2361*, 1977.

[20] R. Toivonen, J. Onnela, J. Saramäki, J. Hyvönen, K. Kaski. A model for social networks. *Physica A: Statistical Mechanics and its Applications, Vol. 371, No. 2, 851:860*, 2006.

[21] A. Lancichinetti, S. Fortunato, F. Radicchi. Benchmark graphs for testing community detection algorithms. *Physical Review E, Vol. 78, No. 4, 046110*, 2008.

[22] A. Barabási. Network Science. Sec. 4.8. *Cambridge University Press*, 2016.

[23] G. Palla, I. Derényi, I. Farkas, T. Vicsek. Uncovering the overlapping community structure of complex networks in nature and society. *Nature, Vol 435, 814:818*, 2005.

[24] Y. Ahn, J. Bagrow, S. Lehmann. Link communities reveal multiscale complexity in networks. *Nature, Vol. 466, 761:764*, 2010.

[25] E. Ravasz, A. Somera, D. Mongru, Z. Oltvai, A. Barabási. Hierarchical organization of modularity in metabolic networks. *Science, Vol. 297, 1551:1555*, 2002.

[26] M. Girvan, M. Newman. Community structure in social and biological networks. *PNAS, Vol. 99, 7821:7826*, 2002.

[27] M. Newman. Fast algorithm for detecting community structure in networks. *Physical Review E, Vol. 69, No. 6, 066133*, 2003.

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

[29] L. Freeman. A set of measures of centrality based on betweenness. *Sociometry, Vol. 40, No. 1, 35:41*, 1977.

[30] S. Seidman. Network structure and minimum degree. *Social Networks, Vol. 5, No. 3, 269:287*, 1983.

[31] M. Kitsak, L. Gallos, S. Havlin, F. Liljeros, L. Muchnik, H. Stanley, H. Makse. Identification of influential spreaders in complex networks. *Nature Physics, Vol. 6, 888:893*, 2010.

[32] B. Viswanath, A. Mislove, M. Cha, K. Gummadi. On the evolution of user interaction in Facebook. *Proceedings of the 2nd ACM Workshop on Online Social Networks*, 2009.