Harmonic Influence Centrality on networks: a Message Passing optimization approach

Relatori:
Fabio Fagnani
Paolo Frasca
Asuman Ozdaglar

Candidato:
Luca Vassio

Novembre 2012
Abstract

A key issue in the study of networks is the identification of their most important nodes: the definition of prominence is based on evaluating a suitable function of the nodes, called centrality measure. In this thesis we study a specific example of centrality measure, which is motivated by social networks where nodes are agents who possess evolving opinions.

We consider a gossip model of opinion evolution over networks comprising stubborn agents, who influence their neighbors but do not change their own opinions. Although the presence of such agents precludes convergence of opinions to a consensus, opinions converge in distribution: we thus focus on the expectation of this limit distribution of opinions. For simplicity, we consider opinions between 0 and 1, and assume that the network already comprises a certain number of stubborn nodes with opinion 0. Given a network and a set of stubborn nodes with opinion 0, we define as Harmonic Influence Centrality of a certain node the sum of all the expected limit opinions, which results from setting that node as a stubborn node with opinion 1. Consequently, we define the Optimal Stubborn Agent Placement problem as the problem of finding the node with highest HIC, that is, determining the location where a single stubborn agent has the maximum impact on the network.

This thesis contains three main contributions about computing HIC and solving OSAP problems. First, by an analogy with electrical networks, we characterize the expected opinions as electric potentials and, assuming the network to be a tree, we prove several properties of the solution of the OSAP. These properties are key to our second contribution, which is a message passing algorithm, designed to compute the harmonic influence centrality over tree networks. This algorithm has two advantages, compared to previous approaches in the literature: it runs in a distributed and parallel way among the nodes, which do not need to know the topology of the whole network, and has a lower cost in terms of number of operations. Third, we present a suitable modification of this algorithm, which can be applied to general networks. We prove convergence of this algorithm for d-regular networks and we obtain very encouraging simulation results. Indeed, extensive simulations show that the algorithm converges on general networks. Moreover, although the algorithm is not in principle able to compute correctly the HIC, in many significant examples it achieves a good approximation, which is sufficiently accurate to solve the OSAP problem.

While most of the thesis is devoted to harmonic influence centrality, in the last chapter we broaden our investigation to consider other relevant centrality measures. Specifically, we focus on the family of Bonacich centralities, proposing an iterative algorithm to compute such measures in a distributed way among the agents.

Key words: message passing algorithms, multi-agent systems, centrality measures, social networks, opinion dynamics, distributed computation.
Abstract - Italian version

Un problema chiave nello studio delle reti è l’identificazione dei loro nodi più importanti: la definizione di importanza è data attraverso un’appropriata funzione sui nodi, chiamata *misura di centralità*. In questa tesi studiamo uno specifico esempio di misura di centralità, motivata da applicazioni nel caso di *social network* dove i nodi sono agenti che posseggono delle opinioni che cambiano nel tempo.

Consideriamo un modello di dinamica di opinioni di tipo *gossip* su reti con la presenza di agenti *stubborn*, che influenzano gli altri agenti ma non cambiano mai la loro opinione. La presenza di tali agenti preclude la convergenza al *consensus* e genera disaccordo nella rete; tuttavia le opinioni convergono in distribuzione: noi ci concentriamo sul loro valore atteso asintotico. Per semplicità consideriamo opinioni comprese tra 0 e 1, e assumiamo che la rete comprenda già un certo numero di nodi stubborn con opinione 0. Data una rete e un insieme di agenti stubborn con opinione 0, definiamo la *Harmonic Influence Centrality* (centralità armonica) di un certo nodo come la somma delle opinioni medie asintotiche, scelto tale nodo come *stubborn* con opinione 1. Definiamo quindi il problema di *Optimal Stubborn Agent Placement*, che consiste nel determinare la posizione di un singolo agente *stubborn* in modo tale da avere il massimo impatto sulla rete.

La presente tesi contiene tre principali contributi riguardo il calcolo della centralità armonica e sulla risoluzione del problema OSAP. Per prima cosa, attraverso un’analogia con le reti elettriche, caratterizziamo le opinioni attese come potenziali elettrici e, nel caso di reti ad albero, presentiamo alcune proprietà della soluzione del problema OSAP. In secondo luogo queste proprietà vengono utilizzate per presentare un algoritmo di *message passing*, che calcola la centralità armonica sugli alberi. Questo algoritmo presenta dei vantaggi, paragonato a un precedente approccio in letteratura: agisce in maniera distribuita e parallela e migliora l’efficienza in termini di complessità computazionale. Infine modifichiamo tale algoritmo per poterlo applicare a reti generali. Dimostriamo la convergenza dell’algoritmo per grafi regolari; inoltre, constatiamo la convergenza in tutte le simulazioni effettuate anche per altri tipi di grafi. Anche se l’algoritmo modificato per i grafi non calcola correttamente la centralità armonica, in molte simulazioni su grafi significativi abbiamo ottenuto delle buone approssimazioni, sufficienti a una corretta risoluzione del problema di OSAP.

Mentre la maggior parte della tesi è concentrata sulla centralità armonica, nell’ultimo capitolo consideriamo altre rilevanti misure di centralità. Ci concentriamo in particolare sulla famiglia di misure di centralità di Bonacich, proponendo un algoritmo iterativo per calcolare tali misure in maniera distribuita tra gli agenti.

**Parole chiave**: algoritmi di *message passing*, sistemi multi-agente, misure di centralità, *social network*, dinamica di opinioni, calcolo distribuito.
Contents

1 Introduction and motivation ................................. 1
  1.1 Preliminaries .............................................. 4

2 Gossip models with stubborn agents ....................... 7
  2.1 The gossip model with stubborn agents ................. 7
    2.1.1 Binary homogeneous gossip model with stubborn agents .... 9
  2.2 Expected values of the opinions in the society ....... 10
    2.2.1 Harmonic functions .................................. 12
  2.3 The Optimal Stubborn Agent Placement problem ...... 13
    2.3.1 The Harmonic Influence Centrality .................. 14
  2.4 The electrical analogy for undirected graphs .......... 15

3 Optimal Stubborn Agent Placement for trees .............. 19
  3.1 Some key examples ....................................... 19
    3.1.1 Lines with stubborn agents in the leaves .......... 20
    3.1.2 Trees with two stubborn agents ..................... 22
    3.1.3 HIC on trees with one stubborn agent ............. 24
    3.1.4 HIC on lines with stubborn agents in the leaves . 25
  3.2 A key property ........................................... 27
  3.3 Properties of potentials ................................ 29
7 Other centrality measures for graph models

7.1 Some relevant graph centralities

7.1.1 Degree centrality

7.1.2 Closeness centrality

7.1.3 Betweenness centrality

7.1.4 Eigenvector centrality

7.2 The Bonacich network centrality measures

7.2.1 Example

7.3 Distributed computation of the Bonacich centrality

7.4 An applicative example: Optimal Pricing in networks

7.4.1 Definition and solution of the problem

7.4.2 Distributed computation for the optimal pricing

7.4.3 Some numerical examples

A Graph Theory basics

A.1 Opinion dynamics over graphs

B Markov Chains

Bibliography
Chapter 1

Introduction and motivation

A graph or a network is an abstraction for modeling relationships between objects. Graphs are very powerful modeling tools for modeling natural and man-made systems. For example social networks and the internet can be represented and analyzed as networks. In figure 1.1 on the following page we can see a partial visualization of the graph of the internet: each line between two nodes represents an interconnection between two routers (IP addresses).

A key issue in the study of networks is the identification of their most important nodes: the definition of prominence is based on evaluating a suitable function of the nodes, called centrality measure. To provide some applicative examples, these kind of measures can represent how influential a person is within a social network, or how efficient is a router in a communication network ([12]). In this thesis we study a specific example of centrality measure, which is motivated by social networks where nodes are agents who possess evolving opinions.

Many models for opinion formation within a network predicts that the opinion dynamics should converge to agreement, usually called consensus. In many real cases indeed societies appear to exhibit permanent disagreement, even on topics on which there is frequent communication (see [3]). We present in chapter 2 on page 7 a gossip model of opinion evolution over networks comprising stubborn agents, who influence their neighbors but do not change their own opinions (see [33] and [2]). Although the presence of such agents precludes convergence of opinions to a consensus, opinions converge in distribution: we thus focus on the expectation of this limit distribution of opinions.

For simplicity, we consider a model with opinions between 0 and 1, and assume that the network already comprises a certain number of stubborn nodes with opinion 0. Given a network and a set of stubborn nodes with opinion 0, we define as Harmonic Influence Centrality (HIC) of a certain node the sum of the expected limit opinions, which results from setting that node as a stubborn node with opinion 1. Consequently, we define the Optimal Stubborn Agent Placement problem (OSAP)
as the problem of finding the node with highest HIC, that is, determining the location where a single stubborn agent has the maximum impact on the network. The OSAP problem is therefore an optimization problem on a graph.

An applicative example could be the case of a company that has to sell a new product. The agents are the potential customers and the opinion of an agent is his willingness to buy the new product.
The agents are able to influence each other through the links, but there are some of them, the stubborn ones, that they will never buy the product. The company could therefore be interested in finding who is the most influential person in the social network, and then try to convince him in some way to advertise its product.

This thesis contains three main contributions about computing HIC and solving OSAP problems. First, by an analogy with electrical networks (\[20\]), we characterize the expected opinions as electric potentials and, assuming the network to be a tree, we prove in chapter \[3 on page 19\] several properties of the solution of the optimal stubborn agent placement. In particular we simplify the solution allowing to analyze the harmonic influence centrality on a subset of the whole set of agents.

These properties are key to our second contribution, which is a message passing algorithm, designed to compute the harmonic influence centrality over tree networks. Message passing algorithms (see [19], [24], [25] and [29]) are distributed algorithms related to a tree, in which nodes are objects with computational ability and can send and receive information to their neighbors. The links represent therefore communication channels and the packets of information that flow through the links are called messages. The message passing algorithm proposed in chapter \[4 on page 41\] has two big advantages, compared to previous approaches in the literature (see [33]):

1. it runs in a distributed and parallel way among the nodes, which do not need to know the topology of the whole network but only the set of their neighbors.

2. has a lower cost in terms of number of operations (computational complexity).

We present in chapter \[5 on page 53\] our third contribution, a suitable modification of this algorithm, which can be applied not only to trees but to general networks. We prove convergence of this algorithm for $d$-regular networks in chapter \[6 on page 73\] and we obtain very encouraging simulation results. Indeed, extensive simulations show that the algorithm converges on general networks. Moreover, although the algorithm is not in principle able to compute correctly the HIC, in many significant examples, such as Erdos-Renyi random graphs, it achieves a good approximation, which is sufficiently accurate to solve the OSAP problem.

While most of the thesis is devoted to harmonic influence centrality, in the last chapter we broaden our investigation to consider other relevant centrality measures. Which definition of centrality is the most suitable depends on the context (see [4]). Specifically, we focus on the family of Bonacich centralities ([5]), proposing an iterative algorithm to compute such measures in a distributed way among the agents. We also present an economical application ([8]), where a monopolist wants to make the highest profit selling a divisible good in the market choosing prices for the customers.

In the next section we present the basic notations of graphs, while in the appendix \[A on page 99\]...
and B on page 103 we recall some further notions of graph theory and Markov chains.

1.1 Preliminaries

First are presented some definitions and notations used throughout the text. For more details we refer to the appendix.

A graph $G$ is a pair $G := (I, E)$ where $I$ is a finite set, whose elements are said to be the nodes of $G$ and $E \subset I \times I$ is the set of edges. The nodes in the field of opinion dynamic are also called agents or individuals. $E$ is the set of links representing the connectivity among the agents. The edge $(i, j) \in E$ has the meaning that $i$ can receive information from $j$, while the reverse is not true if $(j, i) \notin E$.

The out-neighbor set of a node $i \in I$ is defined as:

$$N_i^{(out)} = \{j \in I | (i, j) \in E\}$$

while the in-neighbor set of a node $i \in I$ is defined as:

$$N_i^{(in)} = \{j \in I | (j, i) \in E\}$$

A graph such that for every $(i, j) \in E$ exists $(j, i) \in E$ is called symmetric. If the graph is symmetric clearly $N_i^{(out)} = N_i^{(in)}$ and this set is simply called neighbor set of node $i \in I$ and is denoted by $N_i$.

A path of length $l$ from node $i$ to $j$ in $G$ is an ordered list of nodes $(w_0, w_1, \ldots, w_l)$ such that $w_0 = i$, $w_l = j$ and $(w_i, w_{i+1}) \in E$ for every $i \in \{0, \ldots, l - 1\}$.

If either a path from $i$ to $j$ exists, or $j = i$, then $j$ is said to be reachable from $i$.

A path $(w_0, w_1, \ldots, w_l)$ is said to be simple if $\forall i, j \in \{0, \ldots, l - 1\}, j \neq i$ then $w_i \neq w_j$. From now on, paths will be assumed to be simple, unless otherwise stated.

From the notion of path we can derive a notion of distance between two nodes:

$$\text{dst}(i, j) := \min\{l : \text{exists a path in } G \text{ of length } l \text{ from } i \text{ to } j\},$$

provided that $i \neq j$ and $\text{dst}(i, i) := 0 \ \forall i$. If $j$ is not reachable from $i$, then $\text{dst}(i, j) := \infty$.

Then the diameter of a graph can be defined as:

$$\text{diam}(G) := \max\{\text{dst}(i, j) : i, j \in I\}$$
Graphs are strictly related to matrices. In particular a graph can be represented by its *adjacency matrix*. Given a graph $G = (I, E)$ the adjacency matrix $A$ is a matrix in $\{0,1\}^{I \times I}$ such that:

$$A_{ij} := \begin{cases} 
1 & \text{if } (i, j) \in E \\
0 & \text{otherwise.}
\end{cases} \quad (1.2)$$

Given instead a matrix $A \in \mathbb{R}^{I \times I}$ whose entries are nonnegative, we can build a graph $G_A = \{I, E\}$ such that $(i, j) \in E$ if and only if $A_{ij} > 0$. 
Chapter 2

Gossip models with stubborn agents

In many models for opinion formation within a network it is predicted that the opinion dynamics should converge to agreement, usually called consensus. See the appendix for a quick recall. In many real cases instead societies appear to exhibit permanent disagreement, even on topics on which there is frequent communication. We refer to [3] for a recent survey.

Starting from well-known gossip models for opinion dynamics over networks (see [18] and [10]), we add to the model nodes that never change their opinions in order to prevent consensus in the society. We first present a general model, called the gossip model (or belief evolution model) with stubborn agents (see [2]), and then a particular case of it called the binary homogeneous gossip model with stubborn agents (see [33]).

2.1 The gossip model with stubborn agents

The model and the results presented are outlined from [2]. This model is also referred in literature as the belief evolution model with stubborn agents.

Consider a graph $G = (I, E)$ called in this context the social network. $I$ is the population represented by the set of agents and $E \subset I \times I$ is the set of directed links between the agents. A link $(i, j) \in E$ has the meaning that $i$ can access to the information held by $j$ and therefore can be influenced by it. Each node $i \in I$ has an internal time-dependent state $x_i(t) \in \mathbb{R}$. In the field of opinion dynamic over networks this state is called also opinion or belief. We use $x(t) = (x_i(t))_{i \in I}$ to denote the full agent opinion vector at time $t$.

Every edge $(i, j) \in E$ has a clock associated with it that follows an independent homogeneous Poisson process with rate $\gamma_{ij} > 0$. 
We recall that a homogeneous Poisson process is characterized by its rate parameter $\gamma$, which is the expected number of events (or arrivals) that occur per unit time. The time between each pair of consecutive events has an exponential distribution with parameter $\gamma$ and each of these inter-arrival times is assumed to be independent of other inter-arrival times. We remember that the homogeneity is given by the fact that the rate $\gamma$ does not change in time. For further details and for the definition of a Poisson process we refer to the appendix.

For an edge $(i,j) \in E$, when the corresponding clock ticks at time $t$, agent $i$ reads the opinion of $j$ and updates his own opinion according to the following rule:

$$x_i(t) = (1 - \theta_{ij})x_i(t^-) + \theta_{ij}x_j(t^-)$$

(2.1)

where $x_i(t^-)$ stands for the left limit $\lim_{u \uparrow t} x_i(u)$. The scalar $\theta_{ij} \in [0,1]$ is a trust parameter that represents the confidence that agent $i$ has in agent $j$’s opinion. If $\theta_{ij} = 0$ the edge $(i,j) \in E$ is not actually used and therefore is the same of do not considering it.

The dynamic of the opinions over the graph $G$ is fully characterized by equations (2.1) and by the initial conditions $x_i(0) \ \forall i \in I$.

Suppose now that there are some special agents in the society, the stubborn agents. Stubborn agents are so called because they never change their opinion. The set of stubborn agents is denoted as $S \subset I$.

This means that $\theta_{ij} = 0 \ \forall j \in N_i$. These agents never update their opinions, that is equivalent to say that they do not have any out-neighbor:

$$x_s(t) = x_s(0) =: x_s, \ s \in S, t \geq 0$$

(2.2)

Non-stubborn nodes $i \in I \setminus S$ are also called regular agents.

We are particularly interested in the asymptotic behavior of the states of the nodes of the graph.

Assume from now on the following assumption on the graph structure:

**Assumption 2.1.** For each non-stubborn node $i \in I \setminus S$ there exists a direct path from node $i$ to at least one stubborn node $s \in S$.

This assumption imposes the mild restriction that every non-stubborn agent in the society can reach, and therefore be influenced by, at least one stubborn agent (independently from its value) through a multi-hop path.

Then in this model agent states $x_i(t)$ do not necessarily reach a consensus since two agents may be connected to distinct stubborn agents with distinct fixed opinions.

Despite this fact it is possible to prove (see [2]) that the agent state vector $x(t)$ converges in distribution:
Lemma 2.1. Let assumption 2.1 hold. For every value of the stubborn agents beliefs $x_s \in \mathbb{R}, s \in S$ the agent state vector $x(t)$ converges in distribution for $t \to +\infty$.

Hence we have found that in the gossip model with stubborn agents the states do not necessarily reach a consensus, but the asymptotic states assume values according to the unique stationary distribution $\mu$. We will call the asymptotic opinion of the society the random vector $x := \lim_{t \to +\infty} x(t)$. We underline the important fact that the stationary distribution of $x$ does not depend on the initial state conditions $x_i(0)$ of the non-stubborn nodes $i \in I \setminus S$. We refer to [2] and [33] for the proof and further details.

The stubborn agents, who never update their opinions, might represent leaders, political parties or media sources attempting to influence the beliefs in the rest of the society. When the society contains stubborn agents with different opinions, opinion dynamic does not lead to a consensus (among the regular agents). Instead, beliefs in the society almost surely fail to converge, and the belief of each regular agent converges in law to a non-degenerate random variable. The model thus generates long-run disagreement and continuous opinion fluctuations. The structure of the social network, and the location of stubborn agents within it, shapes the opinion dynamics.

2.1.1 Binary homogeneous gossip model with stubborn agents

This model is a particular case of the gossip model seen before, where agents can only assume values between 0 and 1. This particular case is useful to model a simple yes/no decision as a choice between two political parties.

We add the following two assumption of the gossip model with stubborn agents:

Assumption 2.2. Let $S = S^0 \cup S^1$ where $S^0 := \{s \in S | x_s = 0\}$ and $S^1 := \{s \in S | x_s = 1\}$.

Nodes $i \in S^0$ are called type 0 stubborn nodes, while nodes $i \in S^1$ are called type 1 stubborn nodes.

We choose the opinions 0 and 1, but the results are extendable to a general case.

Assumption 2.3. Let $\gamma_{ij} = \frac{1}{|N_i|}$ for all $i, j \in I$ and let $\theta_{ij} = \theta$ for all $i \in I \setminus S, j \in I$.

The assumption 2.3 is an homogeneity assumption. $\theta_{ij} = \theta$ for all $i \in I \setminus S, j \in I$ means that each node in the graph trust every other node in the same way. Moreover every node contact one of its neighbors in average every one unit of time. In fact:

$$\sum_{j \in N_i} \gamma_{ij} = \sum_{j \in N_i} \frac{1}{|N_i|} = 1$$
This new model, that we are going to study, is called binary homogeneous gossip model with stubborn agents.

Notice that this simplification of the gossip model can be seen also as a generalization of another well known model, the binary voter model (see [33]).

2.2 Expected values of the opinions in the society

Consider the gossip model introduced in section 2.1 on page 7. Thanks to theorem 2.1 on the preceding page it is now known that the random vector of asymptotic opinions $x$ in the gossip model has a stationary distribution. We focus on the first moment $E[x]$ of this stationary distribution of the asymptotic opinions.

We provide explicit characterization of the stationary expected beliefs in term of hitting probabilities of a continuous time Markov chain on $G$. For a recall on continuous time Markov chains, hitting times and probabilities see the appendix.

Consider the continuous time Markov chain $Z(t)$ on $I$ with transition rate matrix $Q$ defined as follows:

$$Q_{ij} := \begin{cases} 
\theta_{ij} \gamma_{ij} & \text{if } i \neq j \\
-\sum_{k \neq i} Q_{ik} & \text{if } i = j 
\end{cases} \quad (2.3)$$

Let us denote by $\tau_S$ the hitting time of the Markov chain $Z(t)$ on the set of stubborn agents $S$:

$$\tau_S := \inf \{ t \geq 0 : Z(t) \in S \}$$

Observe that assumption 2.1 on page 8 implies that $\tau_S$ are finite with probability one for every initial distribution of $Z(0)$. Hence, for all $i \in I$ we can define the hitting probability distribution $\phi_i$ over $S$, whose entries are given by:

$$\phi_i^s := P_i(Z(\tau_S) = s), \ s \in S$$

Then we have the following result proved in [2].

**Theorem 2.2.** Let assumption 2.1 hold. For every value of the stubborn agents beliefs $x_s \in \mathbb{R}, s \in S$
\[
E[x_i] = \sum_{s \in S} \phi_s^i x_s \forall i \in I
\]  
\tag{2.4}

Moreover \( \{E[x_i] : i \in I\} \) are the unique vectors in \( \mathbb{R}^I \) satisfying:

\[
\sum_{j \in I} Q_{ij} E[x_j] = 0 \forall i \in I \setminus S, \ E[x_s] = x_s \forall s \in S
\]

We can see in equation (2.4) that the asymptotic expected opinion of all the agents in the network are a linear combination of the opinions of the stubborn nodes.

Assume that the number of nodes in the society are \( N \), i.e. \( |I| := N \). Consider the arithmetic mean \( \overline{x}(t) \) between the states of all the nodes in the graph:

\[
\overline{x}(t) := \frac{1}{N} \sum_{i \in I} x_i(t).
\]  
\tag{2.5}

We have that \( \overline{x}(t) \) is therefore the average opinion between all the nodes \( i \in I \) in the graph \( G \) at time \( t \). What we want to study is the asymptotic behavior of \( \overline{x}(t) \).

From theorem 2.1 on page 9 we derive that the average opinion of the society also converges in distribution. We call this asymptotic average opinion as \( \overline{x} \)

\[
\overline{x} := \frac{1}{N} \sum_{i \in I} x_i.
\]  
\tag{2.6}

Since \( \overline{x} \) is also a random variable we will study its expected value:

\[
E[\overline{x}] = \frac{1}{N} \sum_{i \in I} E[x_i].
\]  
\tag{2.7}

This quantity represent therefore the expected asymptotic bias of the society toward the yes decision.

We can therefore outline the bias of the whole society with just one number.

Consider now the binary homogeneous gossip model introduced in section 2.1.1 on page 9 through assumption 2.2 on page 9 and 2.3 on page 9.

Using theorem 2.2 on the facing page we can simplify the expression of the expected stationary beliefs.
Denote by $\phi_i^{S^1}$ the hitting probability in the type 1 stubborn set $S^1$ starting from $i$:

$$\phi_i^{S^1} = \mathbb{P}_i(Z(\tau_S) = s, s \in S^1)$$

We can rewrite theorem 2.2 on page 10:

**Corollary 2.3.** Consider a binary homogeneous gossip model on a graph with assumptions 2.1 on page 8. Then the expected value of the asymptotic opinion of agent $i$ is equal to the probability that the random walk described in (2.3) initiated at node $i$ is absorbed by the set $S^1$, i.e.:

$$\mathbb{E}[x_i] = \phi_i^{S^1} \quad \forall i \in I$$

(2.8)

Moreover $\mathbb{E}[x_i]$ can be characterized as the unique vector in $\mathbb{R}^I$ satisfying:

$$\mathbb{E}[x_i] = \frac{1}{|N_i|} \sum_{j \in N_i} \mathbb{E}[x_j] \quad \forall i \in I \setminus S, \quad \mathbb{E}[x_s] = x_s \quad \forall s \in S$$

(2.9)

So $\mathbb{E}[x_i]$ is the probability that the opinion asked by $i$ hits one of the node in $S^1$ before hitting a node in $S^0$. This probability can be interpreted as the bias of that node.

### 2.2.1 Harmonic functions

Given the continuous-time Markov process $Q$, its embedded Markov chain (EMC) is a regular discrete-time Markov chain in which each element of the one-step transition probability matrix, $P$, is denoted by $P_{ij}$, and represents the conditional probability of transitioning from state $i$ into state $j$. These conditional probabilities may be found by:

$$P_{ij} = \begin{cases} 
\frac{Q_{ij}}{\sum_{k \neq i} Q_{ik}} & \text{if } i \neq j \\
0 & \text{otherwise.}
\end{cases}$$

(2.10)

Then we can define an harmonic function related to the Markov process with state space $I$.

**Definition 2.1.** A function $W : I \rightarrow \mathbb{R}$ is called harmonic in $i \in I$ if:

$$W(i) = \sum_{j \in I} P_{ij} W(j)$$

(2.11)

A function is harmonic on $J \subset I$ if it is harmonic at every state $i \in J$.
2.3 The Optimal Stubborn Agent Placement problem

Consider the binary homogeneous Gossip model described in section 2.1.1 on page 9. In the following part we define the problem of the optimal stubborn agent placement for such model. Let the stubborn agents set $S = S^1 \cup S^0$ in which $S^1 \neq \emptyset$ and $S^0 \neq \emptyset$.

The problem consists in choosing the set $S^1$ of size fixed in order to maximize the expected average opinion of the society in (2.7), given the graph $G$ and $S^0$.

In the maximization of the quantity (2.7) $N$ does not actually play a role and therefore we could simplify (2.7) denoting with $H$ the new quantity:

$$H(S^1) := \sum_{i \in I} \mathbb{E}[x_i]$$  \hspace{1cm} (2.12)

Notice that $H(S^1)$ is $N \cdot \mathbb{E}[\overline{x}]$.

$H$ represents the sum of the asymptotic expected values of the opinions in the society. We have that $H$ depends both on the topology of the graph $G = (I, E)$ and on the choice of $S^0$ and $S^1$. We use the notation $H(S^1)$ to remark the dependence of $H$ from the choice of $S^1$. We remark the fact that $|S^1|$ should be less than or equal to $|I| - |S^0|$. We have therefore to solve the following optimization problem with objective function $H(S^1)$:

$$\arg\max_{S^1 \subseteq I \setminus S^0} H(S^1)$$  \hspace{1cm} (2.13)

(2.13) is called the Optimal Stubborn Agent Placement (OSAP) problem. We want therefore to maximize the sum of the expected values of the opinions in the society choosing the type 1 stubborn nodes set.
2.3.1 The Harmonic Influence Centrality

We focus for simplicity from this point on the problem in which $|S^1| = 1$ and therefore $S^1 = \{l\}$. With a slight abuse of notation we write $H(l)$ to denote the dependance of $H$ on the choice of the location of the stubborn agent. The *Optimal Stubborn Agent Placement* (OSAP) problem consists in finding:

$$\arg\max_{l \in I \setminus S^0} H(l).$$  \hspace{1cm} (2.14)

We can see that $H(l)$ is a measure of how much node $l$ is good to influence all the other nodes and how much is able to change the average opinion over all the network. Notice that since $S^1 = \{l\}$ the node $l$ is the only responsible for moving the asymptotic opinion of the nodes in the graph. We can then think $H(l)$ as a measure of centrality of the node $l$.

**Definition 2.2.** [Harmonic Influence Centrality - HIC] Consider the binary homogeneous gossip model. For all $l \in I \setminus S^0$, let $S^1 = \{l\}$. Then

$$H(l) := \sum_{i \in I} \mathbb{E}[x_i], \hspace{1cm} \forall l \in I \setminus S^0$$  \hspace{1cm} (2.15)

is called harmonic influence centrality of a node $l \in I \setminus S^0$. Moreover for $l \in S^0$ we define:

$$H(l) := 0 \hspace{1cm} \forall l \in S^0$$

Follows from the definition the fact that for every graph:

$$1 \leq H(l) \leq N - |S^0|, \hspace{1cm} \forall l \in I \setminus S^0$$

The name harmonic (and the capital letter $H$) comes from the fact that according to proposition 2.4 on the preceding page $\mathbb{E}[x_i]$ is an harmonic function $\forall i \notin S$ (see definition 2.1 on page 12).

We will see in the next chapters how to characterize the harmonic influence centrality and how to compute it in a distributed way.

As written in literature (see [33]) there are already analytical methods to solve exactly problem (2.14), but we want to find different methods to exploit in some particular cases, using what we present in paragraph 2.4 on the facing page. The electrical interpretation has not been taken in
account in [33] for solving the optimal stubborn agent placement problem.

2.4 The electrical analogy for undirected graphs

We outline this part from [20], presenting only the fundamental results to understand the rest of the thesis. From now on we consider symmetric graphs: if \((i,j) \in E\) then \((j,i) \in E\).

We have seen in section 2.2 on page 10 through theorem 2.2 and corollary 2.3 that we are able to characterize the asymptotic opinion of the nodes of society through the study of particular Markov chains over the graph \(G\).

Electrical networks provide a different language for reversible Markov chains. For further details on reversible Markov chains see again the appendix. In particular this point of view is useful because of the insight gained from the familiar physical laws of electrical networks.

An electrical network is a finite undirected connected graph \(G\) with vertex set \(I\) and edge set \(E\), endowed additionally with non-negative numbers \(\{c(e)\}_{e \in E}\), called conductances, that are associated to the edges of \(G\). Clearly \(c(i,j) = c(j,i)\). The reciprocal \(R(e) := \frac{1}{c(e)}\) is called the resistance of the edge \(e\).

If we consider the Markov chain on the nodes of \(G\) with transition matrix:

\[
P_{ij} = \begin{cases} 
c(i,j) \sum_j c(i,j) & \text{if } (i,j) \in E \\ 
0 & \text{otherwise}
\end{cases}
\] (2.16)

this process is called the weighted random walk on \(G\) with edge weights \(c(e)\). This Markov chain is reversible with respect to the probability \(\pi\) defined by:

\[
\pi(i) := \frac{\sum_j c(i,j)}{\sum_{i,j} c(i,j)}
\] (2.17)

Otherwise if \(P\) is a transition matrix on a finite set which is reversible with respect to the probability \(\pi\), then we can define conductances on edges by \(c(i,j) = \pi(i)P_{ij}\).

**Proposition 2.5.** The Markov chain described in (2.3) for the binary homogeneous gossip model on a symmetric graph is reversible for all \(i \in I \setminus S\). Moreover we can consider \(c(i,j) = 1\) for all \((i,j) \in E\).

**Proof.** Consider the embedded Markov chain in (2.10) of (2.3) for the binary homogeneous gossip model:
\[
P_{ij} = \begin{cases} 
\frac{1}{|N_i|} & \text{if } (i, j) \in E \\
0 & \text{otherwise.} 
\end{cases} \tag{2.18}
\]

The probability:

\[
\pi(i) = \frac{|N_i|}{\sum_{j \in I} |N_j|} \tag{2.19}
\]

is the stationary probability of the chain (2.18). In fact satisfies \( \pi^*P = \pi^* \).

The chain is reversible since it holds:

\[
\pi(i)P_{ij} = \pi(j)P_{ji} = \frac{1}{\sum_{k \in I} |N_k|} \quad \forall (i, j) \in E
\]

If we consider \( c(i, j) = 1 \) for all \((i, j) \in E\) we obtain that the transition matrix in (2.16) is the embedded Markov chain (2.18) and the probability (2.17) is the stationary probability (2.19).

We distinguish two nodes in the electrical network, \( a \) and \( z \), which are called the source and the sink of the network. A function \( W \) which is harmonic on \( I \setminus \{a, z\} \) is called a voltage. We know that the voltage is completely determined by its boundary values \( W(a) \) and \( W(z) \).

A current flow \( C \) is a function on oriented edges which is antisymmetric, meaning that \( C((i, j)) = -C((j, i)) \) with \((i, j) \in E\). Observe that flows that are defined on oriented edges, while conductance and resistance are defined for unoriented edges. We may of course define them (for future notational convenience) on oriented edges by \( c((i, j)) = c((j, i)) = c(i, j) \) and \( R((i, j)) = R((j, i)) = R(i, j) \).

Given a voltage function \( W \) on the network, the current flow \( C \) associated with \( W \) is defined on oriented edges by:

\[
C(i, j) = \frac{W(i) - W(j)}{R(i, j)} = c(i, j)[W(i) - W(j)]. \tag{2.20}
\]

Notice that (2.20) is nothing but the Ohm’s law.

It is often possible to replace an electrical network by a simplified one without changing quantities of interest, for example the equivalent resistance between a pair of nodes. We define as \( R(i, j) \) the resistance between node \( i \) and node \( j \). The following laws are very useful:

**Series Law.** Resistances in series add: if \( i \in I \setminus \{a, z\} \) is a node of degree 2 with neighbors \( j \) and \( k \), the edges \((j, i)\) and \((i, k)\) can be replaced by a single edge \((j, k)\) of resistance \( R(j, k) := R(j, i) + R(i, k) \).
All potentials and currents in $G^* = (I \setminus \{i\}, E \setminus \{(j, i), (i, k)\})$ remain the same and the current that flows from $j$ to $k$ equals $C(j, k) = C(j, i) = C(i, k)$.

**Parallel Law.** Conductances in parallel add: suppose edges $e_1$ and $e_2$, with conductances $c_1$ and $c_2$, respectively, share vertices $i$ and $j$ as endpoints. Then both edges can be replaced with a single edge $e$ of conductance $c_1 + c_2$ without affecting the rest of the network. All voltages and currents in $E \setminus \{e_1, e_2\}$ are unchanged and the current $C(e)$ equals $C(e_1) + C(e_2)$. In terms of resistances we have a single edge with equivalent resistance of $(\frac{1}{R_1} + \frac{1}{R_2})^{-1}$

**Gluing.** Another convenient operation is to identify vertices having the same voltage, while keeping all existing edges. Because current never flows between vertices with the same voltage, potentials and currents are unchanged.

In the case of the binary homogeneous gossip model $a$ and $z$ are respectively the stubborn node of type 1 and 0 (glued if needed) and the potential imposed is therefore $W(i) = 0$ if $i \in S^0$ and $W(i) = 1$ if $i \in S^1$. Since $E[x_i]$ is an harmonic function (Proposition 2.4 on page 13) with $W(i) = E[x_i] \forall i \in S$, then it is unique and therefore $W(i) = E[x_i] \forall i \in I$. Thanks to Proposition 2.5 on page 15 it follows:

**Proposition 2.6.** Consider the binary homogeneous gossip voter model on a graph such that $G^* = (I \setminus S, E \setminus \{(i, s), s \in S\})$ is a symmetric graph. Then $W(i) = E[x_i] \forall i \in I$, where $W(i)$ is the voltage of the electrical network where each $(i, j) \in E$ are links with unitary resistances, nodes in $S^0$ have potential constant equal to 0 and nodes in $S^1$ have potential constant equal to 1.

Hence with the electrical analogy we have found how to compute the asymptotic opinion of the agents studying the potentials or voltages $W(i)$ in the graph seen as an electrical network. Therefore equation (2.7) reduces to a sum of voltages over the network.
Chapter 3

Optimal Stubborn Agent Placement for trees

Consider the binary homogeneous gossip model described in section 2.1.1. Our goal is to characterize the solution of the optimal stubborn agent placement problem defined in (2.14) and therefore compute the Harmonic Influence Centrality defined in definition 2.2 on page 14.

In this chapter we restrict to a particular kind of graphs: trees. A graph $G = (I, E)$ that is symmetric, acyclic, and connected is called a tree. For details about these properties see the appendix.

A tree with $N$ nodes always has $N - 1$ undirected edges. An important property of trees is that for any pair of nodes $(i, j)$ there is exactly one path from $i$ to $j$ in $G$.

Remark 3.1. Since trees are symmetric, by definition if the directed edge $(i, s) \in E$ exists then also exists the directed edge $(s, i) \in E$, with $s \in S, i \in I$. This edge $(s, i) \in E$, with $s \in S, i \in I$ will not play any role in the dynamic since from equation (2.2) we see that agent $s \in S$ never update its opinion. Hence in the discussion we do not consider the link $(i, s)$, without creating a contradiction with the symmetry property of trees.

3.1 Some key examples

We explicitly compute the stationary expected beliefs for a line with two stubborn agents in the leaves in section 3.1.1 on the next page and for a tree with just two stubborn agents in section 3.1.2 on page 22. Then we will compute the HIC and solve the OSAP problem (2.14) for two other simple examples in section 3.1.4 on page 25 and section 3.1.3 on page 24.

Those examples are presented to the reader to better understand the general results which are
presented later, and because they already provide some useful insights.

### 3.1.1 Lines with stubborn agents in the leaves

Consider first a line $L_N$, a particular case of tree with $N$ undirected edges and therefore $N + 1$ nodes. Label the nodes from 0 to $N$. $L_N = (I, E)$ is called the line graph if $E = \{(i, j) \in I \times I : |i - j| = 1\}$. Suppose that both the two leaves are the only stubborn nodes. Node 0 and node $N$ has constant opinion value, respectively 0 and 1. See figure 3.1.

Consider the binary homogeneous gossip model with stubborn agents. We will compute explicitly the stationary expected beliefs $W(i) := \mathbb{E}[x_i]$, for all $i \in I$. We know $W(0) = 0$ and $W(N) = 1$. We present in the proof two equivalent approaches.

![Figure 3.1. The line graph with two stubborn agents in the leaves](image)

**Proposition 3.1.** The stationary expected beliefs $W(i) := \mathbb{E}[x_i]$ in the line $L_N$ with two stubborn agents in the leaves 0 and $N$ with opinion respectively 0 and 1 are:

$$W(i) = \frac{i}{N} \quad 0 \leq i \leq N \quad (3.1)$$

**Proof.** As we have seen in corollary 2.3 on page 12 we know that the expected stationary beliefs can be characterized as the unique vector in $\mathbb{R}^I$ satisfying:

$$W(i) = \frac{1}{d_i} \sum_{j : (j, i) \in E} W(j) \quad \forall i \in I \setminus \{0, N\}, \quad W(0) = 0, \quad W(N) = 1$$

where $d_i := |N_i|$ is the number of neighbors of node $i$. This is an harmonic condition that in our case, due to the structure of the graph, becomes:
3.1 – Some key examples

\[ W(i) = \frac{W(i + 1) + W(i - 1)}{2} \quad \forall i \in \{1, 2, \ldots, N - 1\} \]

This system of equations has as unique solution:

\[ W(i) = \frac{i}{N} \quad 0 \leq i \leq N \quad (3.2) \]

Then the stationary expected beliefs are the sum of values proportional to the relative distance from the stubborn nodes.

The same result can be obtained using the electrical analogy. As we saw on paragraph 2.4 on page 15 we have to consider the equivalent electrical network having \( N \) wires in series with unitary resistances, i.e. every edge \((i, j) \in E\) has \( R(i, j) = 1 \). It is well known that in a series of wires the voltage is proportional to their resistances. We denote by \( R(h, k) := |k - h|, h, k \in I \) the equivalent resistance from node \( h \) to node \( k \) and by \( W(h, k) := W(k) - W(h) \) with \( h, k \in I \) the difference of potentials from node \( h \) to node \( k \). The current \( C(0, N) \) that flows through the line is given by the Ohm’s law:

\[ C(0, N) = \frac{W(0, N)}{R(0, N)} = \frac{1}{N} \]

Therefore the potential at a node \( i \) is equivalent to:

\[ W(i) = R(0, i)C(0, i) = i \frac{1}{N} \quad 0 \leq i \leq N \]

Which is equivalent to the result stated above in (3.1).

\[ \square \]

Hence the expected asymptotic opinion is proportional to the distance from the node stubborn of type 0. See figure 3.2 on the following page in which the electrical interpretation is also remarked.

If we consider the sum of the expected opinions in the society we get:

\[ \sum_{0 \leq i \leq N} W(i) = \frac{N + 1}{2} \quad (3.3) \]

The average asymptotic opinion \( \bar{x} \) in a line between a type 0 and a type 1 stubborn agents is then equal to:
3.1.2 Trees with two stubborn agents

This example is an extension of the one presented before, and it has already been partially discussed in [2]. We will see here the computation of the stationary expected beliefs, i.e. the potentials $W(i)$ for the binary homogeneous gossip model. $G = (I, E)$ is a tree and the stubborn agent set $S$ consists of only two elements $s_0$ and $s_1$, with beliefs respectively 0 and 1. See figure 3.3 on the next page for an example.

Denote the length of the unique path between two nodes $i, j \in I$ (see equation (A.1)) as the distance $\text{dst}(i, j)$. Let $m := \text{dst}(s_0, s_1)$ and $J = \{j_0 = s_0, j_1, \ldots, j_{m-1}, j_m = s_1\}$ where $\{j_{i-1}, j_i\} \in E$ for all $1 \leq i \leq m$ be the unique path from $s_0$ to $s_1$. Then we can partition the rest of the node set as $I \setminus J = \bigcup_{0 \leq i \leq m} J_i$ where $J_i$ is the set of nodes $k \in I \setminus J$ such that the unique paths from $k$ to $s_0$ and $s_1$ both pass through $j_k$. In figure 3.3 we identify nodes in the same set $J_k$ with the same alphabetic letter.

**Proposition 3.2.** The stationary expected beliefs $W(i)$ in the tree $G = (I, E)$ with two stubborn agents with opinion respectively 0 and 1 are:

$$W(j_i) = \frac{i}{m}, \quad \forall j_i \in J, 0 \leq i \leq m$$

and
3.1 – Some key examples

\[ W(k) = W(j_i) \quad \forall k \in J_i, 0 \leq i \leq m \] (3.5)

**Proof.** Since the set of neighbors of every \( k \in J_i \) is contained in \( J_i \cup \{j_i\} \), through the corollary \ref{corollary:neighborhood} it immediately follows:

\[ W(k) = W(j_i) \quad \forall k \in J_i, 0 \leq i \leq m \]

that proves (3.5).

For the nodes in \( J \) we can repeat the same argument as in section 3.1.1 on page 20 to prove:

\[ W(j_i) = \frac{i}{m} \quad 0 \leq i \leq m, j_i \in J \]

and therefore all the expected beliefs are computed.

Clearly the same result can be interpreted through the electrical analogy. In fact the edges that are
not in a path between two stubborn nodes should have resistance equal to 0 and the same potential of their neighbors. See figure 3.4 in which we draw the resistances and we use the same color for nodes with same potential.

\[ S_0 = J_0 \quad J_5 = S_1 \]

Figure 3.4. The tree of the examples. Nodes with same color have same potential

3.1.3 HIC on trees with one stubborn agent

Consider the binary homogeneous gossip model on a tree \( G = (I, E) \) in which there is a single type 0 stubborn node \( S^0 = \{s_0\} \), \( W(s_0) = 0 \). We will explicitly compute the solution of the OSAP problem.

Suppose first that the stubborn node is a leaf. From the previous section we know that if \( S^1 = \emptyset \) all the expected beliefs will be equal to 0 (thanks to (3.5)) and therefore also the sum of them. If instead \( S^1 = \{l\} \) and we want to solve problem (2.13), the solution is trivial: choose as \( l \in S^1 \) the unique neighbor \( l \) of \( s_0 \). In fact in this case we will end up with Harmonic Influence Centrality equal to:

\[ H(l) = |I| - 1 \]

In fact no one of the nodes in the tree will be able to access the information stored in \( s_0 \) since all the paths will be absorbed in \( s_0 \).
If we remove the hypothesis that $s_0$ is located in a leaf then if the neighbor set of $s_0$ is such that $|N_{s_0}| \geq 2$, node $s_0$ clearly divides the tree in $|N_{s_0}|$ regions. In fact consider the restriction $G$ to $I \setminus \{s_0\}$ we have that $G|_{I \setminus \{s_0\}} = \bigcup T_i$ where $T_i, i \in N_{s_0}$ are trees and are exactly $|N_{s_0}|$. To solve the problem we should simply choose the tree $T_i = (I_i, E_i)$ that has the biggest number of nodes:

$$\arg\max_{k \in I \setminus S_0} H(k) = \arg\max_{k \in N_{s_0}} |I_k|$$

See Figure 3.5 where we see an example in which $N_{s_0} = \{p, q\}$. Then $p$ and $q$ are the only candidates to maximize the harmonic influence centrality in the graph of the example:

$$\arg\max_{k \in I \setminus S_0} H(k) = \arg\max_{k \in \{p, q\}} H(k) = q$$

since $H(q) = |I_q| = 7$ and $H(p) = |I_p| = 3$.

### 3.1.4 HIC on lines with stubborn agents in the leaves

We will now solve the Optimal Stubborn Agent Placement problem on a line $L_N$ with $N + 1$ nodes. Label the line with numbers from 0 to $N$. In this case we consider the two leaves stubborn nodes of type 0, i.e. $S^0 = \{0, N\}$. See figure 3.6. To solve the OSAP problem we need to know the maximum of the Harmonic Influence Centrality.
Proposition 3.3. The Harmonic Influence Centrality on a line $L_N$ where $\{0, N\} = S^0$ is equal to:

$$H(i) = \frac{N - 1}{2} \quad \forall i \in I \setminus S^0$$

Proof. Wherever we put the type 1 stubborn node $l$ the following situation arises: the graph is composed by two lines that have a difference of potential in their extremes equal to 1 (see again Figure 3.6). These two lines have $N_1 + 1$ and $N_2 + 1$ agents with the property that $N = N_1 + N_2 + 1$.

We want to compute the sum of the expected beliefs and this is what we have already done for a line in (3.3). We obtain for the new graph:

$$\sum_{0 \leq i \leq N} W(i) = \frac{N_1 + 1}{2} + \frac{N_2 + 1}{2} - 1 = \frac{N - 1}{2}$$

(3.6)

as we can check this result does not change with respect to $N_1$ and $N_2$ and therefore does not depend on the choice of $l$.

We remark that we have put the minus 1 in second part of equation (3.6) since we are counting twice the node stubborn of type 1 (see again figure 3.6), that has a fixed voltage equal to 1.

Hence we obtained that the harmonic influence centrality in this case is constant for all $i \in I \setminus S^0$. A consequence is that the solution of the problem (2.14) is the whole set $I \setminus S^0 = \text{argmax}_{l \in I \setminus S^0} H(l)$. We have obtained that on a line we can put the stubborn node of type 1 wherever we want to solve the OSAP problem.
3.2 A key property

In view of the examples above, we move on to consider general cases. We first present a direct consequence of theorem 2.2 on page 10 very useful to compute the harmonic influence centrality, thanks also to the electrical analogy presented in section 2.4 on page 15.

Since from now on we will need to evaluate the harmonic influence centrality $H(l)$ and the potentials $W(i)$ over different subgraphs $\tilde{G} = (\tilde{I}, \tilde{E})$, when needed we will use the notation and $H_{\tilde{G}}(l)$ and $W_{\tilde{G}}(i)$ to emphasize the dependance on the graph of such properties. Then

$$H_{\tilde{G}}(l) := \sum_{i \in \tilde{I}} W_{\tilde{G}}(i)$$

is the harmonic influence centrality evaluated in $l \in \tilde{I}$ of problem (2.14) restricted from $G$ to $\tilde{G}$. We will call $R_{\tilde{G}}(i)$ the equivalent resistance in $\tilde{G}$ between between $i$ and type 0 stubborn nodes (which may be thought as glued together, see [20], chap. 9).

Consider the binary homogeneous gossip model with two stubborn agents sets $S^0$ and $S^1$ on a graph $G(I, E)$. Let $W$ be the potential function associated with this model, such that $x_s = 1$ $\forall s \in S^1$ and $x_s = 0$ $\forall s \in S^0$.

Consider now the same model but let $\hat{W}$ be another potential function such that $\forall s \in S^0$, $x_s = w_0$ and $\forall s \in S^1$, $x_s = w_1$. Then for every node $i \in I$ it holds:

$$\hat{W}(i) = W_0 + (w_1 - w_0)W(i) \quad \forall i \in I.$$

Notice that this is a general result not only true for trees.

From this property we can simplify the computation of the potentials and therefore of the solution of problem (2.14) in trees. In fact we are able to recompute easily the potentials in a part of the tree if we change the location of the type 1 stubborn node.

We show how this can be done with a general example. Let $G = (I, E)$ be an undirected tree and suppose we want to compute the harmonic influence centrality in the nodes (see figure 3.7 on the next page). Take $i, j \in I \setminus S^0$ such that $(i, j) \in E$. Assume there exist two nodes $v', v'' \in S^0$ such that the simple path between $v'$ and $v''$ contains both $i$ and $j$. Call $\tilde{G} = (I, E \setminus (i, j))$. Then $\tilde{G} = T_{ij} \cup T_{ji}$ where $T_{ij} = (I_{ij}, E_{ij})$ and $T_{ji} = (I_{ji}, E_{ji})$ are the connected components of $\tilde{G}$ and $i \in I_{ij}$, $j \in I_{ji}$.

$T_{ij}$ and $T_{ji}$ are trees. We use the notation $T_{ij}$ to remember that it is one of the two trees obtained disconnecting the edge $(i, j) \in E$ and $i \in I_{ij}$, $j \notin I_{ij}$.

Assume to know the voltage function $V$ on $T_{ij}$ having $i$ as stubborn node of type 1. Moreover assume to know $H_{T_{ij}}(i)$ and $R_{T_{ij}}(i)$. Consider now $T'_{ij} = (I_{ij} \cup j, E_{ij} \cup (i, j))$.

We can compute the new voltage function $\hat{W}$ on the tree $T'_{ij}$ that has node $i$ as stubborn of type 1. In fact $w_j$ is given by the Ohm’s law:
Figure 3.7. From $G$ to $\tilde{G}$ and then from $T_a$ to $T^*_ij$. Blue nodes are type 1 stubborn ($V=0$), the ones with red stripes are the type 1 stubborn ($V=1$).

\[
w_j = \frac{R_T(i)}{R_T(i) + 1}
\]

and hence it holds:

\[
\hat{W}(i) = w_j W(i) \quad \forall i \in T_{ij}.
\]

Now $H_{T^*_ij}(j)$ can be computed using the following formula:

\[
H_{T^*_ij}(j) = w_j H_{T_{ij}}(i) + 1,
\]

Notice moreover that the following holds:

\[
H(j) = H_{T^*_ij}(j) + H_{T_{ji}}(j) - 1.
\]

The computation \((3.8)\) will be used many times for its simplicity: it allows to compute the harmonic
centrality in a node simply through the harmonic centrality of a neighbor. Notice that the harmonic influence centrality is a sum of potentials that are themselves another potential function, computed through a system of equations. This computation will reflect the core of the algorithm proposed in chapter 4 on page 41.

### 3.3 Properties of potentials

We present now two results about resistances and potentials useful to prove the results in the next section. This section can be avoided during a first reading.

**Proposition 3.4.** Let $G = (I, E)$ be an electrical network with $N$ nodes and where each $(i, j) \in E$ has resistance $R(i, j) = 1$. Then for every two nodes $a$ and $b$ the equivalent resistance $R(a, b)$ between $a$ and $b$ satisfies the following inequality:

$$R(a, b) \leq N - 1$$

**Proof.** We only sketch the proof; for further details see chapter 9 in [20]. This simple electrical property derives from the property of parallel and series wires. Each edge has a resistance equal to 1. The greatest resistance that can be made with $N - 2$ internal nodes is the series of the $N - 2$ internal nodes, with $N - 2$ edges of unitary resistance. Then the equivalent resistance of the network would be:

$$R(a, b) = N - 1$$

If we add an edge the total resistance could not increase because we have added a parallel section in the network (see also Corollary 9.13. in [20]).

Notice that this is a general result and we did not restrict to the case of trees, differently from the following one:

**Proposition 3.5.** Consider the binary homogeneous gossip model with stubborn agents on a tree $G = (I, E)$. Consider the two stubborn nodes $a$ and $b$. $a$ is gluing the set $S^0$ as explained in section 2.4 on page 15 and $S^1 = \{b\}$. Then the equivalent resistance between $a$ and $b$ (see figure 3.8) satisfies:

$$R(a, b) \leq 2 \sum_{i \in I} W(i) - 1$$

(3.10)
Proof. We have to prove the inequality for a tree where $b$ is a single node with potential 1 and $a$ is gluing many nodes with same potential equal to 0. Then the potential difference between $a$ and $b$ is constant equal to 1.

Assume that all the nodes in the network are in a simple path between $a$ and $b$. This fact does not compromit the result: in fact a node that is not inside a simple path between $a$ and $b$ does not modify the resistance $R(a, b)$, while the $\sum_{i \in I} W(i)$ cannot decrease since by definition $W(i) \geq 0 \ \forall i \in I$.

We study the parallel and series cases and then we combine them.
First consider a series of \( n \) edges (figure 3.9 on the facing page (A)). We have that \( R(a, b) = n \) and \( \sum_{i \in I} W(i) = \frac{n + 1}{2} \). Hence we get:

\[
2 \cdot \frac{n + 1}{2} - 1 = n \geq n = R(a, b),
\]

satisfying 3.10 on page 29.

Consider instead a network consisting in a parallel of two lines (figure 3.9 on the facing page (B)): call \( n_1 \) and \( n_2 \) the length of the two branches. Then we have \( R(a, b) = \frac{n_1 n_2}{n_1 + n_2} \) and

\[
\sum_{i \in I} W(i) = \frac{n_1 + 1}{2} + \frac{n_2 + 1}{2} - 1 = \frac{n_1 + n_2}{2}.
\]

Since \( n_1 n_2 \leq n_1^2 + n_2^2 - n_1 - n_2 = (n_1 + n_2)^2 \) we easily show that:

\[
R(a, b) = \frac{n_1 n_2}{n_1 + n_2} \leq \frac{(n_1 + n_2)^2 - n_1 - n_2}{n_1 + n_2} = n_1 + n_2 - 1 = 2 \sum_{i \in I} W(i) - 1
\]

for every integer \( n_1, n_2 \geq 0 \).

Now consider a parallel between two equivalent resistance \( R_1 \) and \( R_2 \) respectively of node set \( A \) and \( B \)

\[
A \subseteq I, B \subseteq I, A \cap B = \emptyset
\]

Suppose \( R_1 \) and \( R_2 \) satisfy 3.10 on page 29 (see figure 3.9 on the facing page (C)). Then it holds

\[
R(a, b) = \frac{R_1 R_2}{R_1 + R_2} \leq R_1 \leq 2 \sum_{i \in A} W(i) - 1 \leq 2 \sum_{i \in I} W(i) + 1
\]

proving the fact.

Last we prove that combining in series two general equivalent resistances the result is still valid. See figure 3.9 on the preceding page (D). Consider two sections with equivalent resistances \( R_1 \) and \( R_1 \) respectively of node set \( A \) and \( B \) connected by a node \( x \) with potential \( W(x) \), that is equal to:

\[
W(x) = \frac{R_1}{R_1 + R_2}.
\]
Suppose that $R_1$ and $R_2$ satisfy 3.10 on page 29 if the potential difference in the two extremities of this subnetworks is 1. Call these potentials changed $V(i)$.

From proposition 3.2 on page 27 we can compute the potential $W(i)$ in terms of $V(i)$. We have that

$$W(i) = W(x)V(i) \quad \text{if } i \in A$$

and

$$W(i) = (1 - W(x))V(i) + W(x) \quad \text{if } i \in B.$$ .

Then it holds that

$$\sum_{i \in A} W(i) = W(x) \sum_{i \in A} V(i) \geq W(x) \frac{1}{2}(R_1 + 1)$$

and

$$\sum_{i \in B} W(i) = (1 - W(x)) \sum_{i \in B} V(i) + W(x)|B|$$

Using proposition 3.4 on page 29 we have that $|B| \geq R_2 + 1$ and hence:

$$\sum_{i \in B} W(i) \geq \frac{1}{2}(R_2 + 1)(1 - W(x)) + W(x)(R_2 + 1) = \frac{1}{2}(R_2 + 1)(1 + W(x))$$

obtaining:

$$2 \sum_{i \in I} W(i) - 1 \geq \frac{1}{2}(R_2 + 1)(1 + W(x)) + W(x)\frac{1}{2}(R_1 + 1)$$

With some algebraic passages we get to

$$2 \sum_{i \in I} W(i) - 1 \geq R_1 + R_2 = R(a, b)$$

that proves the proposition.  

\[ \square \]
Therefore we have found that the sum over all the opinion inside the network is related with the equivalent resistance between the two sets of stubborn nodes. Notice that the sum of all the voltages in the electrical network is the Harmonic Influence Centrality in \( l \), if we consider \( S^1 = \{l\} \).

### 3.4 The Harmonic Influence Centrality on trees

We now want to characterize the Harmonic Influence Centrality on trees and in particular the solution of the Optimal Stubborn Agent Placement problem presented in (2.14):

\[
\arg\max_{l \in I \setminus S^0} H(l).
\] (3.11)

The main result of the chapter is theorem 3.10 on page 38.

Recall some notions of basic mathematical analysis (proofs can be found on [26]).

**Lemma 3.6.**

- A continuous, twice differentiable function \( f(x) \) over \([a, b] \subset \mathbb{R}\) is convex if \( \frac{d^2 f(x)}{dx^2} \) is nonnegative in \([a, b]\).
- If \( f(x) \) and \( g(x) \) are convex functions over \([a, b] \subset \mathbb{R}\), then \( h(x) = f(x) + g(x) \) is still convex over \([a, b]\).
- If \( f \) is a convex function on \([a, b] \subset \mathbb{R}\), then its maximum is in \( a \) or in \( b \).

We provide some results that shed some light on problem (2.14) giving some property of the solution and on the harmonic influence centrality for trees.

From the considerations in equation (3.5) in section 3.1.2 on page 22 we can directly deduce the following result:

**Lemma 3.7.** Consider a tree \( G = (I, E) \) and a node \( i \in I \setminus S \) not in a simple path between two stubborn nodes. Then \( i \) has the same expected belief of the closest node that is in a simple path between two stubborn nodes.

We present now some new notation useful for explaining the next result.

Define a **restriction** \( G|_J \) of a graph \( G = (I, E) \) to the set of nodes \( J \subseteq I \) as:

\[
G|_J = (J, E \cap (J \times J))
\] (3.12)

Fix \( S^0 \subset I \) the set of type 0 stubborn nodes. Call \( G' = G|_{I \setminus S^0} \). Then
\[ G' = \bigcup_{h=1}^{n} T_h \quad n \geq 1 \]

where \( \{T_h = (J_h, E_h)\}_{h \in \{1, \ldots, n\}} \) are the disjoint connected components of \( G' \). We have that \( \{T_h\}_{h \in \{1, \ldots, n\}} \) are trees and \( G' \) is a forest since \( G \) is an undirected tree.

For every \( h \in \{1, \ldots, n\} \) define the set \( S^0_h \):

\[ S^0_h = \{v \in S^0 | \exists w \in J_h : (w, v) \in E\}. \]

\( S^0_h \) represents the set of type 0 stubborn nodes that are neighbors in the original graph to nodes in \( T_h \).

Call \( T^*_h = G|_{S^0_h \cup J_h} \). \( T^*_h \) is therefore \( T_h \) augmented with its type 0 stubborn neighbors in the original graph \( G \). \( T^*_h \) is still a tree.

In figure 3.10 on the next page it is shown in an example the derivation of \( T_h \) and \( T^*_h \) from a tree \( G = (I, E) \). Blue nodes represents the type 0 stubborn agents.

We introduce this notation since we can state the following property for the optimal stubborn agent problem:

**Proposition 3.8.** Consider the binary homogeneous gossip model with stubborn agent set \( S \) over a tree \( G = (I, E) \) with harmonic influence centrality \( H \). Consider now the same model on \( \hat{G} = \bigcup_{h=1}^{n} T^*_h \) with harmonic influence centrality \( \hat{H} \). Then:

\[ \hat{H}(l) = H(l) \quad \forall l \in I \]

**Proof.** We have mapped the graph \( G \) into a forest of trees \( T^*_h \), in which the only nodes \( i \in I \) repeated in different trees are by construction some stubborn nodes \( s \in S^0 \). For those nodes, since they are stubborn, an harmonic influence centrality is always zero in each graph by definition:

\[ \hat{H}(s) = H(s) = 0 \quad \forall s \in S^0 \]

On the other nodes that are not repeated and not stubborn \( i \in I \setminus S^0 \), \( i \in I_h \) for a certain \( h \), we have to prove that the harmonic influence centrality of \( i \) on the whole graph \( G \) is equal to the harmonic influence centrality of \( i \) on the tree \( T^*_h \).

We recall that a node \( i \) has influence on a node \( j \) if a path exists from \( j \) to \( i \). Such path in \( G \) cannot
pass from a node stubborn. In fact stubborn nodes \( s \in S^0 \) are absorbing, and therefore as remarked in 3.1 edges \((s,j), j \in I\) can be removed without modifying the model. Hence since in a tree there exists one and only one path between two nodes, if there is a stubborn node inside a path between \( j \) and \( i \), then \( j \) cannot be influenced by \( i \).

Obviously also a node in \( T^*_k \) and a node in \( T^*_h \) for all \( k \neq h \) cannot influence each other since by definition the two trees are disconnected. Then

\[
\hat{H}(l) = H(l) \quad \forall l \in I \setminus S^0
\]

that proves the proposition.
With this proposition we $G = (I, E)$ are now allowed to study the harmonic influence centrality on the different subtrees $T_h^n$, $h \in \{1, \ldots, n\}$, where the stubborn nodes $s \in S^0$ are always located in the leaves of the subtrees. Moreover it holds:

$$\max_{l \in I \setminus S^0} H(l) = \max_{h \in \{1, \ldots, n\}} \max_{l \in I_h} \tilde{H}(l)$$

Thanks to this property we will just consider from now on trees with stubborn agents in the leaves.

**Proposition 3.9.** Assume that $G = (I, E)$ is an undirected tree. Let $S^0 \subset I$ be the set of type 0 stubborn nodes. Let $s', s'' \in S^0$ and assume that the unique simple path $s'_{i_1}i_2 \ldots i_k s''$ connecting $s'$ to $s''$ in $G$ is such that $K = \{i_1, \ldots, i_k\} \subseteq I \setminus S^0$. Let $K' = \{l \in K | d_l \geq 3\}$ where $d_l$ is the degree of $l$. If $K' \neq \emptyset$ then:

$$\arg \max_{l \in K} H(l) = \arg \max_{l \in K'} H(l)$$

**Proof.** Consider a candidate point $j_x$ for the maximization inside the path $s'_{i_1}i_2 \ldots i_k s''$ such that its degree is equal to 2 and exists at least a node with degree three. Then $j_x \in K \setminus K'$. Note that it cannot have degree 1 or 0 since it is in a simple path between two nodes. This point has to be inside a line $(j_1, j_2, \ldots, j_x, j_{x+1}, \ldots, j_n)$ with $n$ nodes (all with degree 2). Clearly $1 \leq n \leq k - 1$ and $1 \leq x \leq n$.

If $S^1 = \{j_x\}$ then it divides the line itself in two subsets $X = \{j_1, \ldots, j_{x-1}\}$ and $Y = \{j_{x+1}, \ldots, j_n\}$, $|X| + |Y| + 1 = n$. See figure [3.11]

Call $a = N_{j_1} \setminus \{j_2\}$ and $b = N_{j_n} \setminus \{j_{n-1}\}$. $a$ and $b$ are the extremities of the line and they are two nodes with degrees greater than or equal to 3 or a stubborn node and a node with degree greater
than or equal to 3. Consider the tree $T_{a_j}$ with node set $\alpha \cup \{a\}$ and $T_{b_j}$ with node set $\beta \cup \{b\}$ (see again figure 3.11 on the facing page) and consider the equivalent resistance $R_\alpha := R_{T_{a_j}}(a)$ and $R_\beta := R_{T_{b_j}}(b)$. Using this notation for the whole network we get:

$$H(j_x) = W(a)H_{T_{a_j}}(a) + W(b)H_{T_{b_j}}(b) + \frac{1 + W(a)}{2}|X| + \frac{1 + W(b)}{2}|Y| + 1$$  \hspace{1cm} (3.13)

We have that $W(a) = \frac{R_\alpha}{R_\alpha + |X| - 1}$ and $W(b) = \frac{R_\beta}{R_\beta + |Y| - 1}$. $H(j_x)$ depends on the choice of $x$, $1 \leq x \leq n$ and:

$$H(j_x) = F(x) + G(x) + \text{constant}$$

Where:

$$F(x) = \frac{R_\alpha(H_{T_{a_j}}(a) + \frac{x-1}{2})}{R_\alpha + x}$$

and

$$G(x) = \frac{R_\beta(H_{T_{b_j}}(b) + \frac{(n-x)-1}{2})}{R_\beta + (n - x - 1)}$$

$F(x)$ is convex. In fact:

$$\frac{\partial^2 F}{\partial x^2} = \frac{2R_\alpha(H_{T_{a_j}}(a) - \frac{R_\alpha-1}{2})}{(R_\alpha + x)^3}$$

The following correspondence holds:

$$\frac{\partial^2 F}{\partial x^2} \geq 0 \iff R_\alpha \leq 2H_{T_{a_j}}(a) + 1$$

Using proposition \[3.5 on page 29\] the second condition is always satisfied and therefore the function $F$ is convex.

With the same argument we prove that $G(x)$ is also convex. For lemma \[3.6 on page 33\] the harmonic influence centrality $H(j_x)$ is a convex function.

Hence from theorem \[3.6\] we conclude that the value in $a$ or in $b$ of the HIC is greater or equal than the value on internal nodes of the line.
We can now present the main result of the chapter that outlines the characterization of the harmonic influence centrality for trees, and the solution of the optimal stubborn agent placement problem in 2.1 on page 14.

**Theorem 3.10.** Consider the binary homogeneous gossip model with type 0 stubborn agent set $S^0$ over a tree $T = (J, E)$ with stubborn agent in the leaves. Then one of the following cases holds:

1. If $S^0 = \{s\}$ let $i \in J$ be the unique node such that $(i, s) \in E$, then:

$$\arg\max_{l \in J} H(l) = i$$

2. a. If $|S^0| \geq 2$, let $K = \{i \in J | \exists s', s'' \in S^0$ such that $i$ is inside a simple path in between $s'$ and $s''\}$. Then:

$$\arg\max_{l \in J} H(l) = \arg\max_{l \in K} H(l)$$

b. Moreover if $K' := \{l \in K | d_l \geq 3\} \neq \emptyset$ then

$$\arg\max_{l \in J} H(l) = \arg\max_{l \in K'} H(l)$$

**Proof.** We already proved the point 1. in the example 3.1.3 on page 24.

To prove 2.a we have to show that if a node is not in a simple path between two stubborn nodes then it cannot maximize the centrality.

Consider a node $i \in K$ and let $j$ a node in $J$ such that all the paths to stubborn nodes pass through $i$. Then we have the situation in Figure 3.12 on the next page. Let $X_2 = \{k \in J | \text{all simple paths from } k \text{ to a stubborn node pass through } j\}$ and $X_1 = \{j\} \cup \{k \in J | \text{all simple paths from } k \text{ to a stubborn node pass through } i \text{ and not through } j\}$ and $Y = \{k \in J | k \text{ is in a simple path between two stubborn nodes}\}$. We now show that in this case $H(i) > H(j)$. Call $V(k)$ the potential function if $S^1 = \{i\}$ and $W(k)$ the potential if $S^1 = \{j\}$.

Then

$$H(i) = \sum_{y \in Y} V(y) + |X_1| + |X_2|$$

(3.14)
and

\[ H(j) = \sum_{y \in Y} W(y) + \sum_{x \in X_1} W(x) + |X_2| \]  

(3.15)

Then (see (3.7)):

\[ \sum_{y \in Y} W(y) < \sum_{y \in Y} V(y) \]  

(3.16)

and

\[ \sum_{x \in X_1} W(x) \leq |X_1| \]  

(3.17)

by definition. Using (3.16) and (3.17) in (3.15) we get to the conclusion \( H(j) < H(i) \).

Thanks to the point 2.a and the proposition 3.9 on page 36 the point 2.b result proved.

\[ \square \]

See on figure 3.13 the application of the theorem to the three different cases that can arise from a \( T \). We have that blue nodes are type 0 stubborn and the ones with green stripes are subset of the nodes that can be solutions of the OSAP problem thanks to the theorem just proved.

See also for example figure 3.10 on page 35 where there is a tree \( G = (I, E) \). Applying the previous proposition we have that the candidate nodes for solving problem (2.14) are reduced from 22 to 4

39
We have obtained that it is sufficient to compute the harmonic influence centrality for the maximization only in some agents, that can be very few with respect to the total number of nodes. Moreover we can compute the harmonic influence centrality for every candidate node since we are able to compute resistances and therefore potentials in the network. In particular it is sufficient to compute for each of these nodes the equivalent network created if they were stubborn of type 1 and so evaluating the HIC. The last passage is to compute the maximum between all of those nodes and the optimal stubborn agent placement problem is solved.
Chapter 4

A distributed Message Passing Algorithm for trees

4.1 Message Passing Algorithms

Message passing algorithms are distributed algorithms related to a graph, in which nodes are objects with computational ability and can send and receive information to their neighbors. The links represent therefore communication channels and the packets of information that flow through the links are called messages. Messages can be bytes, complex data structures, or even segments of code.

This kind of algorithm has been already widely studied (see [19], [24], [25] and [29]). They are commonly used in artificial intelligence and information theory and has demonstrated empirical success in numerous applications including low-density parity-check codes, turbo codes, free energy approximation, and satisfiability problem (see [6]).

Message passing algorithms are built to compute in a distributed way some quantities of interest on trees. The algorithm designer only chooses the algorithm that each node has to perform. All nodes run the same algorithm simultaneously. The system must work regardless of the structure of the network. The algorithm works by passing messages along the edges between the nodes. These messages contain the influence that a node exerts on another.

We outline the structure of a message passing algorithm on a tree. Define a node $i$ in the graph as a root.

In the first step, messages are passed inwards: starting at the leaves, each node passes a message along the unique edge towards the root node. The tree structure guarantees that it is possible to obtain messages from all other adjoining nodes before passing the message on. This continues until
4 – A distributed Message Passing Algorithm for trees

the root $i$ has obtained messages from all its adjoining nodes.
The second step involves passing the messages back out: starting at the root, messages are passed in
the reverse direction. The algorithm is completed when all leaves have received their messages.
Note that every node in the graph can be chosen as a root.

We will usually shortened from this point message passing algorithm with MPA

4.2 MPA for computing the harmonic influence centralities on trees

Our purpose is to compute in a distributed way the harmonic influence centrality without a leader
that knows the topology of the graph. We will build a message passing algorithm that allows each
node $i \in I$ to compute exactly its own harmonic influence centrality $H(i)$. To do so we exploit
the interesting properties already shown in section 3.2. Every node inside the graph is able to
communicate only with its neighbors.

We recall the notation. Consider the binary homogeneous gossip model over an undirected tree
$G = (I, E)$ with stubborn agents $S^0$. If we consider an undirected edge $(i, j) \in E$ and we remove it we
obtain two trees, $T_{ij} = (I_{ij}, E_{ij})$ and $T_{ji} = (I_{ji}, E_{ji})$. $T_{ij}$ and $T_{ji}$ are connected components and we
assume $i \in T_{ij}, j \in T_{ji}$. We use the notation $T_{ij}$ to remember that it is one of the two trees obtained
disconnecting the edge $(i, j) \in E$ and $i \in T_{ij}, j \notin T_{ij}$. We also define $T^*_{ij} := (I_{ij} \cup j, E_{ij} \cup (i, j))$ the
tree $T_{ij}$ augmented with the node $j$.

Let $H_T(i)$ and $R_T(i)$ the harmonic influence centrality of $i$ and the equivalent resistance between
stubborn nodes and $i$, both measured on a subgraph $T$ of $G$.

We have already seen in lemma 3.2 on page 27 and through equation (3.8) how the harmonic influence
centrality can be computed in a node on a particular subtree thanks to the information received
by neighbors. Messages sent on an edge $(i, j) \in E$ between the nodes $i$ and $j$ are related to the
subgraphs $T_{ij}$ and $T_{ji}$.

Consider a generic node, $i \in I \setminus S^0$, and call it the root. Let $N_i$ be the set of its neighbors. If
we remove the undirected edges $(i, j), \forall j \in N_i$ we obtain the node $i$ and $|N_i|$ trees, denoted by
$T_{ji}, j \in N_i$. See figure 4.1 on the next page. Each tree $T_{ji}$ can contain or not stubborn nodes in $S^0$.

Consider a tree $T_{ji}$ that contains at least a stubborn node. Assume that node $j$ has computed
$H_{T_{ji}}(j)$ and the equivalent resistance $R_{T_{ji}}(j)$.
If node $j$ communicates these quantities to node $i$ through link $(i, j) \in E$ then node $i$ can compute
$H^*_{T_{ji}}(i)$ and the equivalent resistance $R^*_{T_{ji}}(i)$. 

42
4.2 – MPA for computing the harmonic influence centralities on trees

\[ R_{T_{ji}}(i) = R_{T_{ji}}(j) + 1, \]  
\[ H_{T_{ji}}(i) = \frac{R_{T_{ji}}(j)}{R_{T_{ji}}(i)} H_{T_{ji}}(j) + 1, \]

Figure 4.1. The general structure seen by the root \( i \)

because \( R(i, j) = 1 \) by definition and because \( R(i, j) \) and \( R_{T_{ji}}(j) \) are resistances in series. Recalling equation (3.8):

\[ H_{T_{ji}}(i) = \frac{R_{T_{ji}}(j)}{R_{T_{ji}}(i)} H_{T_{ji}}(j) + 1, \]  

as explained in section 3.2 on page 27. \( H_{T_{ji}}(i) \) represents the harmonic influence that node \( i \) has over a tree \( T_{ji}^* \) that is a subgraph of \( G \).

Consider now instead a tree \( T_{ji} \) that do not contains stubborn nodes \( s \in S^0 \). Assume that node \( j \) knows \( H_{T_{ji}}(j) \) and the equivalent resistance \( R_{T_{ji}}(j) \). Those values will be:

\[ R_{T_{ji}}(j) = 0 \]

since there are not stubborn nodes in \( T_{ji} \). Hence from example 3.1.3 on page 24.
\[ H_{T^*_ji}(j) = |I_{ji}| \]

Then node \( i \) can compute \( H_{T^*_ji}(i) \) and the equivalent resistance \( R_{T^*_ji}(i) \).

\[ R_{T^*_ji}(i) = 0 \]

(4.3)

since there are no stubborn nodes also in \( T^*_ji \).

\[ H_{T^*_ji}(i) = H_{T^*_ji}(ji) + 1 = |I^*_ji|. \]

(4.4)

When the root \( i \) has collected all the messages related to all its neighbors \( j \in N_i \) then the harmonic influence centrality for the whole graph is simply:

\[ H(i) = \sum_{j \in N_i} H_{T^*_ji}(i) - |N_i| + 1 \]

The minus \( |N_i| \) is added because node \( i \in S^1 \) will have average asymptotic opinion counted as many times as are the neighbors of \( i \). Remember also that this computation is possible since all trees \( T^*_ji \) do not have any common node.

Notice that we can take any node \( i \) as root. To do the final computation of the harmonic influence centrality node \( i \) needs all the messages from its neighbors \( N_i \). Instead node \( i \) can send to a neighbor \( j \) information about \( T_{ij} \) before receiving messages from \( j \) itself. For sending a message through a link \( (i,j) \in E \) node \( i \) needs to know all the messages from all the \( (k,i) \in E, k \neq j \).

Notice that we have to distinguish if a tree contains or not a stubborn node. This is easily achievable adding a new variable, that we will call flag \( F \). This variable can only assume value 0 or 1. A values for the flag \( F_{T_{ij}} = 0 \) means that tree \( T_{ij} \) contains at least one stubborn \( s \in S^0 \), while \( F_{T_{ij}} = 1 \) means that in tree \( T_{ij} \) there are not.

The messages that has to flow from node \( i \) to node \( j \) through a communication channel \( (i,j) \in E \) are therefore:

- A flag \( F_{T_{ij}}(i) \). \( F_{T_{ij}}(i) \) is equal to 0 if \( \exists s \in S^0 \cap T_{ij} \). \( F_{j}(i) \) is equal to 1 if this does not hold and therefore there are no stubborn nodes in \( T_{ij} \).
- The equivalent resistance \( R_{T_{ij}}(i) \) in \( T_{ij} \) between type 0 stubborn nodes and \( i \) in the tree \( T_{ij} \).
- The harmonic influence centrality \( H_{T_{ij}}(i) \) of \( i \) in the graph \( T_{ij} \).
Each node has to receive messages from all its neighbors and send messages to all its neighbors. In particular \( i \) sends a message on an edge \((i, j) \in E\) when he received all the messages from edges \((i, k), k \in N_i, k \neq j\).

To characterize the algorithm, we need to give the initial conditions at the leaves. In a message passing algorithm the messages are started from the leaves. Notice that also stubborn nodes in \( S^0 \) will be considered leaves from proposition 3.8 on page 34 and since they have potential equal to 0. Then a leaf \( i \notin S^0, \vert N_i \vert = 1 \) does not have to wait messages to send to its only neighbor \( j \) and the messages will be:

\[
F_{T_{ij}}(i) = 1, \quad R_{T_{ij}}(i) = 0, \quad H_{T_{ij}}(i) = 1
\]

Also a stubborn \( i \in S^0 \) does not have to wait messages and the messages to send to its neighbors \( j \) will be also trivial:

\[
F_{T_{ij}}(i) = 0, \quad R_{T_{ij}}(i) = 0, \quad H_{T_{ij}}(i) = 0
\]

Once a node \( i \) knows all the messages from all the \((k, i) \in E, i \neq j\), node \( i \) can compute the messages to send through a link \((i, j) \in E\) to \( j \):

\[
F_{T_{(i)}}(i) = \begin{cases} 
1 & \text{if } i \notin S, \{k \in N_i \setminus \{j\} | F_{T_{(k)}}(k) = 0\} = \emptyset \\
0 & \text{otherwise}
\end{cases} \tag{4.5}
\]

for the definition of the flag. \( i \) has to compute a parallel of the resistances that it has received:

\[
R_{T_{(i)}}(i) = \begin{cases} 
\left( \sum_{k \neq i, F_{T_{(k)}}(k)=0} \frac{1}{R_{T_{(k)}}(k) + 1} \right)^{-1} & \text{if } k \notin S, \{k \in N_i \setminus \{j\} | F_{T_{(k)}}(k) = 0\} \neq \emptyset \\
0 & \text{otherwise}
\end{cases} \tag{4.6}
\]

and finally for the harmonic influence centrality:

\[
H_{T_{(i)}}(i) = \begin{cases} 
1 + \sum_{k \neq i, F_{T_{(k)}}(k)=0} \frac{R_{T_{(k)}}(k)}{R_{T_{(k)}}(k) + 1} H_{T_{(k)}}(k) + \sum_{k \neq i, F_{T_{(k)}}(k)=1} H_{T_{(k)}}(k) & \text{if } k \notin S \\
0 & \text{otherwise}
\end{cases} \tag{4.7}
\]
Hence we have characterized the message passing algorithm over a tree.

4.2.1 The pseudo-code

The general pseudo-code of the algorithm that each node has to perform is reported below. It is a message passing type algorithm.

Every \( i \in I \setminus I^0 \) has to receive and send messages from and to all the nodes \( j \in I \) such that \( (j, i) \in E \).

This is the pseudo-code of the algorithm that each node \( i \in I \) with neighbor set \( N_i \) has to perform in parallel with the other nodes:

1: begin
2: \( M_i = N_i \)
3: if \( i \in S^0 \) then
4: for each \( j \in N_i \) do
5: \( F_{T_{ij}}(i) = 0 \);
6: \( R_{T_{ij}}(i) = 0 \);
7: \( H_{T_{ij}}(i) = 0 \);
8: Send to \( j \) the message \( (F_{T_{ij}}(i), R_{T_{ij}}(i), H_{T_{ij}}(i)) \)
9: end for
10: \( H(i) = 0 \)
11: else
12: while \( M_i \neq \emptyset \) do
13: for each \( j \in M_i \) do
14: if all messages from \( N_i \setminus \{j\} \) are received then
15: \( F_{T_{(j)}}(i) = \begin{cases} 1 \text{ if } \{k \in N_i \setminus \{j\} \mid F_{T_{(k)}}(k) = 0\} = \emptyset \\ 0 \text{ otherwise} \end{cases} \)
16: \( R_{T_{(j)}}(i) = \begin{cases} 0 \text{ if } \{k \in N_i \setminus \{j\} \mid F_{T_{(k)}}(k) = 0\} = \emptyset \\ \left( \sum_{k \neq i, F_{T_{(k)}}(k) = 0} \frac{1}{R_{T_{(k)}}(k) + 1} \right)^{-1} \text{ otherwise} \end{cases} \)
17: \( H_{T_{(j)}}(i) = 1 + \sum_{k \neq i, F_{T_{(k)}}(k) = 0} \frac{R_{T_{(k)}}(k)}{R_{T_{(k)}}(k) + 1} H_{T_{(k)}}(k) + \sum_{k \neq i, F_{T_{(k)}}(k) = 1} H_{T_{(k)}}(k) \)
18: Send to \( j \) the message \( (F_{T_{ij}}(i), R_{T_{ij}}(i), H_{T_{ij}}(i)) \)
19: \( M_i = M_i \setminus \{j\} \)
20: end if
21: end for
22: end while
4.2 – MPA for computing the harmonic influence centralities on trees

23: \[ H(i) = 1 + \sum_{k: F_T(k) = 0} \frac{R_T(k)}{R_T(k)} H_T(k) + \sum_{k: F_T(k) = 1} H_T(k). \]
24: \[ \text{end if} \]
25: \[ \text{end} \]

The harmonic influence centrality of a node \( i \) will be stored in \( H(i) \).
Notice that the algorithm will end only when all messages from all the nodes in \( N_i \) are received.

4.2.2 Time of convergence for trees

We have seen in section 4.2 on page 42 how the algorithm works and why for tree the result of the algorithm processed in \( i \) yields to the harmonic influence centrality of \( i \).

When a node \( i \) has received all messages from its neighbors it can compute the final value of the harmonic influence centrality terminating the algorithm. It is easy to see that the algorithm will end in a tree in finite time. Since there are no loops in a tree, variables are not feed-backed as seen in equations (4.7), (4.6) and (4.5). There is no possibility that a message sent by a node \( i \) will influence node \( i \) itself.

We also want to characterize the time of convergence. Convergence is enforced when a node receives all the messages from its neighbors. Hence we have to study what is the maximum amount of time that nodes have to wait to receive all the messages.

We suppose that a message takes a unit of time to move in the communication channel between a pair of nodes, and we neglect the computation time of a node. Then messages are sent every time unit. Since a node has to wait for messages from the leaves of its neighbors, if a node is further from the leaves then it has to wait more to perform the final computation.

The biggest distance between all pair of nodes in a tree is defined as the diameter of the graph:

\[ \text{diam}(G) := \max_{i,j \in T} \{\text{dst}(i, j)\}. \]

We remember that stubborn nodes divide the graph into subtrees as explained in proposition 3.8 on page 34:

\[ G|_{I \setminus S^0} = \bigcup_{h=1}^n T_h \]

where \( T_h \) are the disjoint connected components of \( G|_{I \setminus S^0} \). \( T_{h}^0 \) are the trees \( T_h \) augmented with type 0 stubborn nodes neighbors in the original graph. Defining the number:
\[ T_{\text{max}} := \max_{h \in \{1, \ldots, n\}} (\text{diam}(T_h)) \]

as the biggest diameter of the subtrees \( T_h \), we have that the maximum time that a node has to wait to receive a message is therefore \( T_{\text{max}} \). Then it holds:

**Proposition 4.1.** The message passing algorithm described in 4.2.1 on page 40 processed at a node \( i \in I \) converges in finite time for every tree \( G(I, E) \) to the Harmonic Influence Centrality \( H(i) \). Moreover the time of converges will be bounded by:

\[ O(T_{\text{max}}) \]

In the worst case scenario we have \( O(|I|) \), where \( |I| \) is the total number of agents in the society.

We have obtained that the distributed algorithm converges, computing the harmonic influence centrality in linear time.

### 4.3 Complexity analysis

We study in this section the complexity of the algorithm in terms of computational and communication cost.

**Proposition 4.2.** The complexity of the MPA in section 4.2.1 on page 40 is described by the following items.

- The communication needed for each node \( i \) in terms of messages is \( O(|N_i|) \).
- The communication needed across the whole network is \( O(|I|) \).
- The number of operations for each node \( i \) is \( O(|N_i|^2) \).
- The number of operations for the whole network is \( O(\sum_{i \in I \setminus S^0} |N_i|^2) \).

We consider as one message the triple \( (F_{T,i}(i), R_{T,i}(i), H_{T,i}(i)) \) sent from node \( i \) to node \( j \). Each node \( i \) has to send and receive a number of messages equal to twice its degree \( |N_i| \):

\[ 2|N_i| = O(|N_i|) \]

We then study the number of messages that are sent across the whole network. The total number of messages sent in the network during the execution of the algorithm are:
4.3 – Complexity analysis

\[ \sum_{i \in I} 2|N(i)| = 4(|E|) = O(|E|) \]

and for a tree we have \( O(|E|) = O(|I|) \).

The cost for each node \( i \) to compute a message for a neighbor \( j \) is proportional to:

\[ O(|N_i|) \]

hence proportional to the degree. Since there are \( |N_i| \) neighbors of \( i \), node \( i \) has to compute \( |N_i| \) different messages. The total computational cost for a node \( i \) is:

\[ O(|N_i| \cdot |N_i|) = O(|N_i|^2) \]

Since all nodes but the stubborn ones have to perform the algorithm the total amount of computation for the whole graph is:

\[ O(\sum_{i \in I \setminus S^0} |N_i|^2) \tag{4.8} \]

that in a tree is always smaller than \( O(|I|^2) \) since it holds:

\[ \sum_{i \in I \setminus S^0} |N_i|^2 \leq (\sum_{i \in I \setminus S^0} |N_i|)^2 \leq (2|E|)^2 = 2(|I| + 1)^2. \]

We have obtained therefore a distributed algorithm that computes the harmonic influence centrality in all the nodes with number of operations increasing quadratically with respect to the number of agents.

Moreover if the node degrees are bounded when \( |I| \) goes to infinity the performance in \( (4.8) \) is linear, i.e. \( O(|I|) \). In fact if \( |N_i| < c \) with \( c \) constant for every \( G \), for every \( i \in I \) then:

\[ O\left( \sum_{i \in I \setminus S^0} |N_i|^2 \right) \leq O\left( \sum_{i \in I \setminus S^0} c^2 \right) \leq O(|I\setminus S^0|^2) \leq O(|I|) \tag{4.9} \]

Notice that in real networks usually the degree is bounded or grows in sub-logarithmical way (see [1]).

The results obtained are very interesting, compared with an already existing algorithm that computes
the harmonic influence centrality (see [33]). We will see why in the next section.

### 4.4 Comparison with the centralized algorithm

In [33] is explained how to compute in a centralized way the harmonic influence centrality. The algorithm has to know the whole topology of the graph, given by the adjacency matrix $A$. This algorithm works exactly for every graph that is connected, not only for trees.

The total complexity of such algorithm is:

$$O((|I| - |S^0|)^3)$$

due in principal to the inversion of a matrix related to the continuous time Markov process described in (2.3) in chapter 2. We will not see how this algorithm works (see [33]) but we compare it with the message passing algorithm proposed in this chapter.

The message passing algorithm has the nice property that is distributed, works in parallel on all the nodes and it is adapted to the graph. Moreover it has a very good performance in terms of number of operations needed in comparison to the centralized algorithm. In fact we proved in section 4.3 on page 48 that the complexity of the MPA is always bounded by $O(|I|^2)$ while for the centralized algorithm the complexity is bounded by $O(|I|^3)$. The counter-part is that the MPA is designed only for trees, that are a very small subset of all the possible graphs.

We summarize the properties of the centralized algorithm proposed in the work [33] with the MPA proposed in this thesis in the following table:

<table>
<thead>
<tr>
<th>centralized algorithm [33]</th>
<th>MPA</th>
</tr>
</thead>
<tbody>
<tr>
<td>centralized</td>
<td>distributed</td>
</tr>
<tr>
<td>works for every graph</td>
<td>works for trees</td>
</tr>
<tr>
<td>complexity of $O((</td>
<td>I</td>
</tr>
</tbody>
</table>

Our algorithm as presented improves the performance of the algorithm in [33]. We also foresee that a smarter implementation could reduce the complexity reached. The details of this implementation and the corresponding analysis are left for future work.
4.5 An example

We implement the code in Matlab, both for the centralized algorithm in [33] and for the distributed algorithm proposed in this chapter. Consider the simple example proposed in figure 4.2, where there is a tree $G = (I, E)$ with 10 nodes in which nodes 1, 2 and 3 are stubborn, hence in $S^0$.

![Figure 4.2. The graph of the example. Nodes 1, 2 and 3 are in $S^0$.](image)

We run both the algorithm, obtaining as expected the same result. Calling $H := [H(i), i \in I]$ the vector of the aggregate results we obtain as centralities:

$$H \approx [0, 0, 0, 4, 4.333, 5.300, 3.737, 2.857, 4.429, 3.263]$$

that are also shown in figure 4.3 on the following page. Hence the maximum of the centrality is in node 6.

Notice that using theorem 3.10 on page 38 in the chapter [3] we can state a-priori that the maximum of the harmonic influence centrality for this tree could only be located in node 4 or node 6.
Figure 4.3. The results of the harmonic influence centrality for the graph in figure 4.2.
Chapter 5

Message Passing Algorithm applied to general graphs

5.1 Generalization of the MPA

Message passing algorithms are designed to work and to give correct results for trees, but they can be applied with some modification also to general graphs. Usually the application is just empirical, without even proof of convergence.

The Message Passing Algorithm described in section 4.2 on page 42 has very good performance as explained in section 4.4 on page 50, but it is designed to compute correctly the Harmonic Influence Centrality only on trees. We will see in this section how to apply the MPA to every graph, with some modification in order to manage the new issues.

The main differences between a tree and a graph are that a general graph can contain cycles and might not contain any leaves. In this case can happen that a node never receives a message that it is waiting for, since there are cycles. For avoiding this case one initializes all variable messages to a certain value and uses the same message definitions for the tree cases, updating all messages at every iteration. Hence every node at every time step sends and receives all the messages. Then we rewrite the algorithm in an iterative way. At every time step messages are spread over the network. This approach does not preclude the convergence in a tree, since there are no cycles. After a time less than or equal to the diameter of the tree all the messages will be equal to the ones in the original MPA algorithm.

This is a common approach for message passing algorithms. Applying a MPA on a general graph does not guarantee that the quantity computed is correct, but sometimes has empirical success (see...
To understand what happens in case of cycles we can use the so called analysis via computation trees (for further details see [29]). We will apply it in our case in section 5.1.2 on page 56 of this chapter.

In symmetric graphs with cycles when we remove an undirected edge \((i, j) \in E\) we are no more obtaining two disjoint connected components \(T_{ij}\) and \(T_{ji}\) as in chapter 4. In fact another path between \(i\) and \(j\) can exist.

Hence the messages \(F_{T_{ij}}(i)\), \(R_{T_{ij}}(i)\) and \(H_{T_{ij}}(i)\) that \(i\) sends to \(j\) will not have the same interpretation as in section 4.2 on page 42 in case of graphs with cycles. We will see the interpretation through computation trees in section 5.1.2 on page 56. To avoid a misinterpretation and for simplicity we change the notation writing:

\[
\begin{align*}
F_{ji} & := F_{T_{ij}}(i) \\
R_{ji} & := R_{T_{ij}}(i) \\
H_{ji} & := H_{T_{ij}}(i).
\end{align*}
\] (5.1)

Moreover since the new messages will vary on time we will use an apex denoting the time step for the iteration \(t\):

\[
\begin{align*}
F_{ji}^{(t)}, \quad R_{ji}^{(t)}, \quad H_{ji}^{(t)}
\end{align*}
\] (5.2)

We have to define the initial conditions for the messages for every node \(i \in I\). At time \(t = 0\) every node \(i \in I\) send to its neighbors \(j \in N_i\) the messages:

\[
\begin{align*}
F_{ji}^{(0)} & = 0, \quad R_{ji}^{(0)} = 0, \quad H_{ji}^{(0)} = 0.
\end{align*}
\]

At every time step \(t \geq 1\) node \(i\) knows all the messages from all the \((k,i) \in E, i \neq j\). Node \(i\) can compute the new messages to send to \(j\) through a link \((i,j) \in E\). The iterative counterpart of (4.5), (4.6) and (4.7) in section 4.2 on page 42 are, respectively:

\[
\begin{align*}
F_{ji}^{(t+1)} = \begin{cases} 1 \text{ if } i \notin S, \{k \in N_i\setminus\{j\}|F_{ik}^{(t)} = 0\} = \emptyset \\
0 \text{ otherwise} \end{cases}
\end{align*}
\] (5.3)

for the definition of the flag,
5.1 – Generalization of the MPA

\[ R_{ji}^{(t+1)} = \begin{cases} \left( \sum_{k \neq i, F_{ik}^{(t)} = 0} \frac{1}{R_{ik}^{(t)} + 1} \right)^{-1} & \text{if } k \notin S, \{k \in N_i \setminus \{j\} | F_{ik}^{(t)} = 0\} \neq \emptyset \\ 0 & \text{otherwise} \end{cases} \]  \hspace{1cm} (5.4)

for the resistance in \( i \) and finally:

\[ H_{ji}^{(t+1)} = \begin{cases} 1 + \sum_{k \neq i, F_{ik}^{(t)} = 0} \frac{R_{ik}^{(t)}}{R_{ik}^{(t)} + 1} H_{ik}^{(t)} + \sum_{k \neq i, F_{ik}^{(t)} = 1} H_{ik}^{(t)} & \text{if } k \notin S \\ 0 & \text{otherwise} \end{cases} \]  \hspace{1cm} (5.5)

for the harmonic influence centrality.

By these definitions of messages we have defined the message passing algorithm for general graphs. We present in section 5.1.1 the pseudo-code of the modified algorithm, in section 5.1.2 on the next page the interpretation of the messages in the new algorithm and in section 5.1.3 on page 58 the complexity analysis and convergence in case of trees.

5.1.1 The pseudo-code

The pseudo-code of the general algorithm that computes the objective function at every point is reported below. The algorithm is iterative and therefore we have to select some stopping criteria. These will be discussed just after the pseudo-code.

The following instructions are performed by each node \( i \in I \) with neighbor set \( N_i \), in parallel with the other nodes:

1: begin
2: \( F_{ji}^{(0)} = 0, R_{ji}^{(0)} = 0, H_{ji}^{(0)} = 0 \)
3: for \( t = 0 \) to stopping criteria do
4: if \( i \in S^0 \) then
5: for each \( j \in N_i \) do
6: \( F_{ji}^{(t+1)} = 0; \)
7: \( R_{ji}^{(t+1)} = 0; \)
8: \( H_{ji}^{(t+1)} = 0; \)
9: Send to \( j \) the message \( (F_{ji}^{(t+1)}, R_{ji}^{(t+1)}, H_{ji}^{(t+1)}) \)
10: end for
11: \( H^{(t+1)}(i) = 0 \)

55
5. Message Passing Algorithm applied to general graphs

12: \textbf{else} \\
13: \quad \textbf{for each } j \in N_i \textbf{ do} \\
14: \quad \quad F_{ji}^{(t+1)} = \begin{cases} 
1 & \text{if } \{k \in N_i \setminus \{j\} | F_{ik}^{(t)} = 0 \} = \emptyset \\
0 & \text{otherwise} 
\end{cases} \\
15: \quad R_{ji}^{(t+1)} = \begin{cases} 
0 & \text{if } \{k \in N_i \setminus \{j\} | F_{ik}^{(t)} = 0 \} = \emptyset \\
\frac{1}{\sum_{k \neq i, F_{ik}^{(t)} = 0} R_{ik}^{(t)} + 1} & \text{otherwise} 
\end{cases} \\
16: \quad H_{ji}^{(t+1)} = 1 + \sum_{k \neq i, F_{ik}^{(t)} = 0} \frac{R_{ik}^{(t)}}{R_{ik}^{(t)} + 1} H_{ik}^{(t)} + \sum_{k \neq i, F_{ik}^{(t)} = 1} H_{ik}^{(t)} \\
17: \quad \text{Send to } j \text{ the message } (F_{ji}^{(t+1)}, R_{ji}^{(t+1)}, H_{ji}^{(t+1)}) \\
18: \quad \textbf{end for} \\
19: H^{(t+1)}(i) = 1 + \sum_{k : F_{ik}^{(t)} = 0} \frac{R_{ik}^{(t)}}{R_{ik}^{(t)} + 1} H_{ik}^{(t)} + \sum_{k : F_{ik}^{(t)} = 1} H_{ik}^{(t)} \\
20: \textbf{end if} \\
21: \textbf{end for} \\
22: \textbf{end} \\

The approximation of the harmonic influence centrality of \(i\) will be stored in \(H^{(t)}(i)\).

We can think of a few natural stopping criteria. For example:

1. Fixed number of iterations \(t\). It is chosen for the algorithm a priori. Possible choices are time available, number of nodes, number of edges, diameter of the graph, etc.

2. When a node has received in two or more consecutive time step messages varying for less than a certain threshold.

3. When a node has sent for a certain number of consecutive times messages that change for a quantity below a certain threshold.

5.1.2 Interpretation via computation trees

To understand what happen in case of cycles we can use the so called \textit{analysis via computation trees} (for further details see [29]).

See in figure 5.1 (a) a graph on which we perform a generic message passing algorithm choosing as root node 1. We build the first 4 iterations of the \textit{computation tree} of message-passing updates in
5.1 – Generalization of the MPA

We track at every time step the nodes that have sent a message that reaches the root at that time step. At level $t$ of the tree (node at distance $t$ from the root) there are all nodes whose messages reach the root (node 1) after $t$ iterations of the message-passing algorithm.

The computation tree generated by a graph containing cycles has a number of nodes which diverges when $t$ goes to infinity. Note instead that if the original graph is a tree itself, the computation tree is just equal to the original graph. In our MPA if we choose a root $i$ we are therefore computing the harmonic influence centrality in the node $i$ in the computation tree and not on the original graph. Notice that changing number of iterations change also the computation tree. It is not obvious that the procedure still converges and it is clear that the result will be incorrect.

See an example of the computation tree for the first four iterations of the MPA in figure 5.2 on the next page. In (a) is presented the original graph. Node 0 is a type 0 stubborn node. We choose as root node 1. We present in (b), (c), (d), and (e) the first four time steps, showing what is the computation tree, that is the tree on which we are computing the quantities of interest, in particular the harmonic influence centrality. Blue nodes are stubborn nodes.

Notice that the initial condition $F^{(0)}_{ji} = 0 \; \forall i, j \in I$ is equivalent to require that each node in its 1st iteration behaves as if it has a stubborn neighbor. Consequently the computation tree includes a virtual stubborn node appended to the leaves. This is depicted in figure 5.2 on the following page (b)(c)(d) and (e) adding stubborn nodes without a label.
Despite the fact that we are not computing the centrality on the original graphs we will try to apply the algorithm to general graphs in section 5.2 on the next page and we compared the results to the correct one computed through the centralized algorithm (see [33]). Moreover we will see in chapter 6 on page 73 that under some conditions it is proved the convergence, even in presence of cycles.

5.1.3 Complexity and speed of convergence for trees

In this case we have an iterative version on the MPA analyzed in 4.2 on page 48. The algorithm for a single time step $t$ has to compute all messages in all nodes. Hence it is clear
that the complexity of a time step is equal to the complexity of the whole algorithm in its original version (section 4.2 on page 42).

The complexity hence depends on the number of iterations used in the algorithm. The number of iterations depends themselves from the stopping criterium chosen. Consider the maximum number of iteration after all the stopping criteria are reached as:

\[ t_{max} = \max_{i \in I} \{ t \text{ needed to reach the stopping criterium of the algorithm in node } i \} \]

Then we can immediately deduce from proposition 4.2 on page 48:

**Proposition 5.1.** The complexity of the MPA in section 4.2.1 on page 46 is described by the following items.

- The communication needed for each node \( i \) in terms of messages is \( O(t_{max}|N_i|) \).
- The communication needed across the whole network is \( O(t_{max}|I|) \).
- The number of operations for each node \( i \) is \( O(t_{max}|N_i|^2) \).
- The number of operations for the whole network is \( O(t_{max} \sum_{i \in I \setminus S_0} |N_i|^2) \).

For trees the convergence comes out after a number of time steps equal to the maximal radius of subtrees. In fact in a tree the messages are always overwritten. When the messages from the leaves reach the further nodes, then the messages will be always to equal themselves and in particular equal to the ones of the original algorithm in 4.2 on page 42.

In particular using the same notation used in section 4.2 on page 48 we define:

\[ T_{max} := \max_{h \in \{1, \ldots, n\}} \{ \text{diam}(T_h) \}, \]

and the algorithm converges for any trees in a time \( t \) not larger than \( T_{max} \).

### 5.2 Simulations

Although our current results only regards trees, we tried to apply the algorithm to some kind of random graphs such as Erdos-Renyi and small world graphs, obtaining very interesting results. First we have experienced that in all simulations the algorithm converges. Second in many cases the results found are very close to the optimal ones found through the algorithm proposed in [33].
We will see some examples in which the algorithm converges to the correct solution of problem \textsuperscript{2.14} on page 14 and others in which it does not.

We study also the differences with the correct result computed through the centralized algorithm in \cite{33} in function of the time step. In particular we define two different kinds of error. We call $H(i)$ the correct harmonic influence centrality of node $i$ and with $\hat{H}^{(t)}(i)$ the output of the algorithm in node $i$ after $t$ steps. We define the \textit{mean deviation error} at time step $t$ as:

$$
\text{error}_{\text{dev}}(t) = \frac{\sum_{i \in I} |H(i) - \hat{H}^{(t)}(i)|}{|I|}
$$

Since we are also interested in the optimal stubborn agent placement problem (see \textsuperscript{2.14} on page 14) we care also about the rank of the nodes, even more than the correct approximation of the harmonic influence centrality. We define as $\text{rank}_F(i)$ the position of $i$ in the sorted vector of nodes, according to the value of the function $F$ in $i$. We define therefore also a \textit{mean rank error} at time step $t$ as:

$$
\text{error}_{\text{rank}}(t) = \frac{\sum_{i \in I} |\text{rank}_H(i) - \text{rank}_{\hat{H}^{(t)}(i)}|}{|I|}
$$

It represent the mean error in the ranking of a node. Notice that when the value of the harmonic influence centrality is equal in two different nodes a wrong rank error could arise since the two nodes should have the same rank.

In the examples reported below we select as a stopping criterium for all the nodes that the mean difference between the output of two consecutive steps has to be below a certain threshold, chosen as $10^{-5}$. The stopping condition is therefore:

$$
\sum_{i \in I} |\hat{H}^{(t)}(i) - \hat{H}^{(t-1)}(i)| < 10^{-5}
$$

We will see only examples with undirected edges, i.e. symmetric graphs, since the algorithm is clearly build to consider symmetric communication channels. Notice that the algorithm could also be applied to directed graphs, but obtaining poor results.

5.2.1 A complete graph

We first consider a complete graph with 15 nodes. Nodes $\{1,2,3\} \in S^0$ are type 0 stubborn nodes (see figure \textsuperscript{5.3} on the next page).

We plot in figure \textsuperscript{5.4} on page 62 the comparison of the harmonic influence centrality in the different
agents $i$ between the centralized algorithm and the distributed algorithm when it has reached the stopping criterion. This happens after 47 time steps. So we are comparing $H(i)$ and $\hat{H}(47)(i)$, for all $i \in I$. As expected, since in a complete graph every node is connected to all the others, all the nodes have equal centrality. The MPA overestimates the values but maintains the equality between the nodes.

We can see the mean deviation error for each time step in figure 5.5 on the next page. Notice that after 6 time steps the error is exactly 0.

In this example we do not consider the rank error since all nodes have the same harmonic influence centrality.

### 5.2.2 A regular circular graph

A $d$-regular circular graph (with $d$ even) is a graph $G = (I, E)$ with $I = \{1, \ldots, n\}$ and such that:

$$E = \{(i, j) \in I \times I : (i - j) \mod n = \{1, \ldots, \frac{d}{2}\}\}$$

Consider the 6-regular circular graph with 15 nodes. Then each one is linked with the three previous and the three following it. See figure 5.6 on page 63. Nodes stubborn are three, labelled 1, 2 and 3.
5 – Message Passing Algorithm applied to general graphs

Figure 5.4. The comparison of the results for the harmonic influence centrality in the nodes

Figure 5.5. The mean deviation error of the distributed algorithm in function of the time step
5.2 – Simulations

Figure 5.6. A 6-regular circular graph; node 1, 2, 3 are type 0 stubborn.

In figures 5.7 on the next page we see the comparison between the results obtained with the MPA (after 55 time steps) and the correct results. Notice that the rank is respected and therefore in both the algorithm node 9 and 10 results as the ones that maximize the harmonic influence centrality and solve the optimal stubborn agent placement problem.

See in 5.8 on the following page the mean deviation error function of the time step (stopping criterium reached after 55 steps). We can see as in the previous example that the error decrease a lot in the first iterations, but after it increases and the algorithm converges.

5.2.3 Erdos-Renyi random graphs

In the Erdos-Renyi model $G(n, p)$ a graph with $n$ nodes is constructed by connecting nodes randomly. Each edge is included in the graph with probability $p$ independent from every other edges (for further details see [1]).

We consider an Erdos-Renyi random graph $G = (I, E)$ with 15 nodes (see figure 5.9 on page 65), three of them stubborn of type 1 (labeled 1, 2 and 3). The probability of putting an edge is $p = 0.2$.

In figures 5.10 on page 65 we compare the harmonic influence centralities correct $H(i)$ and the one computed through the MPA. We can see that our estimate approximates very well the function $H$ (computed exactly by the algorithm in [33]). In particular the estimate permits to correctly locate
5 – Message Passing Algorithm applied to general graphs

Figure 5.7. The comparison of the results for the harmonic influence centrality in the nodes.

Figure 5.8. The mean deviation error of the distributed algorithm in function of the time step.

the maximum of the HIC.
5.2 – Simulations

Figure 5.9. An Erdos-Renyi graph $G = (I, E)$; nodes 1, 2, 3 are type 0 stubborn.

Figure 5.10. The comparison of the results for the harmonic influence centrality in the nodes.

In figure 5.11 on the next page we present the mean deviation error and the mean ranking error function of the time steps. Notice that the ranking error goes to zero after just 4 iterations. The stopping condition is reached after 19 iterations.
Figure 5.11. The mean deviation error and mean ranking error of the distributed algorithm in function of the time step.

We now want to present simulations on larger Erdos-Renyi random graphs. Consider a $G(n, p)$ Erdos Renyi model with $n = 500$ nodes in the following two cases:

- (A) $p = \ln(n)/n \approx 0.012$
- (B) $p = 0.1$

We know from the theory (see [1]) that the probability in case (A) is the lowest that asymptotically guarantees the connectivity of the graph. Instead in case (B) the graph has much more cycles, the diameter is smaller and the graph is more clustered.

For ease of visualization, we do not plot the realization of the graph, nor the harmonic influence centrality which is computed. We instead just plot and discuss the time evolution of the error and analyze the solution of the optimal stubborn agent placement. We have the error results for (A) in 5.12 on the facing page and the error results for (B) in 5.13 on the next page.

The stopping criterion is satisfied at a similar time step (361 in (A) and 365 in (B)). In both the case the node with the higher harmonic influence centrality has been correctly recognized by the MPA. The very good performance can be also seen analyzing the mean ranking error that in both cases is below 3. This means that in average a node has less than 3 wrong ranking positions over the whole network, composed by 500 nodes.
5.2 – Simulations

Figure 5.12. The mean deviation error and mean ranking error of the MPA in function of the time step.

Figure 5.13. The mean deviation error and mean ranking error of the MPA in function of the time step.
5.2.4 Watts-Strogatts random graph

The Watts-Strogatz model is a random graph generation model that produces graphs with small-world properties, including short average path lengths and high clustering. The parameters are the number of nodes in the graph, the number of nearest-neighbours to connect and the probability $p$ of adding a shortcup edge (for further details see [32] and [1]).

Consider a Watts-Strogatz graph formed by a cycle (2-nearest neighbors) of 40 nodes with probability $p = 0.4$ to create a shortcut in the graph. Nodes stubborn are 5, labeled 1-5. See figure 5.14.

In figure 5.15 on the next page we see the comparison of the harmonic influence centrality correct and approximated through the MPA. In figure 5.16 on the facing page is presented the errors function of time steps. In this case we notice that the maximum is not well located in the distributed algorithm, even if the general shape of the curve is preserved.
5.2 – Simulations

Figure 5.15. The comparison of the results for the harmonic influence centrality in the nodes

Figure 5.16. The mean deviation error and mean ranking error of the distributed algorithm in function of the time step
5.2.5 A regular grid

Consider a geometric graph consisting in a 2D 4-regular grid. It has $13 \cdot 13 = 169$ nodes and the exact central node in position $(7,7)$ is a type 0 stubborn one.

In figures 5.17 and 5.18 on the facing page we see a field representation of the harmonic influence centrality computed on the grid, respectively for the centralized algorithm and for the MPA. This example show that the two functions really differer in shape. Moreover the maximum, that should be the neighbors of the stubborn node (figure 5.17), is wrongly located (figure 5.18 on the facing page).

![Figure 5.17. Field representation of the centralized algorithm on the 2D 4-regular grid.](image)
Figure 5.18. Field representation of the MPA on the 2D 4-regular grid.
Chapter 6

Convergence analysis of the MPA for d-regular graphs

We have already shown that the convergence of the message-passing algorithm presented is guaranteed for graphs that are trees (see 4.2 on page 42 and 5.1 on page 53).

In this chapter we want to study with more details the structure of the algorithm and our purpose is to say something more general on the convergence of the algorithm, taking into account the simulations presented in the chapter 5 on page 53.

In particular we want to prove the convergence of the approximation of the harmonic influence centrality, for some kinds of graphs that contain cycles inside.

When cycles are present in the graph, the iterative algorithm in section 5.1.1 on page 55 does not converge in finite time since there are feedbacks. An iterative method is called convergent if the corresponding sequence converges for given initial approximations. We try to provide a mathematically rigorous convergence analysis for some particular problems. The main result of the chapter is theorem 6.1 on page 75.

6.1 The dynamical systems and assumptions on the problem

Consider a graph $G := (I, E)$ with a set of stubborn nodes $S = S^0 \subseteq I$ in which nodes have opinion fixed equal to 0. We recall the assumptions that we already used (assumption 2.1 on page 8) throughout the analysis of the problem in the case of binary homogeneous gossip model.

**Assumption 6.1.** For each non stubborn node $i \in I \setminus S$ there exists a path from node $i$ to a stubborn node $s \in S$. 73
Assumption 6.2. The graph $G|_{I\setminus S_0}$ is symmetric, i.e. does not have directed edges.

Moreover we do not consider stubborn nodes to have out-neighbors. This is not a restriction as observed in remark 3.1 on page 19, it is just used to simplify the notation and the equations (6.2), (6.3) and (6.4) presented below. In fact any stubborn node $s \in S$ cannot be influenced and therefore an edge $(s, l) \in E$ would not have any effect on the evolution.

Consider now the additive assumption on the network:

Assumption 6.3. In the graph $G$ do not exist non-stubborn nodes $i \in I \setminus S$ with out-degree equal to one, i.e. $\{ i \in I \setminus S, |N^\text{out}_i| = 1 \} = \emptyset$.

If the graph is symmetric assumptions 6.1 on the preceding page and 6.3 are the same of saying that if we remove an arbitrary chosen edge (symmetric) the connected components can be trees only if they include a stubborn agent. For example see the picture 6.1 in which are presented a graph with a stubborn $s \in S$ placed in two different position: case A) satisfies the assumption 6.3 while case B) does not satisfy it.

Assumption 6.3 is automatically satisfied if we consider the following assumption:

Assumption 6.4. Consider a symmetric graph $G = (I, E)$ in which all the vertices non-stubborn have the same number of out-neighbors. Hence $|N^\text{out}_j| = d \quad \forall j \notin S$. The graph is called d-regular.

Notice that we are not interested in the out-neighbors of the stubborn nodes since they do not play any role in the algorithm.
First of all we recall which are the variables, which are their initial conditions and which are the equations that governs the evolution.

The variables are the vector of flags $F$, the vector of equivalent resistances $R$ and the vector of the partial harmonic centrality $H$, all indexed with the edges of the graph, i.e. they are in $\mathbb{R}^E$. Since they are computed iteratively they will also be indexed with the number of the iteration $t \in \mathbb{N}$ in the exponent. The initial conditions for $t = 0$ are:

$$
\begin{align*}
F_{ij}^{(0)} &= 0 \forall (i,j) \in E, \\
R_{ij}^{(0)} &= 0 \forall (i,j) \in E, \\
T_{ij}^{(0)} &= 0 \forall (i,j) \in E.
\end{align*}
$$

While the dynamic evolution of those variables $\forall (i,j) \in E, \forall t \in \mathbb{N}$ is given by (as already seen in section 5.1 on page 53):

$$
F_{ij}^{(t+1)} = \begin{cases} 
1 \text{ if } j \notin S, \{k \in N^\text{out}_j \setminus \{i\} | F_{jk}^{(t)} = 0\} = \emptyset \\
0 \text{ otherwise}
\end{cases}
$$

$$
R_{ij}^{(t+1)} = \begin{cases} 
\left( \sum_{k \neq i, F_{jk}^{(t)} = 0} \frac{1}{R_{jk}^{(t+1)}} \right)^{-1} \text{ if } j \notin S, \{k \in N^\text{out}_j \setminus \{i\} | F_{jk}^{(t)} = 0\} \neq \emptyset \\
0 \text{ otherwise}
\end{cases}
$$

$$
H_{ij}^{(t+1)} = \begin{cases} 
1 + \sum_{k \neq i, F_{jk}^{(t)} = 0} \frac{R_{jk}^{(t)}}{R_{jk}^{(t+1)}} H_{jk}^{(t)} + \sum_{k \neq i, H_{jk}^{(t)} = 1} H_{jk}^{(t)} \text{ if } j \notin S \\
0 \text{ otherwise}
\end{cases}
$$

Notice that while $F$ does not depend on the other variables, the resistance $R$ depends on the flags $R(F)$, and the partial centrality $H$ depends on both the flags and the resistances $H(F, R)$.

In this chapter we will prove the following theorem:

**Theorem 6.1.** The MPA algorithm described through the systems of equations (6.2), (6.3) and (6.4) with initial conditions (6.1) on a graph with assumptions 6.1 on page 73 and 6.4 on the facing page converges.

To prove the theorem we have to prove that the systems (6.2), (6.3) and (6.4) converges.
6.2 Convergence of the flags and resistances

**Proposition 6.2.** The flags $F$ given in equation (6.2) with assumptions 6.3 and initial conditions (6.1) are given by:

$$F^{(t)}_{ij} = 0 \quad \forall t \geq 0, \forall (i,j) \in E$$

**Proof.** The evolution of the flags $F$ given in equation (6.2) is independent from the evolution of the other variables. In the evolution either the $F^{(t+1)}_{ij}$ becomes 1 or stays 0. With the help of assumption 6.3 we ensure that the state having all $F_{ij} = 0$ for all $(i,j) \in E$ will remain the same forever. In fact the only case where some $F_{ij}$ could change from the initial condition to 1 is when $j$ is a node with out-grade 1, and it is not a stubborn. With assumption 6.3 on page 74 we have removed a-priori this possibility.

The computation of the resistances $R$ in (6.3) evolves depending on the values of the flags $F$. This will no more happen since all the $F^{(t)}_{ij}$ will be always constant equal to 0 as just proved. The resistances evolves according to the equation (6.3) stating from initial conditions 0. We observe that the resistances are used in our algorithm in (6.4) only through the following quantity:

$$T^{(t)}_{ij} := \frac{R^{(t)}_{ij}}{R^{(t)}_{ij} + 1}$$

(6.5)

Since $R^{(t)}_{ij}$ is a nonnegative number for all $(i,j) \in E$ and for all $t \geq 0$ then $0 \leq T^{(t)}_{ij} < 1$ and we obtain from the previous equation:

$$R^{(t)}_{ij} = \frac{T^{(t)}_{ij}}{1 - T^{(t)}_{ij}}$$

(6.6)

Note that $T^{(t)}_{ij} = 1$ would imply $R^{(t)}_{ij} = +\infty$.

It is clear that $T^{(t)}_{ij}$ converges to a nonnegative number strictly less than one if and only if $R^{(t)}_{ij}$ converges to a nonnegative finite number. We will study the evolution of the variable $T^{(t)}_{ij}$.

Thanks to (6.6) using some algebra we can rewrite the evolution of resistances (6.3) with the new variables $T$ defined in (6.5) as:

$$T^{(t+1)}_{ij} = \left\{ \begin{array}{ll}
\frac{(|N^{\text{out}}_j| - \sum_{k \neq i} T^{(t)}_{jk})^{-1}}{1 - T^{(t)}_{ij}} & \text{if } j \notin S \\
0 & \text{otherwise.}
\end{array} \right.$$
with the initial conditions \( R_{ij} = 0 \) \( \forall (ij) \in E \) that is equivalent to the conditions \( T_{ij} = 0 \) \( \forall (ij) \in E \).

Notice that in (6.7) we used also the knowledge about the \( F_{ij}^{(t)} \) to be sure that if \( j \notin S \), then \( \{k \in N_j \setminus \{i\} | F_{jk}^{(t)} = 0\} \neq \emptyset \).

Observe that (6.7) is a not-linear system and we are looking for a steady state solution, if it exists, starting with particular initial conditions.

We will show that \( T \) follows an evolution and its entries are not decreasing and limited, implying the fact that the sequence is convergent.

**Proposition 6.3.** The iterative system given by (6.7) starting from initial condition \( T_{ij}^{(0)} = 0 \) \( \forall (i,j) \in E \) is convergent for any graph satisfying assumptions 6.1 on page 73, 6.2 on page 73 and 6.3 on page 74. Therefore the convergence status denoted by \( T_{ij} \) is a steady state solution of (6.7).

Moreover

\[
0 \leq T_{ij} < 1 \quad \forall (i,j) \in E
\]

**Proof.** We already know that the sequence \( T_{ij}^{(t)} \) is limited, \( 0 \leq T_{ij}^{(t)} < 1 \) for all \( (i,j) \in E \) and for all \( t \geq 0 \).

We show that the sequence of \( T_{ij}^{(t+1)} \) is not decreasing \( \forall (i,j) \in E \). We prove it by induction. The first step is:

\[
T_{ij}^{(1)} = \begin{cases} 
\frac{1}{|N_{out}|} & \text{if } j \notin S \\
0 & \text{otherwise.}
\end{cases}
\]

And therefore \( T_{ij}^{(1)} \geq T_{ij}^{(0)} \) thanks to the initial conditions \( T_{ij}^{(0)} = 0, \forall (i,j) \in E \).

Now we proceed with the inductive step: suppose that the statement is true for a certain \( t \) and prove it for \( t + 1 \). Then it holds:

\[
T_{ij}^{(t)} \geq T_{ij}^{(t-1)} \quad \forall (i,j) \in E
\]

From this inequality directly holds:

\[
\frac{1}{(|N_{out}| - \sum_{k \neq i} T_{jk}^{(t)})} \geq \frac{1}{(|N_{out}| - \sum_{k \neq i} T_{jk}^{(t-1)})}
\]

That is nothing but the following inequality.

\[
T_{ij}^{(t+1)} \geq T_{ij}^{(t)}
\]
We found that the sequence \( T(t) \) is a non-decreasing sequence that has entries bounded and hence converges to \( T \).

We also want to show that even when \( t \to \infty \), the convergence status \( T_{ij}^{(t)} \to T_{ij} \) cannot assume value 1. We prove it by contradiction. Assume that exists an edge \((i, j) \in E\) such that \( T_{ij} = 1 \).

From the iterative process this implies that:

\[
1 = (|N^\text{out}_j| - \sum_{k \neq i} T_{jk})^{-1}
\]

and therefore using the symmetry assumption

\[
\sum_{k \neq i} T_{jk} = |N^\text{out}_j| - 1
\]

that yields to \( T_{jk} = 1 \forall (j, k) \in E, k \neq i \).

This implies that for all the edges \((m, n) \in E\) such that there exists a path from \((i, j)\) to \((m, n)\) then \( T_{mn} = 1 \). In particular from the assumptions 6.1 on page 73 we will have a \((k, s) \in E, s \in S\) s.t. \( T_{(k, s)} = 1 \). But in 6.7 we have \( T_{(ks)} = 0 \) fixed since \( s \in S \). This is a contradiction that derives from the fact that we have suppose that exists an edge \((i, j) \in E\) such that \( T_{ij}^{(t+1)} = 1 \).

Hence also the steady state vector \( T \) have entries strictly less than 1.

We have proved that the dynamic of the \( T^{(t)} \) converges to \( T \), which entries are nonnegative and strictly less than 1.

### 6.3 Convergence of the harmonic influence centralities

Remains to study the evolution of the function \( H \) expressed in iterative way in (6.4). With the new notation using the \( T \) and the proposition 6.2 on page 76 the system (6.4) can be rewritten as:

\[
H_{ij}^{(t+1)} = \begin{cases} 
1 + \sum_{k \neq i} T_{jk}^{(t)} H_{jk}^{(t)} & \text{if } j \notin S \\
0 & \text{otherwise}
\end{cases}
\]  

(6.8)

Therefore \( H_{ij}^{(t+1)} \) is a linear combination of \( H^{(t)} \) adapted to the graph \( G \).
We observe that we can rewrite the equation (6.8) in matricial form using a new matrix \( \hat{T}(t) \in \mathbb{R}^{E \times E} \) defined as follows:

\[
\hat{T}_{(ij)(hk)}^{(t)} := \begin{cases} 
T_{hk}^{(t)} & \text{if } h = j, k \neq i \\
0 & \text{otherwise.}
\end{cases}
\]

We have directly from the previous proposition that \( \hat{T}(t) \) also converges to a \( \hat{T} \), whose entries are nonnegative and smaller than 1.

The evolution of the vector \( H^{(t)} \in \mathbb{R}^{E} \) expressed in equation (6.8) can be rewritten in the following way:

\[
H^{(t+1)} = \hat{T}(t)H^{(t)} + c
\]

where \( c \) is a constant vector in \( \{0,1\}^{E} \) defined as follows:

\[
c_{ij} := \begin{cases} 
0 & \text{if } j \in S \\
1 & \text{otherwise.}
\end{cases}
\]

If we imagine to replace the \( \hat{T}(t) \) with \( \hat{T} \) we are simply trying to solve the linear system \( AP = C \) in an iterative way where \( A = I - \hat{T} \) is the splitting chosen for \( A \). To prove that this splitting converges we have to show that the matrix \( \hat{T} \) has spectral radius strictly less than 1.

**Proposition 6.4.** The matrix \( \hat{T}(t) \) with assumptions 6.1 on page 73, 6.2 on page 73 and 6.4 on page 74 has spectral radius strictly less than 1 for every \( t \):

\[
\rho(\hat{T}(t)) < 1 \quad \forall t \in \mathbb{N}
\]

Moreover also \( \rho(\hat{T}) < 1 \).

**Proof.** A sufficient condition for a matrix \( A \) to have spectral radius less than 1 is that all the rows has sum of the absolute values of their entry strictly less than 1:

\[
\max_i \sum_j |A_{ij}| < 1 \rightarrow \rho(A) < 1
\]

First we prove by induction that in this case holds the following property:
We have that the initial condition $\hat{T}^{(0)}_{(ij)(jh)} = 0 \forall (i,j), (jh) \in E$ satisfy this property. Then for $t = 1$ we have from (6.7):

$$\hat{T}^{(1)}_{(ij)(jh)} \leq \frac{1}{d} < \frac{1}{d-1}$$

Suppose now that for a certain $t \in \mathbb{N}$ holds $\hat{T}^{(t)}_{(ij)(jh)} < \frac{1}{d-1}$. Then

$$\hat{T}^{(t+1)}_{(ij)(jh)} \leq \frac{1}{d - \sum_{hk} \hat{T}^{(t)}_{(jh)(hk)}} < \frac{1}{d-1}$$

And we have proved the inductive step. Then we have proved that $\hat{T}^{(t)}_{(ij)(jh)} < \frac{1}{d-1} \forall t, \forall (i,j), (j,h) \in E$.

We show now that when $t \to +\infty$ the value $\frac{1}{d-1}$ cannot be assumed by the entries of $\hat{T}$. Suppose by contradiction that there exists an $(i,j) \in E$ such that $\hat{T}^{(t)}_{(ij)(jh)} = \frac{1}{d-1}$. Then this implies that $\sum_{(mn)} \hat{T}^{(t-1)}_{(jh)(mn)} = 1$ that implies that at time $t$ $\hat{T}^{(t)}_{(ij)(kh)} = \frac{1}{d-1} \forall (k,h) \in E$. As in the proof of proposition 6.3 on page 77 we use assumption 6.1 on page 73 to state that repeating the previous argument we will end up with:

$$\hat{T}^{(t)}_{(op)(qs)} = \frac{1}{d-1}$$

where $s \in S$. But this yield to a contradiction since $T^{(t)}_{(ij)(js)} = 0$ if $s \in S$.

Now we are able to prove directly (6.10) since we will have:

$$\sum_{(mh)} \hat{T}^{(t)}_{(ij)(mh)} \leq \sum_{(mh)} \frac{1}{(d-1)} = \frac{d-1}{d-1} = 1 \forall (i,j), (m,h) \in E, \forall t \in \mathbb{N}.$$ 

And this is true also for the limit $\hat{T}^{(t)}_{(ij)(mh)} \forall (i,j), (m,h) \in E$. \hfill \qed

**Proposition 6.5.** The iterative system (6.9) that computes $H^{(t)}$ is converging to the steady state solution of:

$$H = \hat{T}H + c.$$ 

**Proof.** Through proposition 6.4 on the preceding page we just proved that the sequence of the matrix
\( \hat{T}^{(t)} \) all has spectral radius strictly less than one, but since this matrix is changing in time this is not a sufficient condition to prove convergence of the algorithm (6.9).

We can rewrite the equation (6.9) in this way:

\[
H^{(t+1)} = \hat{T}H^{(t)} + (\hat{T}^{(t)} - \hat{T})H^{(t)} + c
\]

Since we already show that \( \rho(\hat{T}) < 1 \) we should prove that \( (\hat{T}^{(t)} - \hat{T})H^{(t)} \to 0 \) to be sure that evolution (6.9) converges.

Then we have to show that there exists \( \pi < +\infty \) s.t. \( \|H^{(t)}\| < \pi \forall t \), that means that the norm of \( H^{(t)} \) is bounded for all \( t \).

We remember that the infinite norm for the vector \( P \) is

\[
\|H\|_\infty = \max_{ij} |H_{ij}|
\]

while for the matrix \( \hat{T}^{(t)} \) is computed as follow \( \forall t \):

\[
\|\hat{T}^{(t)}\|_\infty = \max_{ij} \sum_{(kh)} |\hat{T}^{(t)}_{(ij)(kh)}| \leq \alpha < 1
\]

already bounded thanks to the previous proposition 6.4 on page 79.

For the vector \( c \), \( \|c\|_\infty = 1 \) due to its structure.

Then:

\[
\|H^{(t+1)}\|_\infty \leq \alpha \|H^{(t)}\|_\infty + 1
\]

and therefore:

\[
\|H^{(t+1)}\|_\infty \leq \sum_{k=1}^{\infty} \alpha^j + 1 = \frac{1}{1 - \alpha} = \pi
\]

and therefore we have proved the sentence.

Therefore the dynamic converges to the real solution of the real system. Putting together the previous results of the chapter we are able to prove the theorem 6.1 on page 75.

Notice that the output of the algorithm in section 5.1.1 on page 55 is the harmonic centrality on
the whole graph $G$. This quantity $H(l), l \notin S$ is computed as a linear combination of the vector of harmonic centrality $(H_{ij}, (i, j) \in E)$:

$$H(l) := 1 + \sum_{k,F(lk)=0} T(lk)H(lk) + \sum_{k,F(lk)=1} H(lk) \text{ for all } l \notin S$$  \hspace{1cm} (6.11)

Therefore since $H$ and $T$ are convergent with all entries bounded as proved in theorem 6.1 on page 75, also the output of the algorithm (6.11) will be convergent and bounded.
Chapter 7

Other centrality measures for graph models

7.1 Some relevant graph centralities

In many cases, also in applications, we are interested in which agent in a graph is the most influential, or central. In order to solve this problem one should define mathematically what centrality means. In general a centrality is a map from the set of nodes $I$ to a nonnegative real number. Clearly which definition of centrality is the most suitable depends on the context, and there is not only one measure that is right. Phillip Bonacich discusses some of these issues in paper [5].

Within the scope of graph theory and network analysis, there are various types of measures of the centrality of a vertex within a graph that determine the relative importance of that vertex. To provide some applicative examples, this kind of measure can represent how influential a person is within a social network, or how important a room is within a building or how much well-used a road is within an urban network.

We present in this section four examples of centrality measures that are widely used in network analysis: degree centrality, betweenness, closeness, and eigenvector centrality. For a more complete review as well as generalizations to weighted networks, see the works in [22] and [23].

7.1.1 Degree centrality

Historically first and conceptually simplest is the degree centrality. We refer to the notation introduced in section [1.1]. For a symmetric graph the degree centrality $\deg(i)$ of a node $i \in I$ is defined as:
That is equal to the number of neighbors that a node has. The degree can be interpreted in terms of the immediate risk of a node for catching whatever is flowing through the network.

In the case of a directed network, we usually define two separate measures of degree centrality, namely in-degree $\text{deg}^{(\text{in})}(i)$ and out-degree $\text{deg}^{(\text{out})}(i)$:

$$
\text{deg}^{(\text{in})}(i) := |N_i^{(\text{in})}|
$$
$$
\text{deg}^{(\text{out})}(i) := |N_i^{(\text{out})}|
$$

Accordingly, in-degree counts the number of ties directed to the node and out-degree counts the number of ties from the node directs to the others. When ties are associated to some positive aspects such as friendship or collaboration, in-degree is often interpreted as a form of popularity, and out-degree as gregariousness.

The centrality introduced here in (7.1) corresponds to the simple heuristic rule that who has more neighbors is more central and important in the network. This naive remark is sometimes heard about internet social networks as Twitter or Facebook. However in more complicated contexts this definition is absolutely not enough to describe the centrality of nodes. We would like to be able to weight in a certain manner all the ties, even the ones that are very far from the node of which we are computing the centrality.

### 7.1.2 Closeness centrality

As defined in (A.1) in graphs there is a natural distance metric between nodes, defined by the length of their shortest paths between them.

The farness of a node $i$ is defined as the sum of its distances to all other nodes, and its closeness is defined as the inverse of the farness:

$$
closeness(i) := \left( \sum_{j \neq i} \text{dst}(j, i) \right)^{-1}
$$

Thus, the more central a node is the lower is its total distance to all other nodes. Closeness can be regarded as a measure of how fast it will take to spread information from vertex $i$ to all other nodes sequentially. Note that by our definition of graph distances, the classic closeness centrality of all nodes in a disconnected graph would be 0. For more details refer to [28].
7.1 – Some relevant graph centralities

7.1.3 Betweenness centrality

Betweenness centrality was introduced by Linton Freeman (see [13]) as a measure for quantifying the influence of an individual on the communication between other individuals in a social network. In his definition, the betweenness centrality of a vertex is proportional to its probability to belong to a randomly chosen shortest path between two randomly chosen nodes.

More precisely the betweenness of a vertex \( i \) in a graph \( G = (I, E) \) with \( |I| = N \) vertices is computed as follows:

1. For each pair of vertices \((j, k)\), compute the shortest paths between them.
2. For each pair of vertices \((j, k)\), determine the fraction of shortest paths that pass through the vertex \( i \).
3. Sum this fraction over all pairs of vertices \((j, k)\).

More compactly the betweenness centrality of a node \( i \) can be represented as:

\[
\text{betweenness}(i) := \sum_{j,k \in I, j \neq k \neq i} \frac{\sigma_{jk}(i)}{\sigma_{jk}}
\]

where \( \sigma_{jk} \) is the total number of shortest paths from node \( j \) to node \( k \) and \( \sigma_{jk}(i) \) is the number of those paths that pass through \( i \).

For example, in an undirected star graph, the center vertex which is contained in every possible shortest path would have a betweenness of \((N - 1)(N - 2)\) while the leaves which are contained in no shortest paths would have a betweenness equal to 0.

7.1.4 Eigenvector centrality

Eigenvector centrality is another measure of the influence of a node in a network. It assigns relative scores to all nodes in the network based on the concept that connections to high-scoring nodes contribute more to the score of the node in question than equal connections to low-scoring nodes. For example the well known Google’s Page Rank is a variant of the Eigenvector centrality measure (see [7]).

For a given graph \( G = (I, E) \) let \( A \) be its adjacency matrix. Let \( x_i \) the centrality score of vertex \( i \) and \( x \) the vector of the centralities. Then \( x \) is the eigenvector related to the biggest eigenvalue \( \lambda \) of \( A^T \). We denoted by \( A^T \) the matrix transpose of \( A \).
Then $x$ satisfies:

$$A^T x = \lambda x$$

With a small rearrangement this can be rewritten for every entries $x_i$ as:

$$x_i := \frac{1}{\lambda} \sum_{k \in N_i^{(m)}} x_k = \frac{1}{\lambda} \sum_{k \in I} A_{ki} x_k$$

(7.2)

where $N_i^{(m)}$ is the set of the in-neighbors of $i$ and $\lambda$ is a constant.

In general, there will be many different eigenvalues $\lambda$ for which an eigenvector solution exists. However, the additional requirement that all the entries in the eigenvector must be positive implies by the Perron-Frobenius theorem that only the greatest eigenvalue results in the desired centrality measure (see [21]). The $i^{th}$ component of the related eigenvector then gives the centrality score of the vertex $i$ in the network.

Furthermore, this can be generalized so that the entries in $A$ can be real numbers representing connection strengths, as in a stochastic matrix.

In the case of Page Rank the vertices are the web-pages and the edge $(i, j) \in E$ has the meaning that page $i$ has an hyperlink to page $j$. Therefore through equation (7.2) can be seen that if page $i$ increases its centrality also page $j$ increases its centrality.

### 7.2 The Bonacich network centrality measures

Philip Bonacich proposed in 1987 a family of centrality measures in the paper [3]. Those measures have been studied in many articles with different applications like in [4] and in [8].

Consider a graph $G = (I, E)$ with node set $I$ and edge set $E$. Let $|I| = N$ be the number of nodes. Note that the network is not required to be symmetric. Let $A$ be its adjacency matrix.

Consider now $A^k$ the $k^{th}$ power of $A$ in matricial sense. $A_{ij}^k \geq 0$ measures the number of paths of length exactly $k \geq 1$ in $G$ from node $i$ to node $j$. Note that $A^0 = I_N$ where $I_N$ is the identity matrix of dimension $N \times N$. For more details see [30].

Consider $\sum_{k=0}^{+\infty} (\beta A)^k$ where in $\beta \in \mathbb{R}$. It represents the sum of the number of all the paths between $i$ and $j$, where paths of length $k$ are weighted by $\beta^k$. Hence $\beta$ is a decay factor that scales down the weight of longer paths.
We have that:

\[ \sum_{k=0}^{+\infty} (\beta A)^k = [I_N - \beta A] \]  

(7.3)

This expression is well defined when the absolute value of \( \beta \) is smaller than the inverse of the spectral radius of \( A \). In formulas:

\[ |\beta| \leq \frac{1}{\rho(A)} \]  

(7.4)

Under such condition we have that:

**Definition 7.1.** [Bonacich Centrality] Consider \( G = (I, E) \) a graph and \( A \) its (weighted) adjacency matrix. Fix as parameters a scalar \( \beta \in \mathbb{R} \) such that (7.4) is satisfied and a weight vector \( v \in \mathbb{R}^I \). Then the vector of Bonacich centrality of the nodes is defined as:

\[ b(A, \beta, v) := [I_N - \beta A]^{-1}v \]  

(7.5)

The magnitude of \( \beta \) reflects the degree to which the communication is transmitted locally or to the structure as a whole. Small values of \( \beta \) heavily weights the local structure while large values take more in account the position of the nodes in the network. \( \beta \) can be thought of as a radius within which the agent wishes to asses centrality. \( v \) is a vector that weights differently a priori the nodes in the graph.

Rewriting (7.5) we get:

\[ b(A, \beta, v) = v + \beta Avb(A, \beta, v) \]  

(7.6)

and the \( i \)th entry of \( b(A, \beta, v) \) is the centrality of an agent \( i \):

\[ b_i(A, \beta, v) = v_i + \beta \sum_{j=1}^{N} A_{ij}b_j(A, \beta, v) \]  

(7.7)

If we take \( v := 1 \), where \( 1 \) is the vector of dimension \( N \) with all entries equal to 1, then the \( i \)th entry of \( b(A, \beta, v) \) becomes from (7.5):
Other centrality measures for graph models

\[ b_i(A, \beta, 1) = \sum_{j=1}^{N} \sum_{k=1}^{+\infty} \beta^k A^k_{ij} \]

and it counts the total number of paths in \( G \) that start from \( i \) scaled by their length.

We also want to remark the importance of the parameter \( \beta \). If we take \( v := A1 \), using equations (7.3) and (7.5) we have that:

\[ b(A, \beta, A1) = \sum_{k=0}^{+\infty} \beta^k A^{k+1}1 \]

If \( \beta = 0 \) from (7.5) we get \( b(A, \beta, v) = A1 \) and therefore only the direct links count. This means that we are simply counting the (weighted) degree of a node as measure of centrality. When \( \beta \) increases farther nodes are taken in account and give a larger contribution to the centrality \( b \).

When \( \beta \leq 0 \), even powers of \( A \) are weighted negatively and odd powers are weighted positively. Thus a node \( i \) having many direct connections contributes positively to centrality but if the neighbors of \( i \) have themselves many connections the centrality of \( i \) is reduced.

This fact can have also applicative examples. This is well explained in Caplow's and Gamson's works (see [9] and [14]).

We have seen that Bonacich centrality is actually a very wide family of centralities that can give different rankings depending on the values of its parameters \( \beta \) and \( v \). This measure is then able to catch a part of the ambiguity in the concept of centrality. There are in fact different centralities depending on the degrees to which local and global structures should be weighted and whether that weight should be positive or negative.

Note that with the definition of Bonacich centrality given in 7.1 on the previous page highlight which nodes are more effective in collecting information from the other nodes. This choice contrasts the idea of central node as the most influential. The latter analysis, however, can be simply recovered by looking at the reversed graph, defined as \( A^T \), the transpose of \( A \). The centrality \( b(A^T, \beta, v) \) measures indeed how influential is a node. Moreover note that we can use a weighted adjacency matrix \( A \), with \( A_{ij} \in R^+_0 \) such that \( A_{ij} > 0 \) if and only if the edge \((i,j) \in E \). In this case the definitions and interpretation given are still valid in a weighted sense.

7.2.1 Example

We present a very simple example. Consider the symmetric graph in figure 7.1 on the facing page.
Let $A$ be its adjacency matrix. We want to compute the Bonacich centrality $b(A, \beta, v)$ with $\beta := 0.25$ and $v := 1$.

Notice that $[I_7 - 0.25A]$ is strictly diagonally dominant matrix and therefore is non singular by the Levy-Desplanques theorem (see theorem 6.1.10 in [17]).

Using definition 7.1 we got:

$$b(A, 0.5, 1) := [I_7 - 0.25A]^{-1}1 = [2, 2, 2, 1.5, 1.5, 1.5]$$

and therefore node 1 is the most central.

### 7.3 Distributed computation of the Bonacich centrality

Rewriting the definition of the Bonacich centrality 7.1 in an implicit matricial form as in equation (7.6) we have:

$$b = v + \beta Ab$$

(7.8)

where for simplicity we denoted $b(A, \beta, v)$ just by $b$.

The centrality $b_i$ of a node $i$ can be found through the system of equations (7.8). The centrality of a node $i$ is computed through the centralities $b_j$ of the nodes $j$ that are neighbors of $i$. Because of this feature this computation is said to be adapted to the graph $G$. 

89
The other elements that take a role in the computation of \( b_i \) are the weights of the node itself \( v_i \), the weighted strength of the connections to its neighbors given by the \( i^{th} \) row of the adjacency matrix \( A \) and the value of the parameter \( \beta \).

We then suppose that each node \( i \) knows \( \beta, v_i \), the node that he influences and the strength of this influence given by the \( i^{th} \) row of the adjacency matrix \( A \).

We can notice from equation (7.8) that we have to solve a linear system in the unknowns \( b_i \), \( i = 1, \ldots, N \).

\[
[I - \beta A]b = v
\]  

(7.9)

from this we can simply derive the following iterative formula:

\[
b^{(t+1)} = v + \beta A b^{(t)} \quad t = 0,1,\ldots
\]  

(7.10)

where we denoted with \( b^{(t)} \) the \( t \)-th approximation of the centrality. The expression in equation (7.10) is the expression for solving a linear system in an iterative way. An iterative method is a mathematical procedure that generates a sequence of improving approximate solutions. Such a method is called convergent if the corresponding sequence converges for given initial approximations.

**Proposition 7.1.** The iterative method in (7.10) is convergent for every choice of \( b^{(0)} \) to the Bonacich centrality \( b(A, \beta, v) \).

**Proof.** \( \beta A \) in this case is called the interaction matrix. The iterative method for solving a linear system starting from any initial condition \( b^{(0)} \) converges if and only if the spectral radius of \( \beta A \) is strictly less than one. In formulas:

\[
\lim_{t \to +\infty} b^{(t)} = x, \forall b^{(0)} \iff \rho(\beta A) < 1
\]  

(7.11)

The proof of this theorem can be found in every book of numerical methods. In our case therefore we have to check that the matrix \( \beta A \) has spectral radius less than one to apply the iterative method developed in this paragraph. However the Bonacich centrality on page 87 is well defined only if \( |\beta| < \frac{1}{\rho(A)} \) that yield to the fact that \( \rho(\beta A) < 1 \).

We will see an applicative example in the next section.
7.4 An applicative example: Optimal Pricing in networks

7.4.1 Definition and solution of the problem

We apply the theory developed in section 7.3 to the solution of the problem expressed by Candogan et al. in [8]. We outline here the problem and the solution given by the authors.

Let \( G = (I, E) \) be the graph that represents a society. \( I \) are the individuals and \( E \) represents the set of connections between them. We can represent the social network by a weighted adjacency matrix \( A \). \( A_{ij} \) represents the strength of the influence of agent \( j \) on \( i \). We assume that \( A_{ij} \geq 0 \) \( \forall i, j \in I \) and \( A_{ii} = 0 \) \( \forall i \in I \).

A monopolist wants to make the highest profit selling a divisible good in the market. He is able to choose the price to apply to each customer. Then the monopolist in general can choose a vector \( p \in \mathbb{R}^I \) that is a function \( p : I \rightarrow \mathbb{R}^N \). \( p_i \) then is the price applied to agent \( i \) for the good.

Each agent \( i \) chooses the quantity of good to purchase, represented by the variable \( x_i \). This depends on the price of the good and on the quantities purchased by his neighbors. In particular we assume that he wants to maximize the following utility function:

\[
u_i(x) = a_i x_i - d_i x_i^2 + x_i \cdot \sum_{j \in I} (A_{ij} x_j) - p_i x_i \quad (7.12)\]

where \( x \) is the vector of the aggregate quantities.

The first two terms represent the utility of agent \( i \) irrespective to the consumption of the other agents and depends on the parameters \( a_i \) and \( d_i \). The third term represents instead the positive network effect given by the purchases of the neighbors. The last term is a negative effect due the price of the good.

Then we have a two-stage pricing-consumption game, where the players are the monopolist and the agents.

The monopolist chooses the pricing strategy \( p \) in order to maximize profits:

\[
p = \arg\max_q \sum_{i \in I} (q_i x_i - c x_i) \quad (7.13)\]

where \( c \) is the cost of production of a unitary quantity of good. Agent \( i \) chooses to purchase a quantity \( x_i \) such that:
\[ x_i = \arg \max_{y_i \in [0, +\infty)} u_i(y) \] (7.14)

We want to solve problem in (7.13) when \( x \) is a perfect Nash equilibrium of the consumption game (7.14) played by the nodes.

We add three assumptions in order to give a simpler solution of the problem. These assumptions can be relaxed: details can be found in [8]. We do not go into details since our purpose is to compute in a distributed way the solution of the problem.

**Assumption 7.1.** \( \forall i \in I, \ 2d_i > \sum_{j \in I} A_{ij} \).

This assumption ensures that in this game the optimal consumption level of each agent is bounded.

**Assumption 7.2.** \( \forall i \in I, \ a_i > c \).

This assumption guarantees that in absence of network effects the monopolist would find optimal to charge individual prices low enough so that all consumers purchase a positive amount of the good.

**Assumption 7.3.** \( a_i = a_0 \ \forall i \in I, \ d_i = d_0 \ \forall i \in I \).

This assumption implies homogeneity in the behavior of customers and it is only given in order to simplify the form of the solution expressed and proved in [8]:

**Theorem 7.2.** Under assumptions 7.1, 7.2, and 7.3 the vector of optimal prices \( p \) is given by:

\[
p = \frac{(a_0 + c)}{2} \mathbf{1} + \frac{(a_0 - c)}{8d_0} \cdot A \cdot b\left(\frac{A + A^T}{2}, \frac{1}{2d_0}, \mathbf{1}\right) - \frac{(a_0 - c)}{8d_0} \cdot A^T \cdot b\left(\frac{A + A^T}{2}, \frac{1}{2d_0}, \mathbf{1}\right)
\]

where \( b \) is the Bonacich Centrality given in definition 7.1 on page 87.

The graph \( \frac{A + A^T}{2} \) is the average interaction graph and represents the average interaction between pairs of agents in \( G \). Intuitively the Bonacich centrality \( b\left(\frac{A + A^T}{2}, \frac{1}{2d_0}, \mathbf{1}\right) \) measures how central each agent is with respect to the average interaction graph. The optimal prices given in Theorem 7.2 have three terms:

1. the first one, \( \frac{(a_0 + c)}{2} \mathbf{1} \), is a nominal price irrespective to the agent
2. the second one, \( \frac{(a_0 - c)}{8d_0} \cdot A \cdot b\left(\frac{A + A^T}{2}, \frac{1}{2d_0}, \mathbf{1}\right) \), is a markup due to the utility that this agent derives from his neighbors

92
3. the third one, $-\frac{(a_0-c)}{8d_0} \cdot A^T \cdot b\left(\frac{A+A^T}{2}, \frac{1}{2d_0}, 1\right)$ is a discount offered to the agent since increasing his consumption increases the consumption of his peers.

Remark 7.1. Note that if the interaction matrix $A$ is symmetric then the vector $p$ would be simply $p = \frac{(a_0+c)}{2} \cdot 1$ irrespective to the agent $i$. Hence symmetric graphs are not interesting case studies for this problem.

Thinking about examples of real social network, this problem is therefore interesting for networks such as Twitter, but not in networks as Facebook, where if $a$ is friend of $b$, $b$ must be friend of $a$. Note however that if weighted edges are allowed, also in symmetric networks such as Facebook interesting cases arise.

7.4.2 Distributed computation for the optimal pricing

The monopolist has with theorem 7.2 on the facing page the possibility to compute the optimal prices under the given assumptions 7.1 on the preceding page, 7.2 on the facing page and 7.3 on the preceding page. To be able to perform the computation he should know:

- the cost for producing a unity of the good $c$;
- the behavior of the agents given by $a_0$ and $d_0$;
- the matrix of the interactions $A$;

The first item is obviously known by the monopolist, while the second and the third should be discovered somehow. Sometimes it is not possible to have the whole matrix graph and so the computation should be made locally and distributed.

We have seen in the section 7.3 on page 89 in proposition 7.1 on page 90 a way to compute locally and in a distributed way the Bonacich centrality. Therefore we assume the following:

Assumption 7.4. Each agent $i$ knows $a_0$, $d_0$, $c$ and the strength of the connections between all his out-neighbors $A_{ij}, \forall j \in I$ and in-neighbors $A_{ji}, \forall j \in I$. Moreover each agent is able to communicate and transfer information with all his neighbors (in-neighbors as well as out-neighbors).

Remembering this well-known result:

Lemma 7.3. [Gershgorin cycle theorem] Let $A$ be a complex $N \times N$ matrix, with entries $A_{ij}$. For $i \in 1, \ldots, N$ let $R_i = \sum_{j \neq i} |A_{ij}|$ be the sum of the absolute values of the non-diagonal entries in the $i$th row. Let $D(A_{ii}, R_i)$ be the closed disc centered at $A_{ii}$ with radius $R_i$. Such a disc is called a Gershgorin disc. Then every eigenvalue of $A$ lies within at least one of the Gershgorin discs $D(A_{ii}, R_i)$. 

93
Proof. The proof can be found in [15].

we can then prove the following:

Proposition 7.4. Under assumptions 7.1 on page 92, 7.2 on page 92, 7.3 on page 92 and 7.4 on the previous page the expression defined in 7.4.1 on page 91 is solvable in a distributed way.

Proof. Since we want to compute the Bonacich centrality in a distributed way through proposition 7.1 on page 90 we have to prove that in this particular context the problem satisfy the condition given in (7.11). In this case the condition becomes $\rho(\frac{A+A^T}{2}) < 2d_0$. A matrix $A$ and its transpose $A^T$ have the same eigenvalues. Also $\frac{A+A^T}{2}$ has the same eigenvalues of $A$. In fact if $\lambda$ is an eigenvalue of $A$, i.e. $\exists x \in \mathbb{R}^N$ s.t. $\lambda = \frac{x^T Ax}{x^T x}$, then:

$$\frac{x^T(\frac{A+A^T}{2})x}{x^T x} = \frac{1}{2} \frac{x^T Ax + (Ax)^T x}{x^T x} = \frac{1}{2} \lambda + \frac{1}{2} \lambda = \lambda$$

Remember that in our case $A_{ij} \geq 0 \forall i,j \in I$ and $A_{ii} = 0 \forall i \in I$. Then assumption 7.1 on page 92 and assumption 7.3 on page 92 guarantee that $\sum_j |A_{ij}| < 2d_0 \forall i \in I$. Using theorem 7.3 on the preceding page we have that all the eigenvalues $\lambda$ should be less than $2d_0$ that is actually what we need to prove the convergence of the iterative method in equation (7.10).

Each agent $i$ computes his own Bonacich centrality in an iterative way, that we proved converges. After this computation the optimal price $p_i$ can be computed in one step:

$$p_i = \frac{(a_0 + c)}{2} + \frac{(a_0 - c)}{8d_0} \cdot \sum_{j \in I} A_{ij} b_j(\frac{A + A^T}{2}, \frac{1}{2d_0}, \mathbf{1}) - \frac{(a_0 - c)}{8d_0} \cdot \sum_{j \in I} A_{ji} b_j(\frac{A + A^T}{2}, \frac{1}{2d_0}, \mathbf{1})$$

(7.15)

This shows that when the neighbors have found their own Bonacich centrality in a distributed way according to equation (7.10) they communicate it to their neighbors (in and out). Then applying the formula in (7.15) all the agents are able to compute their own price immediately in one step and they just have to communicate it to the monopolist when they want to buy the divisible good. 

We insert the algorithm in pseudo-code for computing the solution $p_i$ in every node:

1: begin
2: $b_i^{(0)} = 0 \forall i \in I$
3: for $t = 1$ to stop criteria do
4:   for each $i \in I$ do
5:     $b_i^{(t)} = 1 + \frac{1}{4d_0}(\sum_{j \in I} A_{ij} b_j^{(t-1)} + \sum_{j \in I} A_{ji} b_j^{(t-1)})$
6: \textbf{end for}
7: \textbf{end for}
8: \textbf{for each } i \in I \textbf{ do}
9: \quad p_i = \frac{(a_0+c)}{2} + \frac{(a_0-c)}{8d_0} \cdot \sum_{j\in I} A_{ij} b_j^{(t)} - \frac{(a_0-c)}{8d_0} \cdot \sum_{j\in I} A_{ji} b_j^{(t)}
10: \textbf{end for}
11: \textbf{end}

\textit{Remark 7.2.} From the applicative point of view each node will have to collaborate to the monopolist to compute the price. This task is hard to achieve since each agent in the graph has to collaborate with the monopolist, and this is what makes difficult to know the graph topology, i.e. the adjacency matrix \( A \).

7.4.3 Some numerical examples

\textbf{Example 7.4.1.} Consider a network with three agents and assume that the problem parameters are such that \( c = 1 \), \( a_1 = a_2 = a_3 = 2 \), \( d_1 = d_2 = d_3 = 6 \) and the weighted adjacency matrix \( A \) is:

\[
A = \begin{bmatrix}
0 & 10 & 0 \\
1 & 0 & 1 \\
0 & 10 & 0
\end{bmatrix}
\]

The matrix describes a line network showed in Figure 7.2. Agent one and three are influenced only by agent 2, while agent 2 is influenced both by 1 and 2 but with a smaller weight.

As can be seen assumption 7.1 on page 92, 7.2 on page 92 and 7.3 on page 92 are satisfied with \( d_0 = 6 \) and \( a_0 = 2 \). The Bonacich centrality \( b(A + A^T, \frac{1}{2d_0}, 1) \) can be computed either in a centralized or distributed way giving the following numerical results:

\[
b(A + A^T, \frac{1}{2d_0}, 1) \approx [2.52, 3.31, 2.52]
\]

Then we can compute the vector of optimal prices \( p \):
The price offered by the monopolist to agent 2 is lower than the cost of production $c = 1$, since it is in the interest of the monopolist to incentive this agent, who has a large influence over the peers, to purchase larger amounts of the goods.

**Example 7.4.2.** Consider the network showed in Figure 7.3 with seven agents and assume that the problem parameters are such that $c = 1, a_1 = a_2 = a_3 = 2, d_1 = d_2 = d_3 = 2$ and the adjacency matrix $A$ is not weighted ($A_{ij} \in \{0,1\}, \forall i,j \in I$).

![Figure 7.3. The graph of the example 7.4.2](image)

Assumption 7.1 on page 92, 7.2 on page 92 and 7.3 on page 92 are satisfied with $d_0 = 2$ and $a_0 = 2$. The Bonacich centrality results:

$$b\left(\frac{A + A^T}{2}, \frac{1}{2d_0}, 1\right) \approx [2.91; 2.55; 2.55; 2.55; 3.27; 3.27; 3.27]$$

Hence nodes 5, 6 and 7 are more central than node 1 even if they have the same degree, while node 1 is more central than 2, 3 and 4. Then we can compute the vector of optimal prices $p$:

$$p \approx [1.50; 1.50; 1.50; 1.50; 1.50; 1.50; 1.50]$$
Since in this case the network is symmetric the optimal prices are uniform between the agents, as we expected from the theory. We remark the fact that this equality arises even if the centrality of the nodes of the average interaction graph are different.

**Example 7.4.3.** We slightly modify the previous example removing some edges in such a way that the graph is no more symmetric. See figure 7.4.

![Graph](image)

Figure 7.4. The graph of the example 7.4.3

Assumption 7.1 on page 92, 7.2 on page 92 and 7.3 on page 92 are satisfied with $d_0 = 2$ and $a_0 = 2$ are satisfied. The Bonacich centrality results:

$$b\left(\frac{A + A^T}{2}, \frac{1}{2d_0}, 1\right) \approx [1.56; 1.49; 1.49; 1.49; 2.37; 2.37; 2.37]$$

The centralities have changed from the previous example but their rankinks are the same. The vector of optimal prices $p$ results:

$$p \approx [1.22, 1.45, 1.45, 1.45, 1.59, 1.59, 1.59]$$

This shows that the monopolist would offer more discount to the agent 1, even if it is not the one with higher centrality. In fact, even if it is not so central in the network, he cannot be influenced by the others and he influences directly nodes 2, 3 and 4 that have no other out-neighbors but him.
Appendix A

Graph Theory basics

We first present some basic definitions and properties about graph theory useful throughout the text. It is also introduced the notation typical of the subject. Some fundamental definitions and facts are repeated in the chapters.

**Definition A.1. [Graph]** A graph $G$ is a pair $G := (I, E)$ where:

1. $I$ is a finite set, whose elements are said to be the *nodes* of $G$
2. $E \subset I \times I$ is the set of *edges*

$E$ is the set of links or edges representing the connectivity among the nodes. We usually draw an edge $(i, j) \in E$ as an arrow directed from $i$ to $j$.

The out-neighbor set of a node $i \in I$ is defined as:

$$N_{i}^{(out)} = \{j \in I | (i, j) \in E\}$$

while the in-neighbor set of a node $i \in I$ is defined as:

$$N_{i}^{(in)} = \{j \in I | (j, i) \in E\}$$

A graph such that for every $(i, j) \in E$ exists $(j, i) \in E$ is called *symmetric*. If the graph is symmetric clearly $N_{i}^{(out)} = N_{i}^{(in)}$ and this set is simply called neighbor set of node $i \in I$ and is denoted by $N_{i}$:

$$N_{i} = \{j \in I | (i, j) \in E\}$$
The number of elements in $N_i^{(in)}$ is usually called the in-degree of $i$, while the number of $i$ elements in $N_i^{(out)}$ is usually called the out-degree of $i$. For symmetric graphs since they are equal we simply refer to them as the degree of $i$.

We provide an example of a graph in figure A.1. There we have that $N = 6$, $I = \{1,2,3,4,5,6\}$ and $E = \{(2,1), (4,2), (4,6), (5,2), (5,3), (5,4), (6,4)\}$. Moreover $N_1 = \{5\}$, $N_2 = \{1\}$, $N_3 = \emptyset$, $N_4 = \{2,6\}$, $N_5 = \{2,3,4\}$ and $N_6 = \{4\}$

![Figure A.1. An example of a directed graph with 6 nodes.](image)

A graph such that for every $(i, j) \in E$ exists $(j, i) \in E$ is called symmetric.

Properties of graph are often expressed in term of some connectivity inside the graph.

A path of length $l$ from node $i$ to $j$ in $G$ is an ordered list of nodes $(w_0, w_1, \ldots, w_k)$ such that $w_0 = i$, $w_l = j$ and $(w_i, w_{i+1}) \in E$ for every $i \in \{0, \ldots, l - 1\}$.

If a path from $i$ to $j$ exists, or $j = i$ then $j$ is said to be reachable from $i$. A graph $G = (I, E)$ is said to be strongly connected if for every two nodes $(i, j)$, $i$ is reachable from $j$ and viceversa. A graph $G = (I, E)$ is said to be connected if for any pair of nodes $(i, j)$ either $j$ is reachable from $i$ or $i$ is reachable from $j$.

A node $i \in I$ is a globally reachable node if for every other node $j$ there exists a path from $j$ to $i$.

From the notion of path we can derive a notion of distance between two nodes:

$$dst_G(i, j) := \min\{l : \text{exists a path in G of length } l \text{ from } i \text{ to } j\} \quad (A.1)$$
provided that \( i \neq j \) and \( \text{dst}_G(i, i) := 0 \). If \( j \) is not reachable from \( i \) then \( \text{dst}_G(i, j) = \infty \).

Then the diameter of a graph can be defined as:

\[
diam(G) := \max\{\text{dst}_G(i, j) : i, j \in I\}
\]

An important class of of path are cycles: a cycle is a path from a node to itself of length at least 3, with no vertex repeated except the first and the last one.

A graph is said to be acyclic if do not exists cycles in it.

We now present the definition of a tree.

**Definition A.2. [Tree]** A graph \( G = (I, E) \) that is symmetric, acyclic, and connected is called a tree.

An important property of trees is that for any pair of nodes \((i, j)\) there is exactly one path from \( i \) to \( j \) in \( G \).

Graphs are strictly related to matrices. In particular a graph can be represented by its adjacency matrix. Given a graph \( G = (I, E) \) the adjacency matrix \( A \) is a matrix in \( \{0, 1\}^{I \times I} \) such that:

\[
A_{ij} := \begin{cases} 
1 & \text{if } (i, j) \in E \\
0 & \text{otherwise}
\end{cases}
\]  

(A.2)

For example the adjacency matrix of the graph in figure A.1 on the preceding page is the following:

\[
A = \begin{bmatrix}
0 & 0 & 0 & 0 & 1 & 0 \\
1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 1 \\
0 & 1 & 1 & 1 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0
\end{bmatrix}
\]

Given instead a matrix \( P \) which entries are nonnegative we can build a graph \( G_P \) such that \((i, j) \in E\) if and only if \( P_{ij} > 0 \).

Another representation is the incidence matrix. The incidence matrix of a directed graph \( G \) is a matrix \( A \) in \( \{-1, 0, 1\}^{I \times E} \) such that \( A_{ie} = -1 \) if the edge \( e \) leaves vertex \( i \), \( 1 \) if it enters vertex \( i \) and 0 otherwise.
A.1 Opinion dynamics over graphs

We present here the definition of opinion dynamics over graphs, focusing in particular on the case of consensus. For details and proofs we refer to [16] and [31].

Consider a graph $G = (I, E)$. Every node $i \in I$ has a time-varying state $x_i(t) \in \mathbb{R}^q$. The state can represent an opinion, a speed, a measurement, etc.

Call $x(t)$ the vector gathering the states of all $i \in I$. Suppose for simplicity that $q = 1$. Then $x(t) \in \mathbb{R}^{|I|}$.

States evolves according to a law that can be deterministic or probabilistic.

In a deterministic discrete time case we have a law $f_i : \mathbb{R}^{|I|} \rightarrow \mathbb{R}$ that depends on $i$ and hence:

$$x_i(t+1) = f_i(x(t))$$

and considering the vector $x(t)$ we will have a dynamical system:

$$x(t+1) = F(x(t)) \quad (A.3)$$

where $F : \mathbb{R}^{|I|} \rightarrow \mathbb{R}^{|I|}$.

We say that the dynamical system (A.3) is adapted to $G$ if $f_i(x)$ actually only depends on those components $x_j$ of $x$ for which $(i, j) \in E$ and from itself.

**Definition A.3.** The dynamic of $x(t)$ is said to reach a consensus if exists $\alpha \in \mathbb{R}$ such that:

$$\lim_{t \rightarrow +\infty} x(t) = \alpha 1_I$$

where $1_I$ is the vector of all ones and $\alpha$ is called the consensus point.

Then consensus is reached if all the components will have asymptotically value $\alpha$.

Consider a state law that is adapted to the graph, depends linearly from the states and it is time invariant. Then (A.3) can be rewritten as:

$$x(t+1) = Px(t) \quad (A.4)$$

where $P \in \mathbb{R}^{I \times I}$ is a matrix. Usually $P$ is a stochastic matrix.
Appendix B

Markov Chains

Here are presented some definition and results about Markov chains (usually shortened in M.C.). More details can be found for example in [20].

Definition B.1. [Stochastic Process] A stochastic process is a sequence of random variables $X_t$ taking values in some state space $V$.

A Markov chain is a particular stochastic process defined as follows:

Definition B.2. [Markov chain] Any stochastic process $X_t$ on $V$ such that $P(X_{t+1} = v_{t+1}|X_0 = v_0, X_1 = v_1, \ldots, X_t = v_t) = P(X_{t+1} = v_{t+1}|X_t = v_t)$ $\forall t, \forall v_0, v_1, \ldots, v_{t+1} \in V$ is called a Markov chain.

It is a random process usually characterized as memoryless: the next state depends only on the current state and not on the sequence of events that preceded it. This is called the Markov property. In simple words in a Markov chain the future depends from the past only through the present state. Markov chains are often described by a directed graph, where the nodes are the states in $V$ and edges are labeled by the probabilities of going from one state to the other states.

Definition B.3. [Homogeneous M.C.] A M.C. is said to be homogeneous if $P(X_{t+1} = v|X_t = w) = P(X_1 = v|X_0 = w)$ for all $t > 0$.

Hence the homogeneity can be interpreted as a time invariance property.

Then for homogeneous Markov chains we can define the following matrix $P \in \mathbb{R}^{V \times V}$:

$$P_{vw} := P(X_1 = v|X_0 = w) \geq 0$$
It is easy to check that \( P \) is a stochastic matrix since \( \sum_{v \in V} P_{vw} = 1 \).

An homogeneous Markov chain is therefore fully characterized by the stochastic matrix \( P \) and an initial probability states vector \( \mu \).

**Definition B.4. [Stationary distribution]** A distribution \( \pi \) on \( V \) satisfying

\[
\pi = \pi P
\]  

(B.1)

is called a stationary distribution of the Markov chain. Clearly, if \( \pi \) is a stationary distribution and \( \mu = \pi \) (i.e. the chain is started in a stationary distribution), then the distribution of the states \( X_t \) in the Markov chain will be \( \pi \) for all times \( t \).

**Definition B.5. [Irreducibility]** A chain \( P \) is called irreducible if for any two states \( v, w \) there exists an integer \( t \) such that \( P^t_{vw} > 0 \).

This means that it is possible to get from any state to any other state using only transitions of positive probability.
In other words if and only if \( P \) is stochastic and \( G_P \) is strongly connected then \( P \) is irreducible.

**Definition B.6. [Reversibility]** An homogenous Markov chain is said to be reversible if there is a probability distribution \( \pi \) over the states such that:

\[
\pi_v P_{vw} = \pi_w P_{wv}
\]

Then summing the first part of the equation over \( v \) gives

\[
\sum_v \pi_v P_{vw} = \sum_v \pi_w P_{wv} = \pi_w \sum_v P_{wv} = \pi_w
\]

So for reversible Markov chains, \( \pi \) is a stationary distribution of the process since equation (B.1) is satisfied.

A state \( v \) is called **absorbing** if it is impossible to leave this state. Therefore, the state \( v \) is absorbing if and only if

\[
P_{vv} = 1 \text{ and } P_{vw} = 0 \text{ for } v \neq w.
\]

If every state can reach an absorbing state, then the Markov chain is an absorbing Markov chain.

**Definition B.7. [Hitting times]** Given a Markov chain with state space \( V \), the hitting time \( \tau_A \) of a subset \( A \subseteq V \) is defined as as the first time one of the nodes in \( A \) is visited by the chain. If \( X_t \) is
the process, we set
\[ \tau_A := \min_{t \geq 0} \{ X_t \in A \} \]

If \( A = \{ w \} \) we will simply write \( \tau_w \) instead of \( \tau_{\{w\}} \) for simplicity.

We remark the fact that \( \tau_A \) is also a random variable and depends from the initial probability vector \( \mu \).

In case of graphs with absorbing states set \( B \) we will be interested in the probability of being absorbed by a state \( b \in B \). We then define the exit or hitting probabilities as:

**Definition B.8. [Hitting probabilities]** Given a Markov chain with absorbing states set \( B \) the hitting probability is defined for every \( b \in B \) as
\[ \phi^b := P(\tau_B = \tau_b) \]

We remark the fact that \( \phi^b \) depends on the initial condition of the markov chain.

In the special case when \( B = \{ b_1, b_2 \} \), \( |B| = 2 \) and \( P|_{V \setminus B} \) is reversible we can define potentials (see next paragraph).

We can also define continuous time M.C. A Markov process can be thought of as a directed graph of states of the system. The difference is that, rather than moving to a new (even the same) state at each time step, the system will remain in the current state for some random (in particular, exponentially distributed) amount of time and then transition to a different state. The process is characterized by transition rates \( Q_{vw} \) between states \( v \) and \( w \). Let \( X_t \) be the random variable describing the state of the process at time \( t \), and assume that the process is in a state \( v \) at time \( t \). \( Q_{vw} \) (for \( v \neq w \)) measures how quickly that \( v \to w \) transition happens. Precisely, after a tiny amount of time \( h \), the probability the state is now \( w \) is given by:
\[ P(X_{t+h} = w | X_t = v) = Q_{vw}h + o(h) \text{ if } v \neq w \]

where \( o(h) \) represents a quantity that goes to zero faster than \( h \) goes to zero. Hence, over a sufficiently small interval of time, the probability of a particular transition (between different states) is roughly proportional to the duration of that interval. The \( Q_{vw} \) are called transition rates because if we have a large ensemble of \( n \) systems in state \( v \), they will switch over to state \( w \) at an average rate of \( nQ_{vw} \) until \( n \) decreases appreciably.

The transition rates \( Q_{vw} \) are typically given as the \( vw \)-th elements of the transition rate matrix \( Q \) (also known as an intensity matrix). As the transition rate matrix contains rates, the rate of departing from one state to arrive at another should be positive, and the rate that the system remains in a state should be negative. The rates for a given state should sum to zero, yielding the
diagonal elements to be

\[ Q_{vv} = - \sum_{w \neq v} Q_{vw} \]

The probability that no transition happens in some time \( r \) is

\[
P(X_s = v \ \forall \ s \in (t, t + r) \mid X_t = v) = e^{Q_{vv}r}
\]

That is, the probability distribution of the waiting time until the first transition is an exponential distribution with rate parameter \(-Q_{vv}\), and continuous-time Markov processes are thus memoryless processes.

Aperiodicity in the case of continuous M.C. is automatically enforced.

A particular and very useful case of a continuous-time Markov process is the following.

**Example B.0.1. [Homogeneous Poisson process]**

An homogeneous Poisson process is a continuous-time Markov process \( N(t), t \geq 0 \) with the following properties:

1. \( N(0) = 0 \)

2. Independent increments (the numbers of occurrences counted in disjoint intervals are independent from each other)

3. Stationary increments (the probability distribution of the number of occurrences counted in any time interval only depends on the length of the interval)

4. No counted occurrences are simultaneous.

Consequences of this definition include:

1. The probability distribution of \( N(t) \) is a Poisson distribution.

2. The probability distribution of the waiting time until the next occurrence is an exponential distribution.

3. The occurrences are distributed uniformly on any interval of time. (Note that \( N(t) \), the total number of occurrences, has a Poisson distribution over \((0, t]\), whereas the location of an individual occurrence on \( t \in (a, b] \) is uniform.)
The homogeneous Poisson process can be characterized by a rate parameter $\gamma$, also known as intensity, such that the number of events in time interval $(t, t + \tau]$ follows a Poisson distribution with associated parameter $\gamma \tau$. This relation is given as

$$\mathbb{P}[(N(t + \tau) - N(t)) = k] = \frac{e^{-\gamma \tau} (\gamma \tau)^k}{k!} \quad k = 0, 1, \ldots$$

where $N(t + \tau) - N(t) = k$ is the number of events in time interval $(t, t + \tau]$. $\gamma$ is the expected number of "events" or "arrivals" that occur per unit time.

$N(t)$ is a sample homogeneous Poisson process, not to be confused with a density or distribution function.

For further details on stochastic processes we refer to [27].
Bibliography


