Impact of Central Nodes in Information Propagation over Graphs

Oskar Mellegård

Master's thesis
2023:E5

Lund University

Faculty of Engineering
Centre for Mathematical Sciences
Mathematical Statistics

Centrum Scientiarum Mathematicarum

Impact of Central Nodes in Information Propagation over Graphs

Oskar Mellegård

Supervisor:

Prof. Stanislav Volkov

Examinator:

Prof. Erik Lindström

LUND
UNIVERSITY

Centre for Mathematical Sciences
Lund University

Populärvetenskaplig sammanfattning

Många system kan representeras som grafer, alltifrån neurala nätverk och transportnätverk till något så vardagligt som våra umgängeskretsar. Om exempelvis Andrea är vän med Jonathan, så kan vi rita ut Andrea och Jonathan som var sin prick på ett paper och låta deras vänskap representeras av en linje. På samma sätt kan vi enkelt lägga till fler personer (prickar) och rita fler linjer till alla deras vänner. Vi har då representerat ett socialt nätverk med hjälp av en graf; i ett matematiskt sammanhang så benämns prickarna som noder och linjerna som kanter. Relationerna (kanterna) kan förstås ha många olika innebörder. I en epidemi kan kanten representera att Andrea och Jonathan kan råka smitta varandra om den ena är sjuk. I sociala medier kan kanten representera att de är ”vänner” på plattformen. Relationen som de har i dessa exempel innebär alltså att information kan överföras mellan dem. Om Andrea har en sensationell nyhet som hon berättar för Jonathan så är det möjligt att han berättar vidare nyheten för sina vänner - som i sin tur kanske berättar för sina vänner. Informationen som är nyheten sprids då över nätverket som Andrea och Jonathan ingår i. En frågeställning i det här sammanhanget är hur snabbt informationen sprids – når nyheten ut snabbare om Andrea är den som orsakar spridningen, eller Jonathan? I det här exjobbet har vi studerat just den här frågeställningen, fast under mer generella ramverk.

När man studerar omfattande, komplexa nätverk är det eftersträvansvärt att kunna identifiera de mest centrala medlemmarna utifrån nätverksstrukturen. Avstampen för det här projektet bygger vidare på ett teoretiskt mått från en välciterad artikel av Philipp Bonacich, vilken mäter varje nods centralitet i en graf relativt de andra noderna. I projektet har därpå informationsflöden över grafer studerats och en spridningsmodell har för detta ändamål utvecklats; denna har tillämpats för att mäta hur stor inverkan centraliteten har för medlemmar som orsakar spridningarna över nätverk.

I graferna som vi studerar så *kommunicerar* noderna med varandra, vilket innebär att de ständigt transmitterar information till deras grannar. Varje nod tilldelas även en egen Markovkedja, vilken används för att ”aktivera” nodens roll som spridare och mottagare. Vi studerar informationsspridningar som induceras av just en enda nod, varifrån informationen fortsätter att spridas genom att varje nod delar med sig information till närliggande noder, så fort

noden själv är mottaglig och besitter informationen. Med hjälp av olika teoretiska mått mäter vi hur pass centrala noderna är; däribland noden som informationsspridningen börjar ifrån. Resultaten kan påvisa en koppling mellan hur snabbt information kan sprida sig över vissa nätverk och den teoretiska centraliteten hos medlemmen som orsakar spridningen; vi studerar spridningar över Barabási-Albert-nätverk i synnerhet, då dessa besitter egenskaper som kan observeras i riktiga nätverk, såsom sociala nätverk. Vilken nod som initierar spridningen har signifikant betydelse för hur snabbt informationen propagerar över grafen i helhet. Är det en central nod sprids informationen snabbare, vilket ligger i linje med förväntat resultat.

Abstract

There are many systems which can be represented as graphs, to say the least the networks in which we communicate with each other. Thorough understanding of graph structures enables better predictions of the dynamics in real life networks, such as the spreading of a disease in a community or failure propagation in a system.

This thesis investigates information propagation over connected undirected graphs, where the nodes communicate mutually. Every pair of nodes can send information to one another over the edge that connects them. The information propagation is initiated by a single node, which is the source of the spreading; our interest of research lies in how the time until the information has reached the entire graph relates to the theoretical centrality of this node. Furthermore, the thesis treats a power centrality measure proposed in an earlier paper by Phillip Bonacich. Our contribution in this regard is a rigorous derivation of a closed-form expression of the mean-measure and some properties appurtenant to it.

An Information Propagation Model (abbreviated IPM) is presented, devised to emulate the dynamics of mutual sharing of information between nodes. When performing this algorithm on certain Barabási–Albert graphs, results show that the centrality status of the initiator node has notable impact. In agreement with conception, in mean-limit, when the information propagation is initiated by a node with high centrality (according to the theoretical measure), the information is spread significantly faster on the graph. The IPM furthermore displays similar traits in dynamics to the SIR (susceptible-infectious-recovered) compartmental model.

Keywords: Graph theory, Markov chains, Systems theory, Functional analysis

Acknowledgements

I want to thank all the professors involved in this thesis; I thank you Stanislav Volkov (LU), Maria Sandsten (LU), Erik Lindström (LU) and Lukas Käll (KTH/SciLifeLab) for your guidance.

Table of Abbreviations

MC	*Markov Chain*
MBC	*Mean of the Bonacich Centrality*
BBM	*Bonacich β-Centrality Measure*
BAM	*Barabási–Albert Model*
SFN	*Scale-Free Network*
IPM	*Information Propagation Model*
PD	*Positive Definite (alt. Positive-Definiteness)*

Contents

Chapter 1

Introduction

Many human-made systems, some of which are critical infrastructure, can undoubtedly be interpreted as graphs. Economic networks, electrical power grids and social networks, to mention just a few. Consequently, the interest of graph theory has become commonplace in other academical fields, such as finance and artificial intelligence (Zhou et al., 2020). The same can of course be said about industrial applications, as for instance graph databases have gained attention during latter years.

In the analysis of graphs, *centrality* is a key notion and fundamental research problem which involves ranking the importance of nodes' positions in the graph structure. There are as of date numerous measures which determine the centrality with respect to various qualities. For some of these measures, indications have been found empirically that the rankings commonly are highly correlated (Benzi & Klymko, 2015). The discrete and combinatorial structures of graphs allows for a suitable environment to perform algorithms on. If the graph represents the structure of a "real life" system, the dynamics can be tried to be modeled for e.g., the spreading of a disease in a community; if a healthy and a sick individual are respectively represented by nodes, the possibility of infection spread from one to the other can be represented by the edge that connects them.

There are graph generative models which are able to randomly generate graphs which possess features similar to real life networks. An example is the Barabási–Albert preferential attachment model, which generates so called *scale-free networks*. The degree distribution of the networks produced by this model follow a power-law, meaning that some nodes (which in general are relatively few) are expected to have substantially more connections than the vast majority, which is a property that can be observed in many real life networks such as e.g., social networks. Graph generative models enable large-scale simulations which involves many graphs and one may study the graph structures in light of various models performed on them, as well as the models solely.

Advancements in the study of graph (network) structures may have profound impact in how we are able to predict and understand dynamics of real systems. In this thesis, the notion *information propagation* is commonly used, which is the spread of information induced by the nodes. Within the framework of this project, *information* is defined to be anything a node can possess and transmit. Revisiting our previous examples, a disease is for instance something an individual can "possess" as well as transmit to other individuals, and thus it is indeed some form of information in this framework. The dynamics of massive-scale spread of diseases motivated this thesis project to investigate the impact of the centrality of the nodes which initiate the spread; in this analogy this corresponds to what place the *patient zero* has in the network structure corresponding to the community he or she lives in. Furthermore, what impact this single individual has on transmission globally.

As for centrality measures, promising applications are e.g., within risk mitigation, where cascading failures can be studied via dependency risk graphs; here, the nodes can represent critical infrastructures or components and the edges how the failure of one component can lead to the failure of others; see for instance (Stergiopoulos et al., 2015). Centrality is also a widely studied concept within computer science applications (Klein, 2010).

1.1 Thesis Objectives

It is primarily aimed to investigate how big of an impact central nodes have in information propagation over synthetic graphs. The centrality of the nodes shall be determined by various theoretical measures; it is of interest therefore to see how the theoretical centrality of specific nodes, whom are initiating the information propagation, affects the time until the information has reached all nodes in the graph. The information is guaranteed to propagate through the graph, as the nodes are mutually sharing the information with adjacent nodes. In theoretical treatment, all centrality measures assess the nodes involvement in the walk structure of the graph (Borgatti & Everett, 2006). The research problem is to investigate how theoretical measures relate to real settings.

Another objective is to derive the closed form and the domain of the Bonacich β-centrality measure $c(1, \beta)$ which was proposed in the article "Power and Centrality: A Family of Measures" in *American Journal of Sociology* (1987) by Philipp Bonacich; see Definition 2.11, p.15 in this thesis for the definition of the measure. A proof is not presented in the article which regards the parameter choice of β, for which the closed form of the measure can be defined. Another article by Bonacich presents a thorough derivation (Bonacich, 2007); although, this thesis takes another approach of the proof, from which further mathematical treatment of the measure's properties is conducted. Brief exploration of the parameter β is sufficient to learn that the relative centrality statuses of the nodes may indeed change. That is, a node which is more central than another node for some quantity β may not be for another quantity $\beta' \neq \beta$. An example which demonstrates this fact is shown in Figure 1.1, where the underlying graph is generated according to the Barabási–Albert graph generative model, with $|V| = 10$ nodes; see Appendix A for the simulation program. The abscissa in figure shows an interval of defined choices of β, and the ordinate shows the elements of $c(1, \beta) \in \mathbb{R}^{|V|\times 1}$ as functions of β. In other words, each curve corresponds to the centrality score as a function of β for the node it corresponds. The dashed circle in figure highlights a node overtaking another in centrality score. To contribute to more insight in this phenomena and facilitate further analysis, the mean of the Bonacich β-measure (abbreviated MBC in report) is derived over symmetric integration intervals, given that we treat the measure as a function $c : \mathbb{R} \to \mathbb{R}^{|V|\times 1}$. In addition to the derivation, some properties of the MBC are proposed and proven. The β-measure shall furthermore be studied qualitatively against the *eigenvector centrality*, the *degree centrality* and the MBC; the reader is referred to Sections 2.3 and 2.5, respectively.

As for related work, parts of this thesis is based on some of the results in the article "Power and Centrality: A Family of Measures" in *American Journal of Sociology* (1987) by Philipp Bonacich. Graph centrality is a fundamental research problem within graph theory and work which is based on the aforementioned article is for instance "Some unique properties of eigenvector centrality" published in *Social Networks* (2007) by the same author.

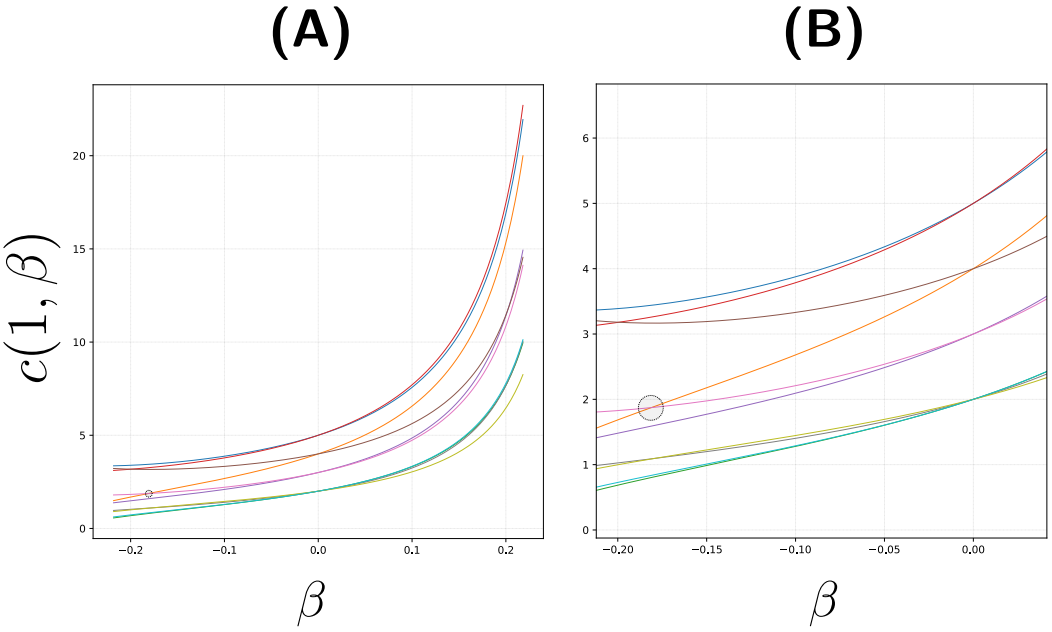

Figure 1.1: The curves correspond the centrality score for each node versus the hyper-parameter choice β, over the domain which the Bonacich centrality measure is defined. The graph on which the scores are computed with respect to is generated according to the Barabási–Albert Model (BAM). The dashed circle in figure marks a point where a node becomes more central than another, which previously was more central. The reader shall be made aware that multiple intersection can be observed in the figure. The plot **(B)** is a zoom-in of the plot **(A)**. The mentioned point of the overtaking in centrality is marked in both figures.

1.2 Thesis Outline

The thesis begins with providing a brief overview of elementary graph theory in the introductory sections of Chapter 2. In the latter sections of this chapter, the closed form of $c(1, \beta)$ is derived and the mean of the β-centrality is proven; some properties of this measure are also proposed and proven. A section is dedicated to the *eigenvector centrality*, which is related to the β-centrality. Furthermore in Chapter 2, an overview of Barabási–Albert model is given, which is later on used to generate graphs which the algorithms are performed on. Characteristically for the BAM as a graph generative model, it produces scale-free networks by its preferential attachment mechanism. That is, nodes that are "entering" the graph are more likely to induce edges which connects them to highly connected nodes already within the graph. In Chapter 3, a brief overview of Markov chains and appurtenant theory is provided;

these have a fundamental role in the information propagation model as they are used to regulate the nodes' roles in the transmission. In Chapter 4, the information propagation model (IPM) is presented. Chapter 5 provides the simulation results and the follow up chapters from there convey discussions and suggestions for further work.

Chapter 2

Graph Theory

The introductory part of this chapter is aimed to provide an overview of fundamental theory of mathematical graphs. Appurtenant to this theory, the chapter afterwards goes in depth of various centrality measures, which determine the importance of the nodes' positions in the graph structure.

A graph may be regarded as a set of objects over which a set of relations act. Such relations specifically act on pairs of objects. Ordinarily, the objects are referred to as *nodes* (or *vertices*) and they may individually contain attributes which may be whatever: a number, a vector, a matrix, an operator or e.g., all mentioned at the same time, under some ordering. The treatment of a *relation* naturally leads to the notion of an *edge*. Fundamentally, an edge is a link between two nodes which affirms that there is some relation between them; the direction of the edge tells *how* this relation acts. An edge may be directed or undirected. As for the directed edge, or in literature called *arc*, such is pointing from one node to another, the latter which may be regarded as the succeeding node. The direction tells how the edge can be traversed. An undirected edge on the other hand points simultaneously in both directions and may thus be traversed in either direction. Like nodes, any edge may also contain attributes of any form. If such is simply a number, then conventionally it is referred to as the *weight* of the edge. In terms of notation, the set of nodes is denoted by V, the set of edges by E and the graph in its entirety by $G := \{E, V\}$. The cardinality $|\cdot| : \Omega \to \mathbb{Z}$ counts the number of elements in some set Ω. As for the sets of edges and vertices E and V for some given graph, it shall be interpreted e.g.,

$$|V| = \text{number of nodes}$$

and so forth.

Most graphs can be depicted diagrammatically, by which the names for the notions *edges* and *nodes* are made more clear. In such form, each node is represented by a circle or dot (hence they are "nodes"), and the edges by straight lines. The direction of the edge is simply represented by an arrow. If undirected though, such line by convention does not display any arrow(s), although it shall be remarked that such line indeed is "doubly" directed, as made

clear in the above paragraph. As an example, Figure 2.1 displays an undirected, connected planar graph in its diagrammatic form.

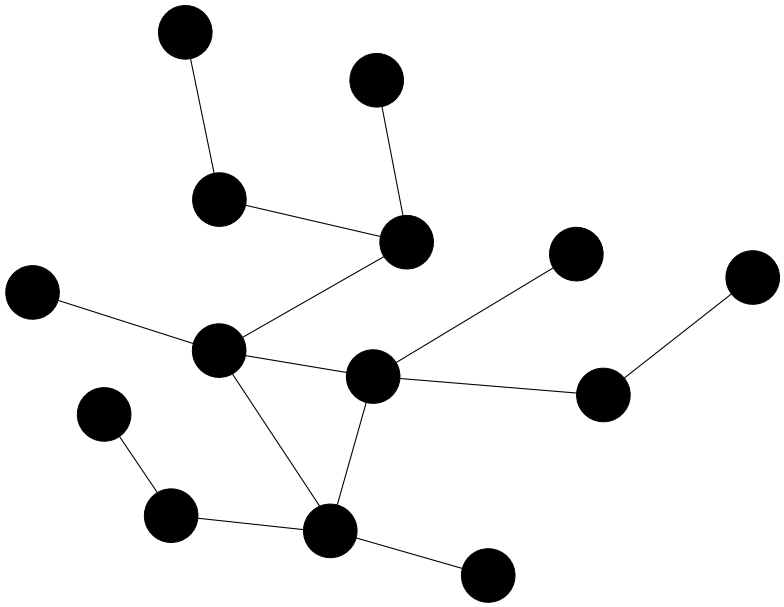

Figure 2.1: A planar, connected undirected graph.

Throughout the thesis, we let $G := \{E, V\}$ refer to some graph over some bijection $\mathcal{H} : V \to [1, |V|] \subseteq \mathbb{Z}$ which maps a unique identifier (or, so called *label*) to each node $v \in V$. In other words, each node is given a distinct number, and only that node. Thus, for any given pair of nodes v, $w \in V$, such bijection enforces $v = w$ if and only if $\mathcal{H}v = \mathcal{H}w$ i.e., two nodes are equal if and only if they have the same identifier. In all contexts of this thesis, a node will be referred by v_i where i denotes its unique identifier. A directed edge from some node v_i to v_j is denoted by (i, j). As regards undirected edges, if v_i and v_j are adjacent, then the edge which connects them $(i, j) \in E$ is unordered i.e., $(i, j) \sim (j, i) \in E$. The method development in this work is based on *connected undirected* graphs i.e., graphs in which each edge is undirected and there is a path from any node v_i to any node v_j. A more formal definition is as follows.

Definition 2.1. *A (completely) undirected graph $G := \{E, V\}$ is a graph in which all edges are undirected. That is, any edge $(i, j) \in E$ which connects two nodes v_i and v_j is unordered and the edge may be traversed in either direction therefore. If furthermore the graph is connected, there is a walk from any node v_i to any node v_j in V; see Definition 2.3 as regards walks.*

Fundamental notions are the node-degree, *walks* and the adjacency matrix and the definitions for these are as follows.

Definition 2.2. *The degree of a node $v_i \in V$ equals the number of edges which are connected to it*

$$\deg v_i := |\{(i,j) \in E : v_j \in V\}| = \#\text{ edges connected to } v_i \in \mathbb{R}_{\geq 0}.$$

Definition 2.3. *A **walk** on an undirected graph $G := \{E, V\}$ is a traverse and is expressed as a sequence of vertices and nodes, beginning in some node $v_i \in V$ and ending in some node v_ℓ,*

$$v_i,\ (i,j), v_j, (j,k), v_k ... v_\ell.$$

*If $v_i = v_\ell$ then the walk is **closed**, otherwise it is **open**.*

Remark. A *walk* may conceptually be imagined as a particle traversing or "walking" along the edges of the graph. If an edge is directed, the particle can not walk the in the opposite of its direction and may thus not go back and forth along the same edge, repeatedly. If an edge is undirected however (which is recalled as "doubly" directed) it may freely walk in either direction along it. In other words e.g.,

$$v_i,\ (i,j), v_j, (j,i), v_i, (i,j) ...$$

is allowed.

Definition 2.4. *The adjacency matrix $\mathbf{A} = (A_{ij}) \in \mathbb{R}^{|V|\times|V|}$ of some graph $G := \{E, V\}$ is defined by,*

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

Remark I). The adjacency matrix $\mathbf{A} \propto G$ stores all information that concerns the neighbor–relations in the graph and has for instance applications in graph spectrality and in node centrality measures; see sections 2.4 and 2.5 in this report. In addition to this remark, it shall be pointed out that no node is defined to be adjacent to itself, why $A_{ii} := 0$.

Remark II). Constructing the adjacency matrix $\mathbf{A}$ from a connected and undirected graph may be justified as follows. As recalled, two nodes $v_i,\ v_j \in V$ are adjacent if they are connected by an edge, that is $(i,j) \sim (j,i) \in E$ for an undirected graph. If so, the entries $A_{ij},\ A_{ji}$ are set as 1 according to definition 2.4. If they are not, both entries $A_{ij},\ A_{ji}$ are set as 0. The nature of this construction imposes several properties on $\mathbf{A}$. Most fundamentally $\mathbf{A} \in \mathbb{R}^{|V|\times|V|}$, as the adjacencies are established over each combination of pairs of nodes in the graph. Furthermore, $\mathbf{A}$ clearly is symmetric, non–negative and irreducible (Stevanovic, 2014).

Remark III). The adjacency matrix has the following fascinating property. If raised to some power $n \in \mathbb{N}_{\geq 0}$, that is $\mathbf{A}^n$, then the number of walks of length n starting in node $v_i \in V$ and ending in $v_j \in V$ equals $(\mathbf{A}^n)_{ij}$, where repetitions are allowed (Stevanovic, 2014). Moreover, summing the i:th row of $\mathbf{A}^n$ gives the total number of walks of length n from $v_i \in V$ to any node in V, including v_i itself.

2.1 Bipartite Graphs

Circumstances in which there are allocations over two classes – for instance resources and consumers – may suitably be interpreted under *bipartite graph* structures. This thesis takes particular interest in when such allocations in sense are bijective. An element from either class may in other words be *compatible* with multiple elements from the other class, although in the end, it will pair with one and only one individual from the other class and vice versa due to the bijection.

The mathematical definition of a bipartite graph is as follows.

Definition 2.5. *A graph $G := \{E, V\}$ is bipartite if the nodes of G can be represented as,*

$$V = \mathcal{U} \cup \mathcal{V}$$

where $\mathcal{U}$, $\mathcal{V}$ are disjoint sets of nodes i.e., $\mathcal{U} \cap \mathcal{V} = \emptyset$ and every edge in E has one end-point in $\mathcal{U}$ and the other in $\mathcal{V}$.

More ordinarily, and in particular in computer science, one may say that nodes of $\mathcal{U}$ and $\mathcal{V}$ are *coloured* and furthermore, that an edge may only link two nodes if they are of different colours – which follows directly from definition 2.5. To make detailed of this concept, let for instance all nodes in $\mathcal{U}$ be *white* and the nodes in $\mathcal{V}$ be *black* (or generally, any other colour than *white*). An edge may therefore only link two nodes if one is *white* and the other is *black*. Consequently, a graph fails to be bipartite if at least one edge does not fulfill this criteria. Figure 2.2 illustrates a basic bipartite graph, in which it shall be noted that each edge has one end-point in a *white* node and the other in a *black* node.

Complete bipartite graphs have a central role in the theory progress of this thesis, as these are used as initial graphs in application of graph generative models.

Definition 2.6. *A bipartite graph $G := \{E, V\}$, for which $V = \mathcal{U} \cup \mathcal{V}$ according to definition 2.5, is* ***complete*** *if any node in $\mathcal{U}$ is adjacent to any node in $\mathcal{V}$ i.e., there is an edge connecting any node in $\mathcal{U}$ to any node in $\mathcal{V}$.*

Figure 2.3 gives an illustration of a complete bipartite graph in which $|\mathcal{U}| = 3$ (white nodes) and $|\mathcal{V}| = 4$ (black nodes).

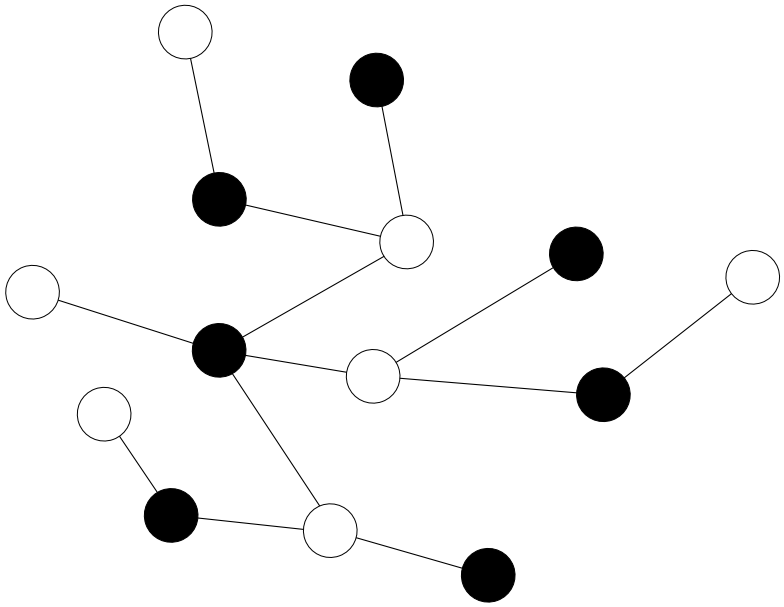

Figure 2.2: A bipartite graph where $\mathcal{U}$ = {white nodes} and $\mathcal{V} = \{$black nodes$\}$.

There are various circumstances which allow for a complete bipartite graph interpretation; it may be illustrative to consider the following hypothetical one. Suppose there are m hospitals departments and n doctors. Each department treats rare cases of illnesses e.g., one might treat tropical diseases, another one rare neurological conditions and so forth. Assume that each department only has the capacity for one doctor, due to cut-downs in the welfare. The doctors each are assumed to have expertise in different fields and it could be theorised that a "wrongly" placed doctor i.e., he or she lacks desired competency of the department, puts the patients at more risk compared to one who is expert in the demanded field of the department. If it is not absolutely obvious at first which doctor shall be placed at which department, the hospital could consider the case where each doctor potentially may be placed at either department. That is, $\mathcal{U} = \{$doctors$\}$ and $\mathcal{V} = \{$departments$\}$ and hence, in mathematical treatment, there is an edge from any doctor $u \in \mathcal{U}$ to all other departments $v \in \mathcal{V}$ which represents the possibility of such assignment; recall Figure 2.3. If the risk can be assessed and *quantified* for each doctor and department, it may be represented as the weight of corresponding edges. Thus, the problem of assigning each doctor a department with regard of minimizing the patient risk, may be reduced to finding the set of edges which weights accumulates to the lowest risk value. This kind of problem is generally referred to the *linear assignment problem*.

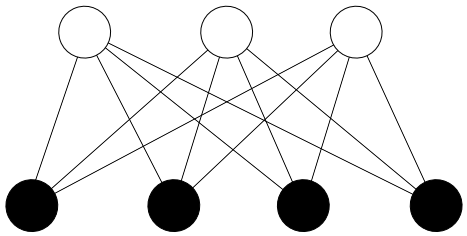

Figure 2.3: A complete bipartite graph for white nodes $\mathcal{U}$ = {doctors} and black nodes $\mathcal{V}$ = {departments}.

Definition 2.7. *A **star** $S_k := \{E, V\}$ is a complete bipartite graph in which any two vertices are connected by exactly one path; the graph S_k has one internal node and any other node is connected to the inner node via an edge, hence there are no edges connecting the leaves except the case for $k = 1$. The graph corresponding $\mathcal{S}_{k \leq 1}$ has $k + 1$ leaves, but it does not have any internal nodes.*

Figure 2.4 shows the star graphs $\mathcal{S}_3$, $\mathcal{S}_4$ and $\mathcal{S}_5$.

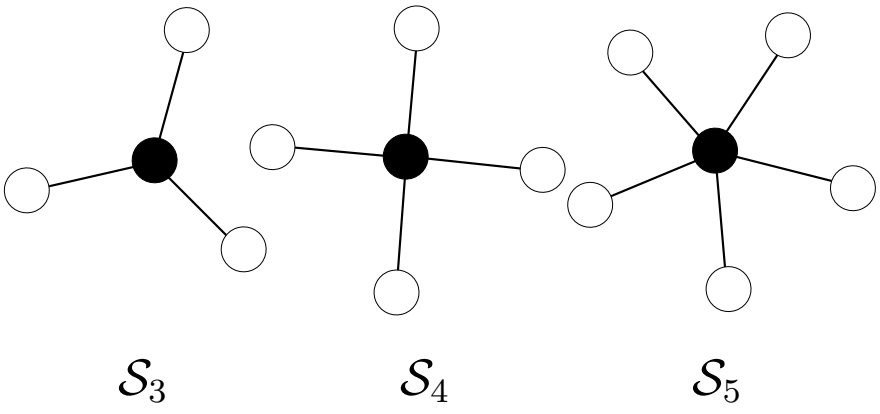

Figure 2.4: The star graphs $\mathcal{S}_3$, $\mathcal{S}_4$ and $\mathcal{S}_5$. The internal nodes have black colour whereas the leaves are white.

2.2 Spectral Graph Theory

The objective of this section is to provide a brief overview of the spectral properties of connected undirected graphs. In view of the properties imposed on an adjacency matrix $\mathbf{A}$ of an undirected graph by Remark II) under Definition 2.4, it shall be reminded that $\mathbf{A} \in \mathbb{R}^{|V|\times|V|}$ is symmetric, non–negative, irreducible and furthermore $(\mathbf{A})_{ii} = 0$ as no node is adjacent to itself.

The *spectrum* of a graph G is the set of all eigenvalues of its adjacency matrix $\mathbf{A} \propto G$.

Definition 2.8. *The spectrum of a graph $G := \{E, V\}$ is defined as*

$$\begin{aligned}\mathrm{Spec}(G) &:= \{\lambda \in \mathbb{C} \,:\, \det(\mathbf{A} - \lambda \cdot \mathrm{Id}) = 0\} \\ &= \{\lambda_1^{(m_1)}, \lambda_2^{(m_2)}, ...\}\end{aligned}$$

where m denotes the multiplicity. Here, $\mathbf{A}$ is the adjacency matrix of G and $\mathrm{Id} \in \mathbb{R}^{|V|\times|V|}$ is the identity matrix.

As the adjacency matrix of an undirected graph is real, non-negative and symmetric, it follows that any $\mathbf{A} \in \mathbb{R}^{|V|\times|V|}$ has $|V|$ number of real, simple eigenvalues $\{\lambda_i\}_{i=1}^{|V|}$ which make up the spectrum i.e., the cardinality is $|\mathrm{Spec}(G)| = |V|$.

The largest eigenvalue of some adjacency matrix $\mathbf{A} \propto G$ is throughout this thesis referred to as $\hat{\lambda}$ for which

$$\hat{\lambda} := \max_i \lambda_i(\mathbf{A}) \tag{2.1}$$

and it plays a particular role in establishing convergence of the *Bonachich β-centrality*; see section 2.4 for further details. Furthermore, as $\mathbf{A} \propto G$ regards connected graphs G, it follows that $\mathbf{A}$ is irreducible and thus, the Perron-Frobenius theorem is applicable on $\mathbf{A}$.

Theorem 2.1. (Perron-Frobenius) *If $\mathbf{A} \in \mathcal{M}_n(\mathbb{R}_{\geq 0})$ is irreducible, then the largest eigenvalue $\hat{\lambda}$ of $\mathbf{A}$ defined by equation* (2.1) *has the following properties (Stevanovic 2014, Horn & Johnson 2012):*

i) $\rho(\mathbf{A}) = \hat{\lambda} \in \mathbb{R}^+$ where ρ is the spectral radius and $\hat{\lambda}$ is simple.

ii) under the ordering $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n$, then $\hat{\lambda} := \lambda_1 \geq |\lambda_i|$ for all $i \geq 2$.

iii) there is an eigenvector $\mathbf{v}_+$ of $\mathbf{A}$ which $\hat{\lambda}$ corresponds to an, for which all components are positive.

Remark I). From iii) it is implied that there exists infinitely many eigenvectors $\bar{\mathbf{v}}_+$ (up to scaling) in which all elements are non-negative and non-zero and $\mathbf{A}\bar{\mathbf{v}}_+ = \hat{\lambda}\bar{\mathbf{v}}_+$.

Remark II). In computational aspects of the *Bonachic centrality* series defined by equation (2.7), the largest eigenvalue $\hat{\lambda}$ of the adjacency matrix is vital in determining the convergence bounds of β. Moreover, $\hat{\lambda}$ is also relevant as regards the *eigenvector centrality* defined by equation (2.10). Numerical methods may be used for computing the spectrum $\mathrm{Spec}(G)$. It shall in the same sentence be briefly remarked that *power iteration* computes solely the largest eigenvalue of a matrix which is diagonalizable. As regards a connected undirected graph G, the corresponding adjacency matrix $\mathbf{A} \propto G$ is real and symmetric (recall Remark II in Definition 2.4), why it is diagonalizable and defined for the power iteration algorithm.

2.2.1 The Laplacian of Graphs and Connectivity

One may, in fact, conclude whether or not a graph $G := \{E, V\}$ is connected by studying its spectral properties. It may be elegantly done using a theorem proven by Miroslav Fiedler; in order to fully assimilate it, consider firstly the *Laplacian* of a graph G.

Definition 2.9. (The Laplacian matrix) *The graph-Laplacian matrix $\mathcal{L} \in \mathbb{R}^{|V|\times|V|}$ of some graph G is defined by,*

$$(\mathcal{L})_{ij} = \begin{cases} \deg(v_i) & \text{if } i = j \\ -1 & \text{if } (i,j) \in E \\ 0 & \text{otherwise.} \end{cases}$$

The graph-Laplacian matrix may also be expressed as $\mathcal{L} = \mathbf{D} - \mathbf{A}$ where

$$\mathbf{D} = \mathrm{diag}(\deg v_1, ..., \deg v_{|V|}),$$

is a diagonal-matrix containing the node degree of node i at place (i, i) and $\mathbf{A}$ is the adjacency matrix of G.

Remark I). It shall be noted that the node degree(s) $\deg v_i$ may be obtained by the product $\mathbf{A}\mathbb{1}$, for which case the i:th row corresponds to the degree of node v_i.

Remark II). The Laplacian is positive semi-definite and the smallest eigenvalue is $\lambda_1(\mathcal{L}) = 0$.

Astonishingly, the connectivity of an undirected graph is related to the spectral property of its graph-Laplacian; if the *second smallest* eigenvalue of $\mathcal{L}$ is strictly positive, then it is equivalent to the graph being connected (de Abreu, 2007). If $\lambda_2(\mathcal{L}) = 0$ therefore, there are at least two disconnected components of the graph, meaning that for some nodes it is impossible to reach one another by *any* walk on the graph. Furthermore, values $\lambda_2(\mathcal{L}) \sim 0$ may be interpreted as the graph being *close* to disconnected.

2.3 Eigenvector Centrality

The *eigenvector centrality* measures the centrality of nodes in an undirected graph $G := \{E, V\}$ by taking into account the neighbors' significance. To further elaborate, the centrality of a node is set as proportional to the sum of centrality scores of its neighbors. The measure is justified as follows; let x_i denote the centrality score of some node $v_i \in V$ and let $\mathbb{I}$ be the index set of all neighbors of v_i and

$$x_i = \kappa^{-1} \sum_{j \in \mathbb{I}} x_j, \tag{2.2}$$

for some proportionality constant $\kappa \in \mathbb{R}$. Now, recall for the adjacency matrix $\mathbf{A}$ that $(\mathbf{A})_{ij} = 1$ if the nodes v_i and v_j are adjacent and 0 otherwise. Therefore, the formula equation (2.2) in terms of $\mathbf{A} \propto G$ equates to

$$x_i = \kappa^{-1} \sum_{j=1}^{|V|} (\mathbf{A})_{ij} x_j. \tag{2.3}$$

It shall be noted by equation (2.3) that a node v_i attains high centrality if e.g., it has many neighbors (under assumption that they have positive centrality scores) or by having highly influential neighbors which themselves have high level of centrality – or both. In matrix form, equation (2.3) is equivalent to

$$\mathbf{A}\bar{\mathbf{x}} = \kappa \bar{\mathbf{x}}, \tag{2.4}$$

where the i:th row of $\bar{\mathbf{x}}$ correspond to the centrality score of node v_i under some parameter κ. Thus, as may be noted from equation (2.4), $\bar{\mathbf{x}}$ is an eigenvector of the adjacency matrix $\mathbf{A}$ and κ is an eigenvalue. If positive centrality scores are desired i.e., every component in $\bar{\mathbf{x}}$ shall be strictly positive, it is implied by the Perron-Frobenius theorem that an eigenvector which can guarantee this property is one which corresponds to the eigenvalue $\kappa = \hat{\lambda}$. See Theorem 2.1. Therefore, after determining $\hat{\lambda}$ and $\bar{\mathbf{x}}$ correspondingly, one may normalize it according to

$$\bar{\mathbf{x}}_+ := \frac{\operatorname{sgn}(\bar{\mathbf{x}})}{\|\bar{\mathbf{x}}\|_2} \bar{\mathbf{x}}, \tag{2.5}$$

where $\operatorname{sgn}(\bar{\mathbf{x}}) := \pm 1$ has the same sign as the components in $\bar{\mathbf{x}}$ and $\|\cdot\|_2$ is the ℓ^2 norm. This guarantees that all elements become strictly positive.

Definition 2.10. **(Eigenvector centrality)** *The eigenvector centrality of an undirected graph $G := \{E, V\}$ is determined from any eigenvector $\bar{\mathbf{x}}$ which corresponds to the eigenvalue $\hat{\lambda} = \max \lambda(\mathbf{A})$,*

$$\bar{\mathbf{x}}_+ := \frac{\operatorname{sgn}(\bar{\mathbf{x}})}{\|\bar{\mathbf{x}}\|_2} \bar{\mathbf{x}}. \tag{2.6}$$

The i:th row of $\bar{\mathbf{x}}_+$ is the centrality score of node $v_i \in V$ under the normalization equation (2.5).

2.4 Bonacich β-Centrality

The Bonacich β-centrality measure is a centrality measure used widely in network analysis, which quantifies the power of influence of a node. Its mathematical definition is as follows.

Definition 2.11. (Bonacich β-Centrality) *The Bonacich β–centrality measure (BBM) scores the relative level of centrality of each node $v_i \in V$ of an undirected, connected graph $G := \{E, V\}$ which has adjacency matrix $\mathbf{A} \in \mathbb{R}^{|V|\times|V|}$,*

$$c(1,\beta) = \sum_{k=0}^{+\infty} \beta^k \mathbf{A}^{k+1}\mathbb{1}, \tag{2.7}$$

where $\beta \in \mathbb{R}$ and $\mathbb{1}$ is a $|V|$ dimensional column vector in which all entries equal 1. The i:th row of $c(1,\beta)$ contains the centrality measure of node v_i (Bonacich, 1987).

The BBM equation (2.7) has an expected value interpretation under certain conditions for $\beta > 0$. Based on the original article, assume firstly that each neighboring pair of nodes may communicate mutually with each other. Moreover, let β be the probability that once a message has been sent, it will be passed on by the receiving node, to *any* of its neighbors. A message can not be passed on until it has been received, although initially each node sends out messages to all of its neighbors. Recalling Remark III) under Definition 2.4 of the adjacency matrix, it shall be noted that for power $n = 1$ i.e., $\mathbf{A}^1$, the number of walks of length $n = 1$ from v_i to any other node in v, including itself, equals the sum of the i:th row of $\mathbf{A}$. Note also that $(\mathbf{A}^1)_{ii} = 0$, as naturally there are no walks of length 1 which immediately lead back to the node itself. Thus, the i:th row-sum for $\mathbf{A}$ equals the degree of v_i and is also the number of initiated walks by node v_i i.e., the number of messages initially sent,

$$\#\text{ initiated walks from } v_i = \deg v_i = \sum_{j=1}^{|V|} A_{ij} = (\mathbf{A}\mathbb{1})_i. \tag{2.8}$$

Component-wise, the centrality for any node $v_i \in V$ is determined by

$$c_i = \sum_{j=1}^{|V|}\sum_{k=0}^{+\infty} \beta^k (\mathbf{A}^{k+1})_{ij}, \quad (\mathbf{A}^k)_{ij} = \#\text{ walks of length } k \text{ from } v_i \text{ to } v_j.$$

That is, c_i is the weighted (or "dampened") sum of all walks, of any length, from v_i to any other node $v_j \in V$, including v_i itself. The dampening effect may be explained as follows. Assuming that the graph is non-trivial i.e., $|V| > 2$, one obtains the lower bound $\hat{\lambda} > 1$; the min-max theorem can be applied to compute the largest eigenvalue, and it follows

$$\hat{\lambda} = \sup_{\mathbf{x}^\mathsf{T}\mathbf{x}=1} \mathbf{x}^\mathsf{T}\mathbf{A}\mathbf{x} \geq \frac{1}{\sqrt{|V|}}\mathbb{1}^\mathsf{T}\mathbf{A}\left(\frac{1}{\sqrt{|V|}}\mathbb{1}\right) = \frac{1}{|V|}\sum_{v\in V}\deg v = \frac{2|E|}{|V|} \geq \frac{2(|V|-1)}{|V|} > 1.$$

In accordance with the parameter bound equation (2.9), $\hat{\lambda} > 1$ implies $|\beta| < 1$ and naturally $\beta^k \to 0$ as $k \to +\infty$. In analogy with the expectation-interpretation i.e., given $\beta \in \mathbb{R}^+$, the fact that $\beta^k \to 0$ may be interpreted as walks of length $k \to +\infty$ are asymptotically impossible, irrespective of how many they are and for any given start and end node in the graph. Small values of β puts more weight on the local structures network whereas larger values, the structure globally (Bonacich, 1987). It can be proven that convergence of the series equation (2.7) can be established if

$$|\beta| < \frac{1}{\hat{\lambda}}, \tag{2.9}$$

where $\hat{\lambda}$ is the largest eigenvalue of $\mathbf{A}$. The closed form of equation (2.7) and inequality above may be jointly proven as follows, wherein it shall be noted that β may indeed also be negative.

Proposition 2.1. *The β–centrality equation* (2.7) *converges if $|\beta| < 1/\hat{\lambda}$ where $\hat{\lambda} := \max \lambda(\mathbf{A})$ and has the closed form,*

$$(\mathrm{Id} - \beta\mathbf{A})^{-1}\mathbf{A}\mathbb{1},$$

where Id *is the identity matrix of dimension $|V| \times |V|$.*

Proof. Firstly, it shall be noted that equation (2.7) is equivalent to,

$$c(1, \beta) = \left\{ \sum_k (\beta\mathbf{A})^k \right\} \mathbf{A}\mathbb{1}. \tag{2.10}$$

The spectral radius of $\beta\mathbf{A}$ clearly is,

$$\rho = |\beta|\hat{\lambda}.$$

It follows from the Perron–Frobenius theorem that $\hat{\lambda}$ is strictly positive and is larger than or equal to the absolute values of all other eigenvalues that regards $\mathbf{A}$; see Theorem 2.1. Moreover, the bracket in equation (2.10) shall be recognised as a Neumann series, which converges to the closed form if and only if $\rho < 1$,

$$\sum_k (\beta\mathbf{A})^k = (\mathrm{Id} - \beta\mathbf{A})^{-1},$$

and the proof may be concluded,

$$|\beta| < \frac{1}{\hat{\lambda}}.$$

■

The proof of Proposition 2.1 may also be conducted by proceeding from the following theorem.

Theorem 2.2. *If X is a Banach space and the bounded linear operator represented by $\mathbf{A}$ has an operator norm within the unit circle i.e., $\|\mathbf{A}\|^* < 1$, then $(\mathrm{Id} - \mathbf{A})^{-1}$ exists and is furthermore bounded; the series converges in operator norm,*

$$(\mathrm{Id} - \mathbf{A})^{-1} = \sum_{k=0}^{+\infty} \mathbf{A}^k.$$

See (Renardy & Rogers, 2010).

The second proof of the closed-form expression of the BBM accordingly follows.

Proof of Proposition 2.1. Firstly, given that $\beta\mathbf{A}$ is linear, bounded and the adjacency matrix $\mathbf{A}$ is real and symmetric, the operator norm can be determined by

$$\begin{aligned}
\|\beta\mathbf{A}\|^* &= \max_i \sqrt{\lambda_i((\beta\mathbf{A})^\intercal(\beta\mathbf{A}))} \\
&= \max_i |\beta| \sqrt{\lambda_i(\mathbf{A}^2)} \\
&= \max_i |\beta||\lambda_i(\mathbf{A})| \\
&= |\beta|\hat{\lambda},
\end{aligned}$$

given $\lambda_i(\mathbf{A}^2) = \lambda_i^2(\mathbf{A})$. Therefore, if $|\beta|\hat{\lambda} < 1$ and equivalently $|\beta| < 1/\hat{\lambda}$, then according to Theorem 2.2,

$$(\mathrm{Id} - \beta\mathbf{A})^{-1} = \sum_{k=0}^{+\infty} (\beta\mathbf{A})^k.$$

This concludes the proof,

$$\begin{aligned}
c(1,\beta) &= \sum_{k=0}^{+\infty} \beta^k \mathbf{A}^{k+1} \mathbb{1} \\
&= \left\{ \sum_{k=0}^{+\infty} \beta^k \mathbf{A}^k \right\} \mathbf{A}\mathbb{1} \\
&= (\mathrm{Id} - \beta\mathbf{A})^{-1} \mathbf{A}\mathbb{1}.
\end{aligned}$$

■

A remark to the BBM is that the measure *usually* converges to the eigenvector centrality when $\beta \to (1/\hat{\lambda})^-$ i.e.,

$$\lim_{\beta \to (1/\hat{\lambda})^-} c(1,\beta) = \bar{\mathbf{x}}, \tag{2.11}$$

where $\bar{\mathbf{x}}$ is a scaling of the eigenvector corresponding to the largest eigenvalue of $\mathbf{A}$ (see Definition 2.10), if $\hat{\lambda}$ distinctively is the largest eigenvalue of the graph. There are cases of symmetrical graphs where the limit does not hold – the reader is referred to the examples which are presented in the original paper (Bonacich, 2007).

The BBM is versatile in the aspect that other centrality measures can be derived from it, for certain choices of β. To elaborate, as aforementioned, if $\beta \to (1/\hat{\lambda})^{-}$ one usually obtains the eigenvector scores; if $\beta := 0$, then instead one obtains the degree centrality scores i.e., the i:th row of $c(1,0)$ equals the degree of node v_i. This may be formally postulated as follows.

Proposition 2.2. *The BBM equals the degree centrality for $\beta := 0$ i.e.,*

$$(c(1,0))_i = \deg v_i.$$

Proof. Trivially, one has

$$c(1,0) = \sum_{k=0}^{+\infty} 0^k \mathbf{A}^{k+1}\mathbb{1} = \mathbf{A}\mathbb{1}.$$

■

We will now investigate some properties of the BBM and firstly look to the following lemma and propositions.

Lemma 2.1. *If the series*

$$\sum_{k=0}^{+\infty} a_k \beta^k, \quad a_k \in \mathbb{R}$$

converges in in the open ball $\mathbb{B}_0(r)$ which has radius $r \in \mathbb{R}^+$ then the sum $f(\beta) := \sum_{k=0}^{+\infty} a_k\beta^k$ is differentiable in $\mathbb{B}_0(r)$ and

$$\frac{\partial f}{\partial \beta} = \frac{\partial}{\partial \beta}\sum_{k=0}^{+\infty} a_k\beta^k = \sum_{k=0}^{+\infty}\frac{\partial}{\partial\beta}\left[a_k\beta^k\right] = \sum_{k=0}^{+\infty} k a_k \beta^{k-1} < +\infty.$$

Proof. See for instance (Hunter, 2014). ■

Proposition 2.3. *All entries in $(\mathrm{Id} - \beta\mathbf{A})^{-1}$ are continuous and differentiable on the closed ball $\mathbb{B}_\vartheta := \{\beta \in \mathbb{R} : |\beta| \leq \vartheta < 1/\hat{\lambda}\}$.*

Proof. As $(\mathrm{Id} - \beta\mathbf{A})^{-1}$ attains finite value for any $|\beta| < 1/\hat{\lambda}$, the series associated with each entry must converge i.e.,

$$\begin{aligned}((\mathrm{Id} - \beta\mathbf{A})^{-1})_{ij} &= \mathrm{Id}_{ij} + \beta\cdot(\mathbf{A})_{ij} + \beta^2\cdot(\mathbf{A}^2)_{ij} + \cdots \\ &= \sum_{k=0}^{+\infty}\beta^k\cdot(\mathbf{A}^k)_{ij} < +\infty.\end{aligned}$$

Each term is a continuous function of β and the series converges uniformly on $\mathbb{B}_\vartheta$. To see this, apply the Weierstraß M-test for $f_k(\beta) := \beta^k(\mathbf{A}^{k+1})_{ij}$,

$$\sup_{\beta\in\mathbb{B}_\vartheta} f_k(\beta) \leq \vartheta^k\left|(\mathbf{A}^{k+1})_{ij}\right| =: m_k.$$

As the ball $\mathbb{B}_\vartheta$ is completely covered by the open ball $\mathbb{B}_0(1/\hat{\lambda})$ the series corresponding $((\mathrm{Id}-\beta\mathbf{A})^{-1})_{ij}$ for $\beta \in \mathbb{B}_\vartheta$ converges. Furthermore, it converges absolutely on $\mathbb{B}_\vartheta$; see p.71 in (Hunter, 2014). One can therefore be sure that

$$\sum_k m_k < +\infty.$$

Uniform convergence may be concluded as the criteria of the M-test are satisfied. Thus, $((\mathrm{Id}-\beta\mathbf{A})^{-1})_{ij}$ is continuous; see Th.1.9, p.24 in (Andrews, 1998). Lemma 2.1 implies that it is differentiable i.e.,

$$\frac{\partial((\mathrm{Id}-\beta\mathbf{A})^{-1})_{ij}}{\partial\beta} = \frac{\partial}{\partial\beta}\sum_{k=0}^{+\infty}\beta^k(\mathbf{A}^k)_{ij} = \sum_{k=0}^{+\infty}k\beta^{k-1}(\mathbf{A}^k)_{ij} < +\infty.$$

■

An interesting property of the BBM is that it may be posed as an ordinary differential equation.

Proposition 2.4. *The BBM defined by equation* (2.7) *for* $|\beta| \leq \vartheta < 1/\hat{\lambda}$ *is a solution to the time-variant homogeneous system*

$$\frac{\partial c}{\partial\beta} = \widehat{\mathbf{A}}(\beta)c, \quad c(\beta_0) = c(1,\beta_0) \tag{2.12}$$

where $\widehat{\mathbf{A}}(\beta) := (\mathrm{Id}-\beta\mathbf{A})^{-1}\mathbf{A}$.

Proof. Differentiation with respect to β of the closed form of the BBM $c := c(1,\beta)$ yields

$$\begin{aligned}\frac{\partial c}{\partial\beta} = \frac{\partial(\mathrm{Id}-\beta\mathbf{A})^{-1}}{\partial\beta}\mathbf{A}\mathbb{1} &= -(\mathrm{Id}-\beta\mathbf{A})^{-1}\frac{\partial(\mathrm{Id}-\beta\mathbf{A})}{\partial\beta}(\mathrm{Id}-\beta\mathbf{A})^{-1}\mathbf{A}\mathbb{1}\\ &= (\mathrm{Id}-\beta\mathbf{A})^{-1}\mathbf{A}(\mathrm{Id}-\beta\mathbf{A})^{-1}\mathbf{A}\mathbb{1}\\ &= (\mathrm{Id}-\beta\mathbf{A})^{-1}\mathbf{A}\,c(1,\beta)\\ &= \widehat{\mathbf{A}}(\beta)c\end{aligned}$$

■

Remark. The differentiation with respect to β is performed entry-wise in $(\mathrm{Id}-\beta\mathbf{A})^{-1}$ and is justified by Proposition 2.3. It shall furthermore be noted that each entry of the derivative attains finite value.

The entries of $\widehat{\mathbf{A}}(\cdot)$ are shown to be continuous on the ball $\mathbb{B}_\vartheta$ according to the proof of Proposition 2.3, on which the system equation (2.12) therefore attains a unique solution, to

be specific the BBM; to justify this assertion, it shall be noted that $\widehat{\mathbf{A}}(\cdot)$ is locally integrable on any open ball centered at the origin $\mathbb{B}_\beta \subset (-1/\hat{\lambda}, 1/\hat{\lambda}) \subseteq \mathbb{R}$,

$$\int_{\mathbb{B}_\beta} \|\widehat{\mathbf{A}}(\xi)\|^* \, d\xi \leq \|\mathbf{A}\|^* \int_{\mathbb{B}_\beta} \left\| \sum_{k=0}^{+\infty} (\xi \mathbf{A})^k \right\|^* d\xi \leq \|\mathbf{A}\|^* \int_{\mathbb{B}_\beta} \sum_{k=0}^{+\infty} \|(\xi \mathbf{A})^k\|^* \, d\xi < +\infty$$

as follows analogously with the derivation presented in the proof of Proposition 2.8. Hence, it follows that there is a unique solution to equation (2.12); see Th.6.4.1, p.113 in (Lukes, 1982). The instantaneous rate of change of centrality of any node is therefore determined by the linear combination of the centrality scores of *all* nodes at given "time" β, as clearly follows algebraically from the expression $\partial c / \partial \beta = \widehat{\mathbf{A}} c$.

Proposition 2.5. *The matrix* $\widehat{\mathbf{A}}(\beta) := (\mathrm{Id} - \beta \mathbf{A})^{-1} \mathbf{A}$ *is symmetric for all* $|\beta| < 1/\hat{\lambda}$.

Proof. Trivially,

$$\left[(\mathrm{Id} - \beta \mathbf{A})^{-1} \mathbf{A} \right]^{\mathsf{T}} = \mathbf{A} (\mathrm{Id} - \beta \mathbf{A})^{-1},$$

as the adjacency matrix $\mathbf{A}$ corresponding an undirected, connected graph is symmetric by construction; recall Definition 2.4. As $|\beta| < 1/\hat{\lambda}$ the proof may be concluded as the Neumann series allows for

$$\mathbf{A} (\mathrm{Id} - \beta \mathbf{A})^{-1} = \mathbf{A} \sum_{k=0}^{+\infty} \beta^k \mathbf{A}^k = \sum_{k=0}^{+\infty} \beta^k \mathbf{A}^{k+1} = (\mathrm{Id} - \beta \mathbf{A})^{-1} \mathbf{A}.$$

■

As $\widehat{\mathbf{A}}(\cdot)$ is shown to be symmetric for all $|\beta| < 1/\hat{\lambda}$, its quadratic form can be used to show boundedness of the derivative of $\|c(1, \beta)\|_2$ and $\|c(1, \beta)\|_2^2$ where $\| \cdot \|_2$ is the ℓ^2 norm.

Proposition 2.6. *Given* $\mathbb{B}_\vartheta := \{\beta \in \mathbb{R} : |\beta| \leq \vartheta < 1/\hat{\lambda}\}$ *and* $c := c(1, \beta)$ *for* $\beta \in \mathbb{B}_\vartheta$,

$$\inf_{\widetilde{\beta} \in \mathbb{B}_\vartheta} \lambda(\widehat{\mathbf{A}}(\widetilde{\beta})) \|c\|_2^2 \leq \frac{1}{2} \frac{\partial \|c\|_2^2}{\partial \beta} \leq \sup_{\widetilde{\beta} \in \mathbb{B}_\vartheta} \lambda(\widehat{\mathbf{A}}(\widetilde{\beta})) \|c\|_2^2. \tag{2.13}$$

Proof. Clearly, for the standard inner product $\langle \cdot \rangle$,

$$\frac{1}{2} \frac{\partial \|c\|_2^2}{\partial \beta} = \frac{1}{2} \frac{\partial}{\partial \beta} [c \cdot c] = \frac{1}{2} \left[c \cdot \frac{\partial c}{\partial \beta} + \frac{\partial c}{\partial \beta} \cdot c \right] = \left\langle c, \frac{\partial c}{\partial \beta} \right\rangle = \left\langle c, \widehat{\mathbf{A}}(\beta) c \right\rangle,$$

as $\widehat{\mathbf{A}}(\beta)$ is real and symmetric; see Proposition 2.5. Accordingly, it can be concluded that

$$\frac{1}{2} \frac{\partial \|c\|_2^2}{\partial \beta} = c^{\mathsf{T}} \widehat{\mathbf{A}}(\beta) c.$$

Moreover, $\widehat{\mathbf{A}}(\cdot)$ can be diagonalized as

$$\widehat{\mathbf{A}}(\beta) := \mathbf{Q}(\beta)^{\mathsf{T}}\Lambda(\beta)\mathbf{Q}(\beta),$$

where $\mathbf{Q}(\cdot)$ is an orthogonal matrix and $\Lambda(\cdot) = \text{diag}(\lambda_1, \lambda_2, ..., \lambda_{|V|})$ is constituted by the eigenvalues of $\widehat{\mathbf{A}}(\cdot)$. It then follows that

$$\begin{aligned} c^{\mathsf{T}}\widehat{\mathbf{A}}(\beta)c &= c^{\mathsf{T}}\mathbf{Q}(\beta)^{\mathsf{T}}\Lambda(\beta)\mathbf{Q}(\beta)c \\ &= (\mathbf{Q}(\beta)c)^{\mathsf{T}}\Lambda(\beta)\mathbf{Q}(\beta)c \\ &= v^{\mathsf{T}}\Lambda(\beta)v, \end{aligned}$$

using the assertion $v = v(\beta) := \mathbf{Q}(\beta)c$. It shall in the same sentence be remarked that

$$v^{\mathsf{T}}v = \sum_{k=1}^{|V|} v_k^2 = [\mathbf{Q}(\beta)c]^{\mathsf{T}}\mathbf{Q}(\beta)c = \|c\|_2^2. \tag{2.14}$$

The upper bound of equation (2.13) may so be attained in accordance with equation (2.14),

$$c^{\mathsf{T}}\widehat{\mathbf{A}}(\beta)c = \sum_{k=1}^{|V|} \lambda_k v_k^2 \leq \sup_{\widetilde{\beta}\in\mathbb{B}_\vartheta} \lambda(\widehat{\mathbf{A}}(\widetilde{\beta})) \sum_{k=1}^{|V|} v_k^2 = \sup_{\widetilde{\beta}\in\mathbb{B}_\vartheta} \lambda(\widehat{\mathbf{A}}(\widetilde{\beta}))\|c\|_2^2.$$

Analogously for the lower bound,

$$c^{\mathsf{T}}\widehat{\mathbf{A}}(\beta)c = \sum_{k=1}^{|V|} \lambda_k v_k^2 \geq \inf_{\widetilde{\beta}\in\mathbb{B}_\vartheta} \lambda(\widehat{\mathbf{A}}(\widetilde{\beta})) \sum_{k=1}^{|V|} v_k^2 = \inf_{\widetilde{\beta}\in\mathbb{B}_\vartheta} \lambda(\widehat{\mathbf{A}}(\widetilde{\beta}))\|c\|_2^2.$$

■

Remark. For the quantity $\|c\|_2$, its derivative may also be shown to be bounded:

$$\frac{\partial\|c\|_2}{\partial\beta} = \frac{\partial\sqrt{c\cdot c}}{\partial\beta} = \frac{1}{2\|c\|_2}\frac{\partial}{\partial\beta}[c\cdot c] = \frac{1}{\|c\|_2}c^{\mathsf{T}}\widehat{\mathbf{A}}(\beta)c, \quad c \neq \mathbf{0}. \tag{2.15}$$

As shown in the proof of Proposition 2.6, the quadratic form has the bounds

$$\inf_{\widetilde{\beta}\in\mathbb{B}_\vartheta} \lambda(\widehat{\mathbf{A}}(\widetilde{\beta}))\|c\|_2^2 \leq c^{\mathsf{T}}\widehat{\mathbf{A}}(\beta)c \leq \sup_{\widetilde{\beta}\in\mathbb{B}_\vartheta} \lambda(\widehat{\mathbf{A}}(\widetilde{\beta}))\|c\|_2^2$$

and it is implied from equation (2.13) and equation (2.15) therefore, that

$$\inf_{\widetilde{\beta}\in\mathbb{B}_\vartheta} \lambda(\widehat{\mathbf{A}}(\widetilde{\beta}))\|c\|_2 \leq \frac{\partial\|c\|_2}{\partial\beta} \leq \sup_{\widetilde{\beta}\in\mathbb{B}_\vartheta} \lambda(\widehat{\mathbf{A}}(\widetilde{\beta}))\|c\|_2.$$

Lastly for this section, it shall be proven that the quantity $\|c(1,\beta)\|_1$ for any $|\beta| < 1/\hat{\lambda}$ has a tight upper bound.

Proposition 2.7. *For any* $|\beta| < 1/\hat{\lambda}$,

$$\|c(1,\beta)\|_1 \leq \frac{\hat{\lambda}}{1-|\beta|\hat{\lambda}}|V|,$$

where $|V|$ *is the number of nodes in the underlying graph and* $\|\cdot\|_1$ *is the* ℓ^1 *norm.*

Proof. Firstly,

$$\begin{aligned}\|c(1,\beta)\|_1 &= \|(\mathrm{Id}-\beta\mathbf{A})^{-1}\mathbf{A}\mathbb{1}\|_1 \leq \|(\mathrm{Id}-\beta\mathbf{A})^{-1}\|^*\|\mathbf{A}\|^*\|\mathbb{1}\|_1 \\ &= \hat{\lambda}|V|\max_i \sqrt{\lambda_i\left((\mathrm{Id}-\beta\mathbf{A})^{-1}\right)^2} \\ &= \hat{\lambda}|V|\max_i \sqrt{\lambda_i^2\left((\mathrm{Id}-\beta\mathbf{A})^{-1}\right)},\end{aligned}$$

recalling that $\mathrm{Id}-\beta\mathbf{A}$ is symmetric. According to Lemma 2.2, $\mathrm{Id}-\beta\mathbf{A}$ is PD if $|\beta| < 1/\hat{\lambda}$. Since $(\mathrm{Id}-\beta\mathbf{A})^{-1}$ likewise is PD, and all eigenvalues indeed are strictly positive for such matrices, one has

$$\|c(1,\beta)\|_1 \leq \hat{\lambda}|V|\max_i \lambda_i\left((\mathrm{Id}-\beta\mathbf{A})^{-1}\right) = \frac{\hat{\lambda}}{\min_i \lambda_i(\mathrm{Id}-\beta\mathbf{A})}|V|.$$

As shown in the proof of Lemma 2.2, it follows that $\min_i \lambda_i(\mathrm{Id}-\beta\mathbf{A}) \geq 1-|\beta|\hat{\lambda} > 0$ which concludes this proof;

$$\|c(1,\beta)\|_1 \leq \frac{\hat{\lambda}}{1-|\beta|\hat{\lambda}}|V|.$$

■

Remark. If $\beta = 0$, then $c(1,0)$ equals the degree centrality, as proven by Proposition 2.2. Since the underlying graph is connected, each node degree must be strictly positive i.e., $\deg v_i > 0$ and it shall be noticed that

$$\|c(1,0)\|_1 = \sum_{i=1}^{|V|}|c(1,0)_i| = \sum_{v\in V}|\deg v| = \sum_{v\in V}\deg v \leq \hat{\lambda}|V|,$$

i.e., it is implied that the average degree is less than or equal to the largest eigenvalue,

$$\frac{1}{|V|}\sum_{v\in V}\deg v = \deg_{\text{avg}} \leq \hat{\lambda},$$

which indeed is a familiar result.

2.5 The Mean of Bonacich β–Centrality

The following section proposes the derivation of the mean of the Bonacich β-centrality measure (abbreviated MBC). Let firstly $\gamma \subset (-1/\hat{\lambda}, 1/\hat{\lambda})$ be any symmetric interval for which any inner point $\beta \in \gamma$ yields convergence of equation (2.7). The MBC shall then defined as

$$\bar{c}(\gamma) := \frac{1}{L(\gamma)} \int_{\gamma} c(1, \beta)\, d\beta, \tag{2.16}$$

where $L(\gamma)$ denotes the length of the interval γ and $c(1, \beta)$ is defined by equation (2.7). It follows algebraically that equation (2.16) is a $|V| \times 1$ column vector; each entry corresponds to a node i.e., the i:th row of $\bar{c}$ corresponds to $v_i \in V$. The performance of this quantity and how it relates to the BBM is presented in Chapter 5. A first generic closed-form expression of equation (2.16) is derived as follows.

Proposition 2.8. *The MBC as defined by equation* (2.16) *has for every* $0 < r < 1/\hat{\lambda}$ *and symmetric interval* $\gamma = (-r,\ r)$ *the closed form*

$$\bar{c}(\gamma) = \frac{1}{2r} \left[\log\left(\mathrm{Id} + r\mathbf{A}\right) - \log\left(\mathrm{Id} - r\mathbf{A}\right)\right] \mathbb{1}, \tag{2.17}$$

for the principal matrix logarithm.

In order to prove this proposition, the following Lemma is used.

Lemma 2.2. *The matrices* $\mathrm{Id} \pm r\mathbf{A}$ *are positive definite if* $|r| < 1/\hat{\lambda}$.

Proof. Firstly, for any $i \in \{1, 2, ..., |V|\}$,

$$\lambda_i(\mathrm{Id} \pm r\mathbf{A}) = 1 \pm r\lambda_i \geq 1 - |r|\hat{\lambda} > 0. \tag{2.18}$$

The justification of equation (2.18) is that Id has $|V|$ eigenvalues which all equal 1. The eigenvalues of $\mathrm{Id} \pm r\mathbf{A}$ are therefore, up to the sign, the same irrespective of the eigenvalue correspondences. In other words, the eigenvalues of $\mathrm{Id} \pm r\mathbf{A}$ are the same as $\pm r\mathbf{A}$ but shifted by 1. Conclusively, $\mathrm{Id} \pm r\mathbf{A}$ are real and symmetric matrices which have $|V|$ number of strictly positive eigenvalues and may therefore, in accordance with the spectral theorem, be decomposed as

$$\mathrm{Id} \pm r\mathbf{A} = \mathbf{Q}^{\mathsf{T}} \Lambda \mathbf{Q}$$

for some orthogonal matrix $\mathbf{Q}$ and diagonal matrix $\Lambda = \mathrm{diag}(\lambda_1, ..., \lambda_{|V|})$ which contains the eigenvalues of $\mathrm{Id} \pm r\mathbf{A}$. To conclude positive-definiteness (PD) and thus the proof, consider firstly

$$\mathbf{u}^{\mathsf{T}}(\mathrm{Id} \pm r\mathbf{A})\mathbf{u} = \mathbf{u}^{\mathsf{T}}\mathbf{Q}^{\mathsf{T}}\Lambda\mathbf{Q}\mathbf{u} = (\mathbf{Q}\mathbf{u})^{\mathsf{T}}\Lambda(\mathbf{Q}\mathbf{u}).$$

Let now $\mathbf{y} := \mathbf{Q}\mathbf{u} = (y_1, ..., y_{|V|})^{\mathsf{T}} \neq \mathbf{0} \in \mathbb{R}^{|V|}$, which implies

$$\mathbf{y}^{\mathsf{T}}\Lambda\mathbf{y} = \sum_{i=1}^{|V|} \lambda_i y_i^2 > 0$$

as $\lambda_i > 0$ and $y_i^2 > 0$ for at least one i. ∎

The proof of Proposition 2.8 can thus be carried through.

Proof of Proposition 2.8. Firstly, we derive

$$\begin{aligned}\bar{c}(\gamma) = \frac{1}{2r}\int_{-r}^{r}\sum_{k=0}^{+\infty}\beta^k\mathbf{A}^{k+1}\mathbb{1}\,d\beta &= \frac{1}{2r}\sum_{k=0}^{+\infty}\left\{\int_{-r}^{r}\beta^k\,d\beta\right\}\mathbf{A}^{k+1}\mathbb{1}\\ &= -\frac{1}{2r}\sum_{k=1}^{+\infty}\frac{1}{k}\left(-r\mathbf{A}\right)^k\mathbb{1} + \frac{1}{2r}\sum_{k=1}^{+\infty}\frac{1}{k}\left(r\mathbf{A}\right)^k\mathbb{1}\\ &= \frac{1}{2r}\left[\log\left(\mathrm{Id}+r\mathbf{A}\right) - \log\left(\mathrm{Id}-r\mathbf{A}\right)\right]\mathbb{1}.\end{aligned}$$

The interchange of the series and integral is justified as follows. Clearly,

$$\int_\gamma\sum_k\beta^k\mathbf{A}^{k+1}\mathbb{1}\,d\beta = \left\{\int_\gamma\sum_k\beta^k\mathbf{A}^k d\beta\right\}\mathbf{A}\mathbb{1}. \tag{2.19}$$

Now, treating the bracket in equation (2.19) one obtains

$$\int_\gamma\sum_k\|\beta^k\mathbf{A}^k\|^*\,d\beta = \int_\gamma\sum_k|\beta|^k\|\mathbf{A}^k\|^*\,d\beta, \tag{2.20}$$

where $\|\cdot\|^*$ denotes the operator norm. Given $\|\mathbf{A}\|^* := \max_i\sqrt{\lambda_i(\mathbf{A}^\mathsf{T}\mathbf{A})}$ and baring in mind that $\mathbf{A}$ is symmetric, for the power $(\mathbf{A}^k)^\mathsf{T} = \mathbf{A}^k$ one has

$$\|\mathbf{A}^k\|^* = \max_i\sqrt{\lambda_i\{(\mathbf{A}^k)^\mathsf{T}\mathbf{A}^k\}} = \max_i\sqrt{\lambda_i(\mathbf{A}^{2k})} = \max_i\sqrt{\lambda_i^{2k}(\mathbf{A})},$$

which implies $\|\mathbf{A}^k\|^* = \max_i|\lambda_i^k(\mathbf{A})| = \hat{\lambda}^k$. In terms of equation equation (2.20) this yields

$$\int_\gamma\sum_k|\beta|^k\|\mathbf{A}^k\|^*\,d\beta = \int_\gamma\sum_k|\beta\hat{\lambda}|^k d\beta. \tag{2.21}$$

Since $|\beta| < 1/\hat{\lambda}$ for any $\beta\in\gamma$, the root test implies that the integrand of equation (2.21) is absolute convergent,

$$\limsup_{k\to+\infty}\sqrt[k]{|\beta\hat{\lambda}|^k} = \limsup_{k\to+\infty}|\beta|\hat{\lambda} < 1.$$

For any $k\geq 0$ the inequality $|\beta\hat{\lambda}|^k \leq (\hat{\lambda}r)^k$ holds. This implies

$$\sum_k|\beta\hat{\lambda}|^k \leq \sum_k(\hat{\lambda}r)^k = \frac{1}{1-\hat{\lambda}r},$$

and equation (2.21) by that means satisfies the inequality:

$$\int_\gamma\sum_k|\beta\hat{\lambda}|^k d\beta \leq \frac{1}{1-\hat{\lambda}r}\int_\gamma d\beta = \frac{2r}{1-\hat{\lambda}r} < +\infty.$$

The legibility of the interchange then follows by Fubuni's theorem. According to Lemma 2.2, the matrices $\mathrm{Id} \pm r\mathbf{A}$ are PD and therefore the logarithm terms in equation (2.17) are uniquely determined; see Th.1.31, p.20 in (Higham, 2008). ■

The closed form in equation (2.17) of the MBC may be further simplified to achieve a more compact, one–term identity according to the following proposition.

Proposition 2.9. *The MBC defined by equation* (2.16) *has for any* $0 < r < 1/\hat{\lambda}$ *the closed form*

$$\bar{c}(\gamma) = \frac{1}{2r} \log \left[2(\mathrm{Id} - r\mathbf{A})^{-1} - \mathrm{Id} \right] \mathbb{1} \tag{2.22}$$

for the principal matrix logarithm and $\gamma = (-r, r)$.

The proof of this proposition relies on the following three lemmas.

Lemma 2.3. *If* $|r| < 1/\hat{\lambda}$, *then*

$$r\mathbf{A}(\mathrm{Id} - r\mathbf{A})^{-1} = (\mathrm{Id} - r\mathbf{A})^{-1}(r\mathbf{A}).$$

Proof. The proof is trivial but is included for illustrative purposes. As recalled by Lemma 2.2, it is clear that $|r| < 1/\hat{\lambda}$ yields positive-definiteness of $\mathrm{Id} - r\mathbf{A}$ and the spectral radius of $r\mathbf{A}$ is contained within the unit circle. Thus, it can be ascertained that

$$r\mathbf{A}(\mathrm{Id} - r\mathbf{A})^{-1} = r\mathbf{A} \left[\sum_k (r\mathbf{A})^k \right] = (\mathrm{Id} - r\mathbf{A})^{-1}(r\mathbf{A}).$$

■

Lemma 2.4. *If* $\lambda(\mathbf{A}),\ \lambda(\mathbf{B}) \in \mathbb{R}^+$ *and* $\mathbf{AB} = \mathbf{BA}$, *then*

$$\log(\mathbf{AB}^{\pm 1}) = \log \mathbf{A} \pm \log \mathbf{B} - 2\pi i\, \mathcal{U}[\log \mathbf{A} \pm \log \mathbf{B}],$$

where $\mathcal{U} : \mathbb{M} \to \mathbb{M}$ *is the matrix unwinding function; see (Aprahamian & Higham, 2014). If* $\arg \lambda_i(\mathbf{A}) + \arg \lambda_j(\mathbf{B}) \in (-\pi,\ \pi]$ *for all corresponding eigenvalues of* $\mathbf{A}$ *and* $\mathbf{B}$, *then*

$$\log(\mathbf{AB}^{\pm 1}) = \log \mathbf{A} \pm \log \mathbf{B}.$$

See (Higham, 2008).

Lemma 2.5. *For any non-singular matrix* $\mathbf{A} \in \mathbb{C}^{m \times m}$,

$$\log \mathbf{A}^k = k \log \mathbf{A},$$

for any $k \in (-1,\ 1]$ *and also for* $k = -1$ *if* $\lambda(\mathbf{A}) \in \mathbb{R}_{\geq 0}$ *(Higham, 2008).*

The proof of Proposition 2.9 which regards the closed form of equation (2.16) is carried out as follows.

Proof of Proposition 2.9. Firstly, it shall be noted under Lemma 2.5 that the following equality holds, as it was proven in Lemma 2.2 that $\mathrm{Id} \pm r\mathbf{A}$ are PD:

$$-\log(\mathrm{Id} - r\mathbf{A}) = \log(\mathrm{Id} - r\mathbf{A})^{-1},$$

This implies that equation (2.17) is equal to

$$\frac{1}{2r}\left[\log(\mathrm{Id} + r\mathbf{A}) + \log(\mathrm{Id} - r\mathbf{A})^{-1}\right]\mathbb{1}. \tag{2.23}$$

Furthermore, as the inverse of a PD matrix likewise is PD and taking into consideration that the eigenvalues are real, it follows for any index tuple (i, j) that concerns the eigenvalue correspondences,

$$\arg\lambda_i(\mathrm{Id} + r\mathbf{A}) + \arg\lambda_j(\mathrm{Id} - r\mathbf{A})^{-1} = 0.$$

In accordance with Lemma 2.4, the logarithm sum in equation (2.23) may be simplified to a single logarithm as the matrices are commuting. To prove this assertion, firstly note that

$$r\mathbf{A}(\mathrm{Id} - r\mathbf{A})^{-1} = (\mathrm{Id} - r\mathbf{A})^{-1}(r\mathbf{A}). \tag{2.24}$$

as it was proven by Lemma 2.3 that equation (2.24) holds under given premise $|r| < 1/\hat{\lambda}$. Thus,

$$\left[\log(\mathrm{Id} + r\mathbf{A}) + \log(\mathrm{Id} - r\mathbf{A})^{-1}\right]\mathbb{1} = \log\left[(\mathrm{Id} + r\mathbf{A})(\mathrm{Id} - r\mathbf{A})^{-1}\right]\mathbb{1}.$$

The proof is concluded,

$$\begin{aligned}
(\mathrm{Id} + r\mathbf{A})(\mathrm{Id} - r\mathbf{A})^{-1} &= (\mathrm{Id} - r\mathbf{A})^{-1} + r\mathbf{A}(\mathrm{Id} - r\mathbf{A})^{-1} \\
&= \sum_{k=0}^{+\infty}(r\mathbf{A})^k + r\mathbf{A}\sum_{k=0}^{+\infty}(r\mathbf{A})^k \\
&= \sum_{k=0}^{+\infty}(r\mathbf{A})^k + \sum_{k=1}^{+\infty}(r\mathbf{A})^k \\
&= 2\sum_{k=0}^{+\infty}(r\mathbf{A})^k - \mathrm{Id} \\
&= 2(\mathrm{Id} - r\mathbf{A})^{-1} - \mathrm{Id}.
\end{aligned}$$

■

Lastly, for the MBC, the closed form of equation (2.17) may also be expressed according to the decomposition equation (2.25).

Proposition 2.10. *The MBC defined by equation* (2.16) *has for any* $0 < r < 1/\hat{\lambda}$ *the closed form*

$$\bar{c}(\gamma) = \frac{1}{2r}\mathbf{Q}\mathbf{X}\mathbf{Q}^{\mathsf{T}}\mathbb{1} \tag{2.25}$$

where $\mathbf{X} \in \mathbb{R}^{|V|\times|V|}$ *is a diagonal matrix for which*

$$(\mathbf{X})_{ii} = \ln\left(\frac{1+r\lambda_i}{1-r\lambda_i}\right),$$

and $\mathbf{Q}$ *is an orthogonal matrix,* $\mathbf{A} = \mathbf{Q}\Lambda\mathbf{Q}^{\mathsf{T}}$ *where* $\Lambda = \mathrm{diag}(\lambda_1, ..., \lambda_{|V|})$ *contains the eigenvalues of* $\mathbf{A}$ *and* $\gamma = (-r, r)$.

Proof. It shall firstly be noted that because $\mathrm{Id} + r\mathbf{A}$ is PD, as concluded by Lemma 2.2, it is diagonalizable as

$$\mathrm{Id} + r\mathbf{A} = \mathbf{Q}\Lambda_{+}\mathbf{Q}^{\mathsf{T}} = \mathbf{Q}(\mathrm{Id} + r\Lambda)\mathbf{Q}^{\mathsf{T}} = \mathrm{Id} + r\mathbf{Q}\Lambda\mathbf{Q}^{\mathsf{T}},$$

where Λ is a diagonal matrix containing the eigenvalues of $\mathbf{A}$. It is therefore implied that $\mathbf{A} = \mathbf{Q}\Lambda\mathbf{Q}^{\mathsf{T}}$. For the negative sign, using this fact yields

$$\mathrm{Id} - r\mathbf{A} = \mathrm{Id} - r\mathbf{Q}\Lambda\mathbf{Q}^{\mathsf{T}} = \mathbf{Q}\mathbf{Q}^{\mathsf{T}} - r\mathbf{Q}\Lambda\mathbf{Q}^{\mathsf{T}} = \mathbf{Q}(\mathrm{Id} - r\Lambda)\mathbf{Q}^{\mathsf{T}} = \mathbf{Q}\Lambda_{-}\mathbf{Q}^{\mathsf{T}}.$$

The principal matrix logarithm is analytic for the given bounds of r, hence each term in equation (2.17) may be expressed accordingly,

$$\log(\mathrm{Id} \pm r\mathbf{A}) = \mathbf{Q}\log(\Lambda_{\pm})\mathbf{Q}^{\mathsf{T}} = \mathbf{Q}\log(\mathrm{Id} \pm r\Lambda)\mathbf{Q}^{\mathsf{T}}$$

which leads to the equality:

$$\log(\mathrm{Id} + r\mathbf{A}) - \log(\mathrm{Id} - r\mathbf{A}) = \mathbf{Q}[\log(\mathrm{Id} + r\Lambda) - \log(\mathrm{Id} - r\Lambda)]\mathbf{Q}^{\mathsf{T}}.$$

Since $\mathrm{Id} \pm r\Lambda$ are diagonal matrices, one has $(\log(\mathrm{Id} \pm r\Lambda))_{ii} = \ln(1 \pm r\lambda_i)$ which for $\mathbf{X} := \log(\mathrm{Id} + r\Lambda) - \log(\mathrm{Id} - r\Lambda)$ yields

$$(\mathbf{X})_{ii} = \ln(1 + r\lambda_i) - \ln(1 - r\lambda_i) = \ln\left(\frac{1+r\lambda_i}{1-r\lambda_i}\right), \quad (\mathbf{X})_{ij} = 0 \ \forall i \neq j.$$

■

2.6 Barabási–Albert Model

The Barabási–Albert Model (BAM) is a graph generative model which may be used to study the growth and evolution of complex networks. It enables constructions of synthetic graphs that resemble real-life graphs such as e.g., social networks or the citation networks. In this section, it is assumed that no nodes are leaving the graphs – only entering them. Furthermore, if a node enters a graph, it must induce at least one edge to a node that is within the graph.

In the Barabási–Albert Model (BAM), the graph expands as *new* nodes are entering it. At each time a node is entering a graph, it will "prefer" to induce edges with nodes already within the graph which have high connectivity. In other words, any entering node have a preference for nodes with high degree, and this is referred as the preferential attachment principle of the model. It is common that complex systems, such as social networks or e.g., economic networks exhibit this mechanism of growth and preferential attachment (Barabási & Albert, 1999). The preferential rule is mathematically defined as

$$p_i = \mathbb{P}\mathrm{r}(\text{entering node connects to } v_i) := \frac{\deg v_i}{2|E|} \tag{2.26}$$

meaning that the probability p_i of an entering node is inducing an edge to a pre-existing node v_i is proportional to the degree of v_i (Albert & Barabási, 2002). The attachment model is stochastic in the sense that any entering node might induce an edge to any node in the graph. The edge set cardinality $|E|$ does not include the edges that will be connected to the entering node.

In algorithmic point of view, the input, from which the graph grows in accordance with the preferential attachment model, is a small connected graph $G_0 := G(t = 0)$ constituted by $\ell_0 > 0$ nodes. At each time instant, a new node is connected to $\ell \leq \ell_0$ distinct nodes in the graph, according to the probability equation (2.26). The procedure terminates when the total number of nodes reaches some specified amount $n \geq \ell_0$. Clearly, the number of nodes after discrete time $t \in \mathbb{N}_0$ follows the formula

$$n(t) = \ell_0 + t, \quad n(0) = \ell_0,$$

because at each time instant a new node enters the graph, which increments the node set cardinality by 1. Similarly, the number of edges after time t can be calculated as $|E(t)| = E_0 + \ell t$, where $E_0 := |E(0)|$ is the number of edges in the initial graph G_0. Figure 2.5 shows a BAM-generated graph consisting $n = |V| = 20$ nodes, generated by $\ell = 1$ and $G_0 := \mathcal{S}_1$. In essence, the initial graph is a small star and at each time instant, just a single "new" node is entering the network.

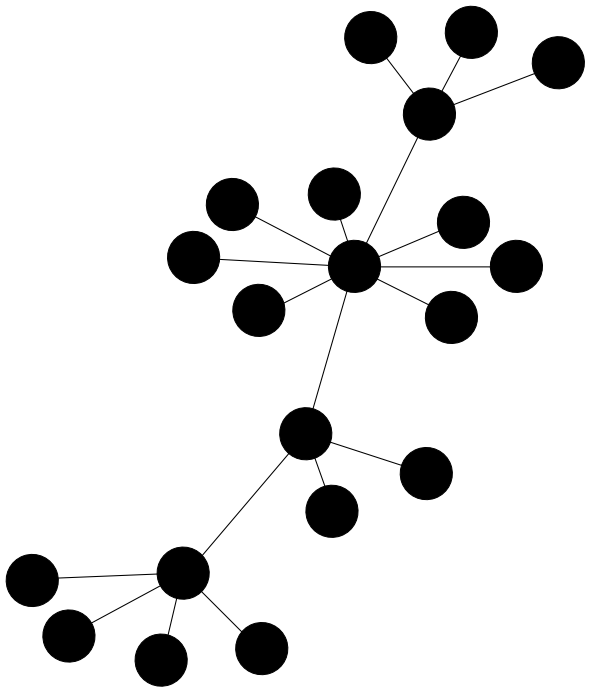

Figure 2.5: A generated Barabási–Albert graph which has $|V| = 20$ nodes; the preferential attachment is $\ell = 1$ and the initial graph is $\mathcal{S}_1$.

2.6.1 Degree Dynamics

The most distinctive property of a Barabási-Albert network, given large enough population size and stabilization, is its scale-free degree distribution i.e., the degree distribution is a power-law in accordance with

$$P(k) \sim k^{-\gamma} \tag{2.27}$$

where k denotes the degree, $P(k)$ is the probability that a randomly selected node has a connection to k other nodes i.e., that its degree is k. Many real life networks possess this property (Barabási & Albert, 1999). Consequently, given a largely populated graph, it is expected that some nodes have high degrees, whereas the vast majority has significantly lesser connectivity. To conclude, the BAM generates so called random scale-free networks (SFN), which have the aforementioned properties and degree distribution according to equation (2.27) for $\gamma \approx 3$.

Chapter 3

Markov Processes

This chapter provides a brief outline of Markov processes and related theory which is incorporated in the information propagation model. Major parts of this chapter is based of Richard Serfozo's book "Basics of Applied Stochastic Processes" (Springer); see (Serfozo, 2009).

Markov chains are stochastic processes with a wide variety of applications. Any dynamical system whose states satisfy the recursion relation

$$X_n = f(X_{n-1}, y_n), \quad n \geq 1, \tag{3.1}$$

can be represented by a Markov chain; here f is a non-random, deterministic function and $y_1, y_2...$ are independent variables which are identically distributed. Any state X_n by equation (3.1) is determined by its previous state X_{n-1} and some exogenous, random variable.

A stochastic process in discrete time $X := \{X_n \,|\, n \geq 0\}$ on a finite state set Γ is defined as a collection of random variables on $(\Omega, \mathcal{F}, \mathbb{P})$. Any $X_n \in \Gamma$ is referred to as the state of M at time n. The finite-dimensional distribution of M is stated formally as

$$\mathbb{P}\mathrm{r}(X_0 = s_0, ..., X_n = s_n), \quad s_0, ..., s_n \in \Gamma \ \forall n \in \mathbb{N}_0.$$

These are the preliminaries and the definition of a Markov chain (MC) thus follows.

Definition 3.1. (Markov chains) *A time-homogeneous Markov chain is a stochastic process in discrete time $X := \{X_i \,|\, n \geq 0\}$ on a finite state space Γ which for any $(s_i, s_j) \in \Gamma \times \Gamma$ satisfies,*

i) $\mathbb{P}\mathrm{r}(X_{n+1} = s_j \,|\, X_0, ..., X_n) = \mathbb{P}\mathrm{r}(X_{n+1} = s_j \,|\, X_n)$
ii) $p_{ij} := \mathbb{P}\mathrm{r}(X_{n+1} = s_j \,|\, X_n = s_i)$.

Here, i) refers to the Markov property i.e., the future state of the process depends only on its current state. Moreover, ii) is the transition probability, that is the probability that the Markov chain transitions from state s_i to s_j. The matrix $\mathsf{P} := (p_{ij})$ is the transition matrix of the Markov chain.

3.1 Classification of States

Naturally, a Markov chain (from here, abbreviated MC) might return to some state after some time; if starting in s_i, then the sequence of states may be on the form

$$s_i, s_j, s_k, ..., s_l, s_i, ...$$

However, a definite return to a state is not generally true, it depends on the transition probabilities which define the chain. If for some state s_m the only transition probability is $p_{mm} = 1$, it follows that for any $m \neq j$,

$$p_{mj} = 0.$$

Consequently, once the MC visits the state s_m, it can never leave it. In such case, the sequence of states will appear as

$$s_i, s_j, s_k, ..., s_l, s_m, s_m, s_m, ...$$

Even over indefinite time, a state may be visited a finite amount of times.

Let $\tau_j := \min\{n \geq 1 : X_n = s_j \in \Gamma\}$ i.e., the time until the MC reaches state s_j for the first time since its birth. Furthermore, for $s_i, s_j \in \Gamma$. define $f_{ij}^1 := p_{ij}$ and

$$f_{ij}^n := \Pr(\tau_j = n \,|\, X_0 = s_i) = \sum_{k \neq j} p_{ik} \, f_{kj}^{n-1}, \quad n \geq 2 \tag{3.2}$$

which is the probability that the MC reaches state s_j for the first time in $n \geq 1$ steps, given that it was initialized in state s_i. Particularly, it is of interest to conclude whether or not some state s_j finally will be reached from state s_i, formally

$$f_{ij} := \Pr(\tau_j < +\infty \,|\, X_0 = s_i) = \sum_{k=1}^{+\infty} f_{ij}^k.$$

Definition 3.2. *A state $s_i \in \Gamma$ is recurrent if $f_{ii} = 1$ i.e., if it is guaranteed to return. A state $s_j \in \Gamma$ is transient if $f_{ii} < 1$. Furthermore, a recurrent state is said to be positive recurrent if $\mathbb{E}[\tau_i \,|\, X_0 = s_i] < +\infty$ and otherwise, null recurrent.*

A state s_j is *accessible* from state s_i,

$$s_i \longrightarrow s_j,$$

if there exists some $n \geq 1$ for which $p_{ij}^n > 0$. In this context, the states s_i and s_j are said to communicate under the equivalence relation,

$$s_i \longleftrightarrow s_j,$$

if $s_i \longrightarrow s_j$ and $s_j \longrightarrow s_i$.

3.2 2-State Markov Chain

This thesis takes particular interest in 2-state Markov chains. Although they might perhaps be the most trivial MC:s there are, they have properties which are sought-after, to regulate the susceptibility of nodes in the information propagation; the reader is referred to Section 4.2 for further details about the incorporation of MC to the IPM. As there are only two states needed to be taken in consideration, if $p_{11} := a$ and $p_{22} := b$ for some $a, b \in [0, 1]$ then it must follow that

$$p_{12} := 1 - a, \;\; p_{21} := 1 - b$$

as these correspond to the complementary events. Clearly, the transition matrix then has the form

$$\mathsf{P} = \begin{bmatrix} a & 1-a \\ 1-b & b \end{bmatrix}.$$

If $a, b \neq 0$, then both states communicate and are recurrent. A discrete time-homogeneous MC constituted by only two states $\Gamma := \{s_1, s_2\}$ is diagrammatically represented in Figure 3.1; the direction of the arrow shows to which state is transitioned to. Using graph theory notions, each edge (i, j) is directed from state s_i to s_j (nodes) and the weight of the edge is the transition probability.

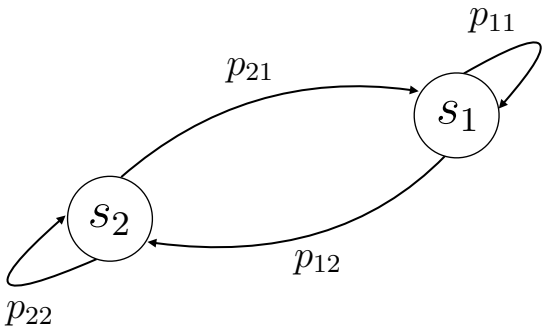

Figure 3.1: A diagrammatic view of the transition probabilities of a 2-state MC. It is worthwhile to compare this view with Figure 4.2, which depicts the inherent MC for each node, which regulates the node's susceptibility.

Chapter 4

Models

As stated, one of the aim objectives is to investigate how the centrality of the initiator node affects the information propagation globally over the graph. To facilitate the understanding of the importance of the initiator node, an information propagation model is devised (which will be referred to as IPM throughout the report) which emulates the spreading over a graph; the information propagation is within this model induced by a single node, which enables us to measure the impact of centrality. The time it takes for the information to reach the entire graph can be assessed against the centrality ranking of the single node from which the spreading is induced; see Chapter 5 for further details in this application.

4.1 Network Constructions

One of the research objectives is to attain understanding of how the theoretical centrality of the initiator node of the information spread affects the propagation globally. Ideally therefore, a graph generative model should cause some nodes to be (definitely) more central than others with respect to the metrics that are applied within this thesis. In similar regard, it is assumed that the centrality rankings correlate with the degrees of the nodes i.e., that each centrality metric yields similar rankings to those obtained from the degree centrality. The reason for this assumption is the relation which the BBM has with the eigenvector and degree centrality – either (usually) can be derived from the BBM by specification of the β. The same can of course be said about the MBC, as it is constructed from BBM, although it is not intrinsically regarded as a centrality measure.

The requirements of the graph generative model can thus be reduced to only generating graphs which have exponentially decreasing node distributions. It is reasonable to assume that most real social networks have exponentially decreasing node distributions, as it is of common conception that some individuals have bigger social circles than others. In a theoretic point of view, the nodes which represent the most "popular" individuals have the highest degrees in this analogy. These networks are scale-free and the IPM algorithm is therefore performed on graphs generated by the Barabási–Albert model. To allow for further comparison,

the IPM will also be performed on a real observed social media network by Stanford university. This is an undirected connected graph which is constituted by 4,039 nodes and 88,234 edges (Mcauley & Leskovec 2012; *SNAP: Network datasets: Social circles* n.d.).

4.2 Information Propagation Model (IPM)

The first step of the model is to calculate and assign the theoretical centrality score for each node. An *important* node is defined to be in the top 1% of the centrality rankings; conversely, an *unimportant* node is within the bottom 1%. The information propagation is initiated by a single node, which is referred to as the initiator node of the spread. Thereafter, the information propagates through the graph as the nodes are transmitting it to adjacent nodes. In simulations of the IPM, the initiator is either randomly selected from the group of important nodes or from the group of unimportant nodes – or either, meaning that the initiator is randomly selected from the set of nodes of the whole graph, which the model is performed on. The IPM runs in discrete time.

A node can either be *susceptible* or *insusceptible* of information. Throughout the lifetime of the model, any node alternates between these two conditions of susceptibility. This is independently regulated by an inherent 2-state MC assigned to each node; the state space is defined as

$$\Gamma := \{\textbf{susceptible}, \textbf{insusceptible}\}. \tag{4.1}$$

Naturally, this construction induces the property that any node can not be simultaneously in both states. Each node also has an attribute which tells whether or not it has received the information. The information defined in the IPM, in its abstraction, is immutable and can be cloned infinitely many times. Therefore, if a node transmits the information to all adjacent nodes and from there, once they have received it, the information continues to be transmitted to forthcoming adjacent nodes, and so forth.

A node by default always transmits the information if it is susceptible and has already received it. Information is never simultaneously transmitted from a node at the time instant it receives it; transmission happens at earliest at the succeeding time instant of the time instant in which the node receives the information. Information is also never transmitted from an insusceptible node, although the node might possibly have received the information at an earlier stage of the process. Inherently, information is also never transmitted from a node which has not received it, irrespective of this node's susceptibility state. The spreading process continues until all nodes in the graph have received the information; the IPM terminates thereof. The susceptibility status of each node is as said determined by a 2-state MC for the given state space defined in equation (4.1). Mathematically, the transitions are formulated for every node $v_i \in V$ as

$$p_{\text{sus}\to\text{insus}}(v_i) := \mathbb{P}\text{r}(v_i \text{ is insusceptible at time } t+1 \mid v_i \text{ is susceptible at time } t) \tag{4.2}$$

and analogously for the reverse transition $p_{\text{insus}\to\text{sus}}(v_i)$; the complementary event is of course that the node remains in its current state e.g.,

$$p_{\text{sus}\to\text{sus}}(v_i) := 1 - p_{\text{sus}\to\text{insus}}(v_i).$$

Furthermore, it is assumed that the inherent MC of each node is time-invariant.

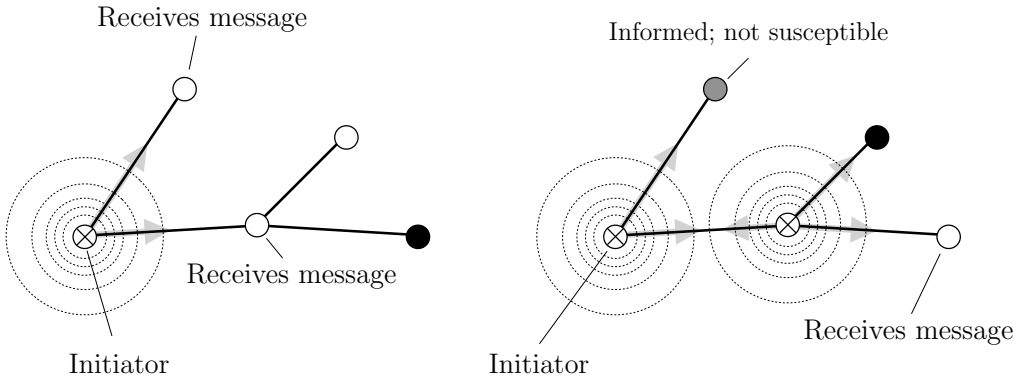

Figure 4.1: A possible situation of the information spreading, which happens locally on a region of the graph which the IPM is performed on.

At the initiation of the information spreading, which happens at time $t = t_0$, the susceptibility of each node is randomly generated i.e., for any $v_i \in V$

$$p_{init}(v_i) := \mathbb{P}\mathrm{r}(v_i \text{ is susceptible at time } t_0), \quad v_i \neq v_{\text{initiator}}. \tag{4.3}$$

As a node by definition must be susceptible in order to transmit the information, naturally the initiator node is susceptible by default i.e.,

$$p_{init}(v_{\text{initiator}}) := \mathbb{P}\mathrm{r}(v_{\text{initiator}} \text{ is susceptible at time } t_0) = 1.$$

A possible local scenario of the propagation is depicted in Figure 4.1, which provides a principal overview of the process. In figure, white nodes are susceptible, but have yet to receive the information. Black nodes are insusceptible; they can not have received the information at an earlier stage, as in this example the initiator node starts the information spread. White nodes with a marked cross are both susceptible and have received the information – thus, they transmit the information. Lastly, the grey node is insusceptible, although informed. The time instant **(A)** shows that information is transmitted from the initiator node to all adjacent nodes; both adjacent nodes to the initiator node are susceptible and thus they will receive the information. In the succeeding time instant, that is shown in **(B)**, one of the neighbors of the initiator node is still susceptible and thus transmits information; the other neighbor has received the information, but has transitioned into the insusceptible state from the previous time instant, and thus no information is transmitted from this node.

It is assumed that the graph is immutable, or *non-dynamic*, during the lifetime of the IPM. In other words, the graph structure is time-invariant, or at least any re-structuring of the graph happens with much slower pace relative to that of the information propagation. As a remark, the model, as in how the information is transmitted, could of course be applied to dynamic graphs.

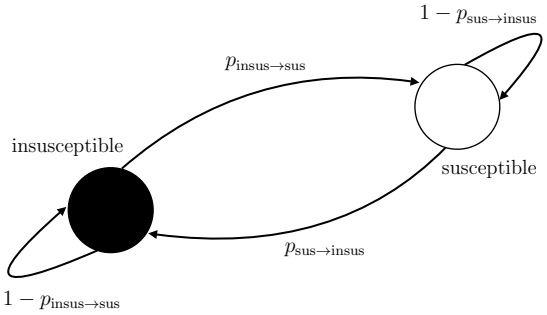

Figure 4.2: A diagrammatic view of the susceptibility state transitions.

The transitioning from one state to another may be regarded as an independent Bernoulli trial; the time spent in certain state before transitioning to another is therefore *first success* distributed. The time which a node v_i e.g., spends as *susceptible* before transitioning to *insusceptible* state has distribution

$$\tau_{\text{sus}\to\text{insus}}(v_i) \in \mathbb{G}(p_{\text{sus}\to\text{insus}}(v_i)).$$

Accordingly, the expectation of this quantity is

$$\mathbb{E}[\tau_{\text{sus}\to\text{insus}}(v_i)] = \frac{1}{p_{\text{sus}\to\text{insus}}(v_i)}, \tag{4.4}$$

the reader may see (Shao, 2014). Naturally, if $p_{\text{sus}\to\text{insus}}(v_i) \to 0$, then $\mathbb{E}[\tau_{\text{sus}\to\text{insus}}(v_i)] \to +\infty$. Obviously, if it is impossible to transition to insusceptible from susceptible, the expected time spent as *susceptible* goes to infinity as the state is never left. In the same regard, given $p_{\text{sus}\to\text{sus}}(v_i) = 1 - p_{\text{sus}\to\text{insus}}(v_i) \to 1$, because $p_{\text{sus}\to\text{insus}}(v_i) \to 0$, it follows

$$\mathbb{E}[\tau_{\text{sus}\to\text{sus}}(v_i)] = \frac{1}{p_{\text{sus}\to\text{sus}}(v_i)} \to 1$$

meaning that the time spent as *susceptible* until *susceptible* again is 1 i.e., at the succeeding time instant. This obviously is in agreement with conception, as $p_{\text{sus}\to\text{insus}}(v_i) \to 0$ means that the MC never leaves the state *susceptible*, once in this state.

Concluding this chapter, the 2-state MC enables us to regulate the nodes' susceptibility as in how they enable transmission of information. In limit-sense therefore, the time which each node spends in a specific state depends on how the transition probabilities are defined. As demonstrated, if e.g., $p_{\text{sus}\to\text{insus}}(v_i) \approx 0$ for some node v_i, then it is expected that it spends long time as susceptible until transition, compared to if $p_{\text{sus}\to\text{insus}}(v_i) \approx 1$ for which case it is expected that transition happens much more rapidly.

Chapter 5

Simulations

This chapter has three parts; the first part demonstrates the centrality measures comparatively and qualitatively for synthetic graphs generated by the Barabási–Albert model. More particularly, the the Bonacich β-centrality measure will be compared to the following: the MBC, the eigenvector centrality and the degree centrality. As already stated, the MBC is not intrinsically considered a measure of centrality, but is nevertheless included to assess possible similarities with the BBM. As for the second part, the information propagation model (IPM) performed is performed on synthetic graphs, generated by the BAM-algorithm. In the ending of this chapter, in the third section, the same simulation is performed as in the second part, but for real network data observed from Facebook.

5.1 Study of Centrality Rankings

Let $G_{\text{BAM}} \propto \mathbf{A}$ be generated according to the BAM-algorithm for $\ell = 2$ and $n = 10^3$, where $\mathbf{A}$ is the corresponding adjacency matrix. Furthermore, let the initial graph be defined as $\mathcal{S}_2$. The following measure of likeness between two centrality score vectors $c_1, c_2 \in \mathbb{R}^{|V|\times 1}$, with respect to the adjacency matrix $\mathbf{A}$, is applied:

$$\Delta(c_1, c_2; \mathbf{A}) := \sum_{i=1}^{|V|} |\widetilde{c}_{1i} - \widetilde{c}_{2i}|, \quad \widetilde{c}_j := \frac{1}{\max_k |c_{jk}|} c_j, \quad j = 1, 2. \tag{5.1}$$

The scaling of each vector according to equation (5.1) naturally yields that the scaled entries lie in $[-1, 1]$. Moreover, the entry-wise subtraction of c_1 and c_2 correspond to the direct difference in the (scaled) centrality score calculated for each node, and equation (5.1) is the aggregation of the absolute values of these differences. A large value of Δ indicates that there is dissimilarity in the rankings between the measures. Figure(s) 5.1-5.4 show scatter plots of the MBC, eigenvalue centrality and degree centrality against the BBM; similar to equation (5.1), all centrality vectors are here scaled by their respective reciprocal of the maximum absolute value.

Notably, the MBC and the eigenvalue centrality show similarity in the scatter plots in how they relate to the BBM, although, as can clearly be read in Table 5.1, usually the MBC has lower dissimilarity to BBM compared to the eigenvalue centrality, judging by the values of Δ.

BBM $\beta := 1/2\hat{\lambda}$		BBM $\beta := -1/2\hat{\lambda}$		BBM $\beta \approx 1/\hat{\lambda}$		BBM $\beta := 0$	
Measure	Δ	Measure	Δ	Measure	Δ	Measure	Δ
MBC	8.99	MBC	14.57	MBC	5.17	MBC	12.67
Degree	13.40	Degree	9.82	Degree	17.77	Degree	0
Eigen	6.72	Eigen	19.14	Eigen	0.01	Eigen	17.78

Table 5.1: Dissimilarity scores determined according to equation equation (5.1); we compute Δ for various parameter choices of the BBM against the degree centrality, eigencentrality and the MBC – each with respect to G_{BAM}.

It can generally be said for this experiment that the big "mass" of centrality scores are concentrated in the lower range, as apparent in Figure(s) 5.1-5.4 (in the lower left corners). It can furthermore be noted, for this experiment, that the measures are relatively similar in terms of how they assign the scores; for instance, nodes which are assigned centrality values in the range $[0, 0.2]$ by the BBM for $\beta := 1/2\hat{\lambda}$ mostly are assigned the same values by the MBC, as shown in Figure 5.1. Although, there are certainly exceptions to this observation as e.g., demonstrated in the same Figure 5.3, where it can be noted that a node receives a centrality score of ≈ 0.35 by the eigenvalue centrality and ≈ 0.6 by the BBM for $\beta := -1/2\hat{\lambda}$. Moreover, there is slightly bigger dispersion in the scatter plots corresponding $\beta := -1/2\hat{\lambda}$ for the BBM; see Figure 5.2.

As a final remark to conclude this section and experiment, as far as highly ranked nodes are concerned, the measures produce similar rankings. This is evident from Table 5.2, which presents the top ten nodes with highest centrality score for each measure computed on G_{BAM} in this experiment; To clarify, the node labels lie in the interval $[0, 999]$. Noticeably, all measures determine the node which has label 0 as most central in G_{BAM} – we refer to this node as $\hat{v}$ from here on. As can be concluded from any scatter plot corresponding the degree centrality in Figure(s) 5.1-5.4, $\hat{v}$ unambiguously has the highest degree. A remark in this regard is that G_{BAM} is generated by nodes successively entering the graph, and inducing edges with nodes which already have high connectivity. Therefore, the "early" nodes have high chance of ending up with the highest degrees, which is possibly the case for $\hat{v}$; in this experiment, each measure also deem $\hat{v}$ as most central.

Measure \ Rank Position	1	2	3	4	5	6	7	8	9	10
BBM $\beta := 1/2\hat{\lambda}$	0	3	7	1	10	19	17	5	22	33
BBM $\beta := -1/2\hat{\lambda}$	0	7	1	3	19	22	10	33	55	17
BBM $\beta := 0$ (Degree centrality)	0	3	7	1	5	10	17	6	19	18
BBM $\beta \approx 1/\hat{\lambda}$ (Eigen centrality)	0	7	3	1	19	10	22	17	33	5
Degree centrality	0	3	7	1	5	10	17	6	19	18
Eigenvector centrality	0	7	3	1	19	10	22	17	33	5

Table 5.2: The top ten highest ranked nodes computed for each measure on G_{BAM} in this experiment; the table displays the labels of these nodes. The node which has label 0 has the highest centrality according to each measure.

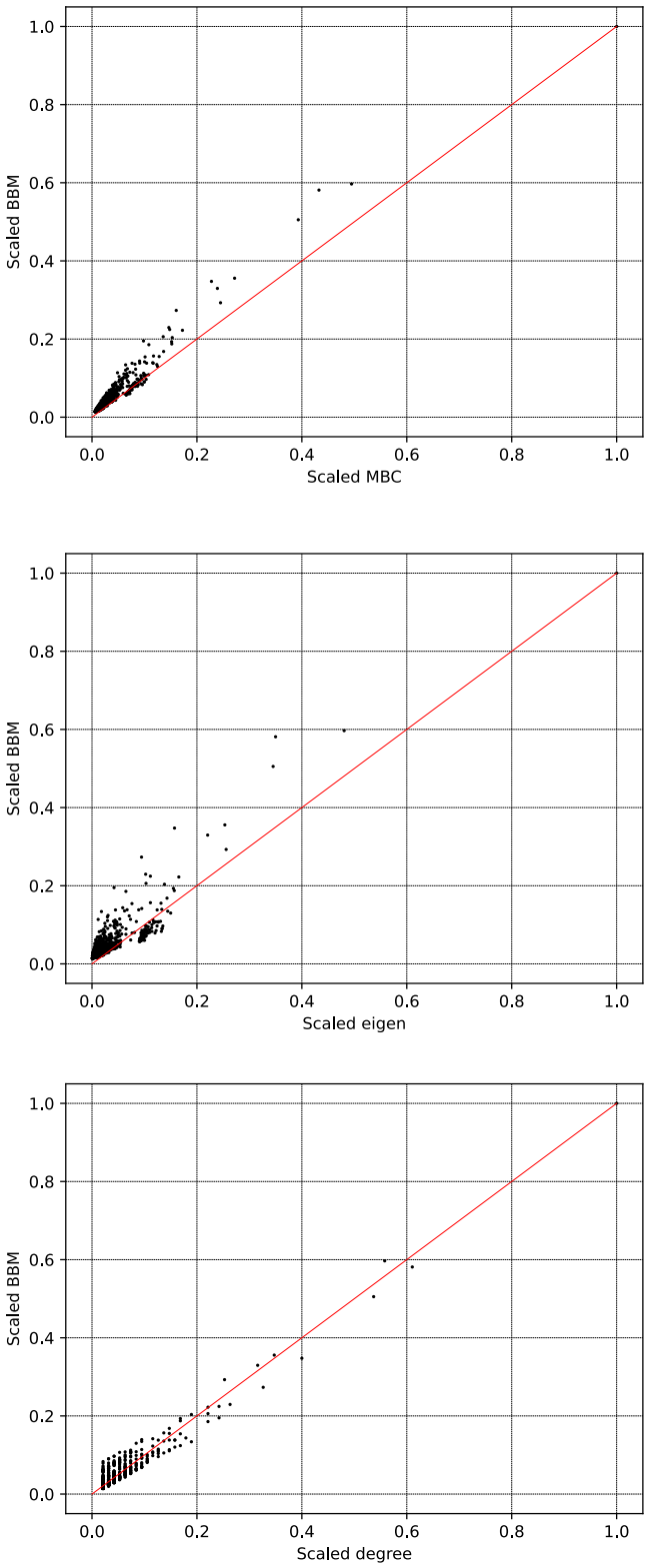

Figure 5.1: Scatter plots for the BBM given $\beta := 1/2\hat{\lambda}$ against the MBC, eigencentrality and the degree centrality - each computed with respect to G_{BAM}.

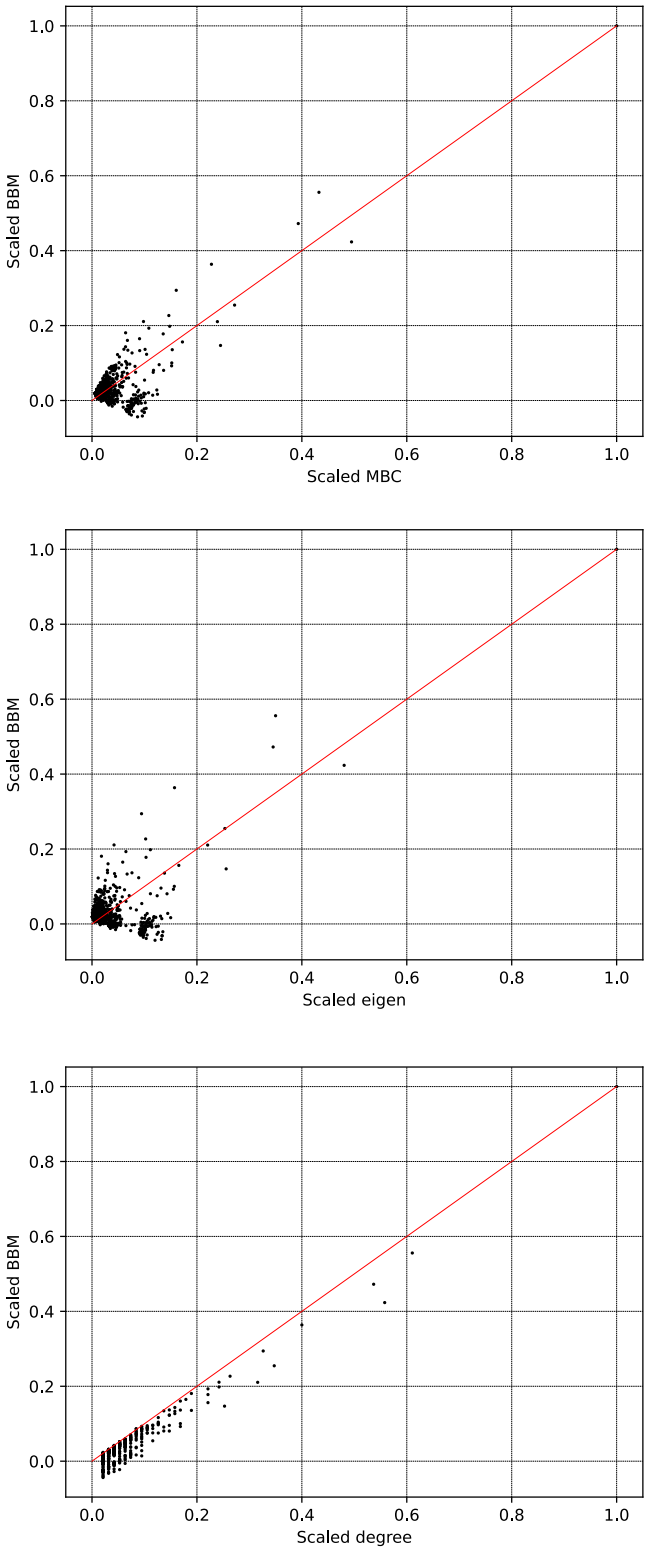

Figure 5.2: Scatter plots for the BBM given $\beta := -1/2\hat{\lambda}$ against the MBC, eigencentrality and the degree centrality - each computed with respect to G_{BAM}.

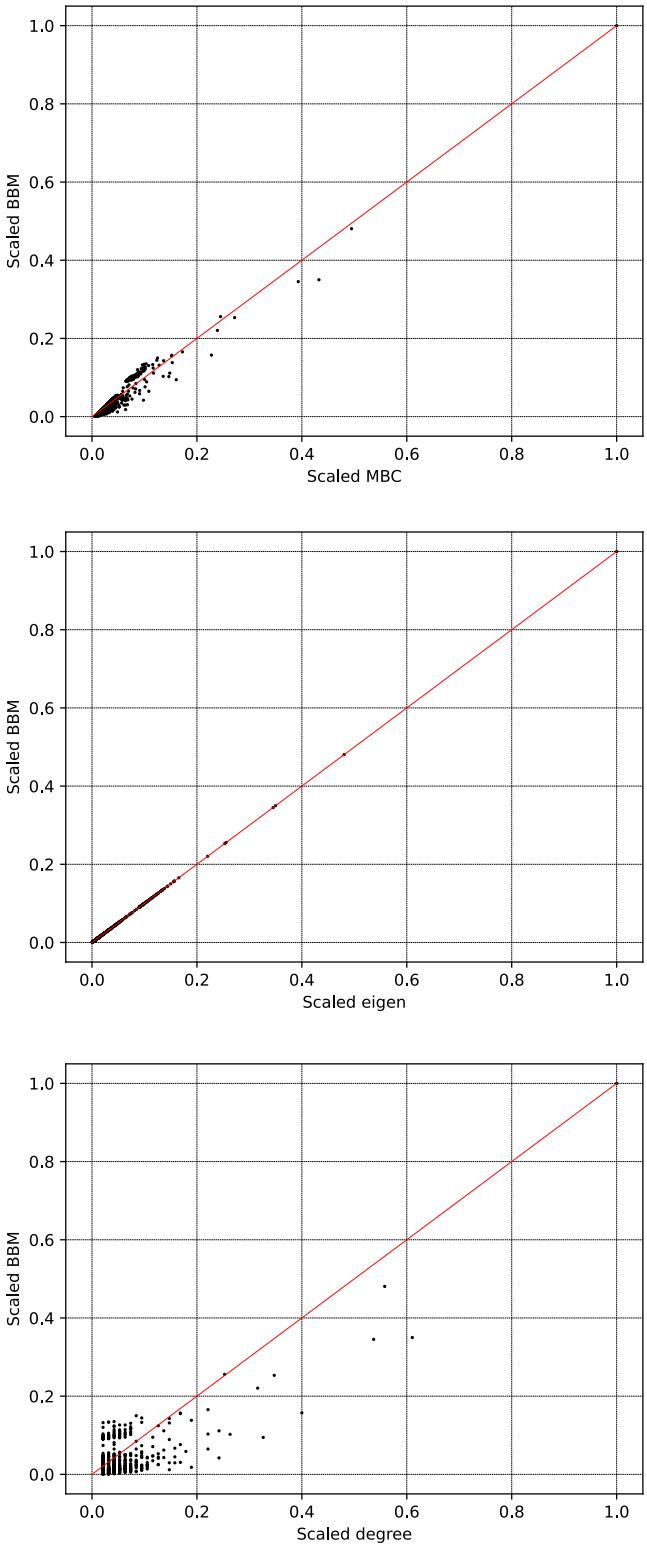

Figure 5.3: Scatter plots for the BBM given $\beta \approx 1/\hat{\lambda}$ against the MBC, eigencentrality and the degree centrality - each computed with respect to G_{BAM}.

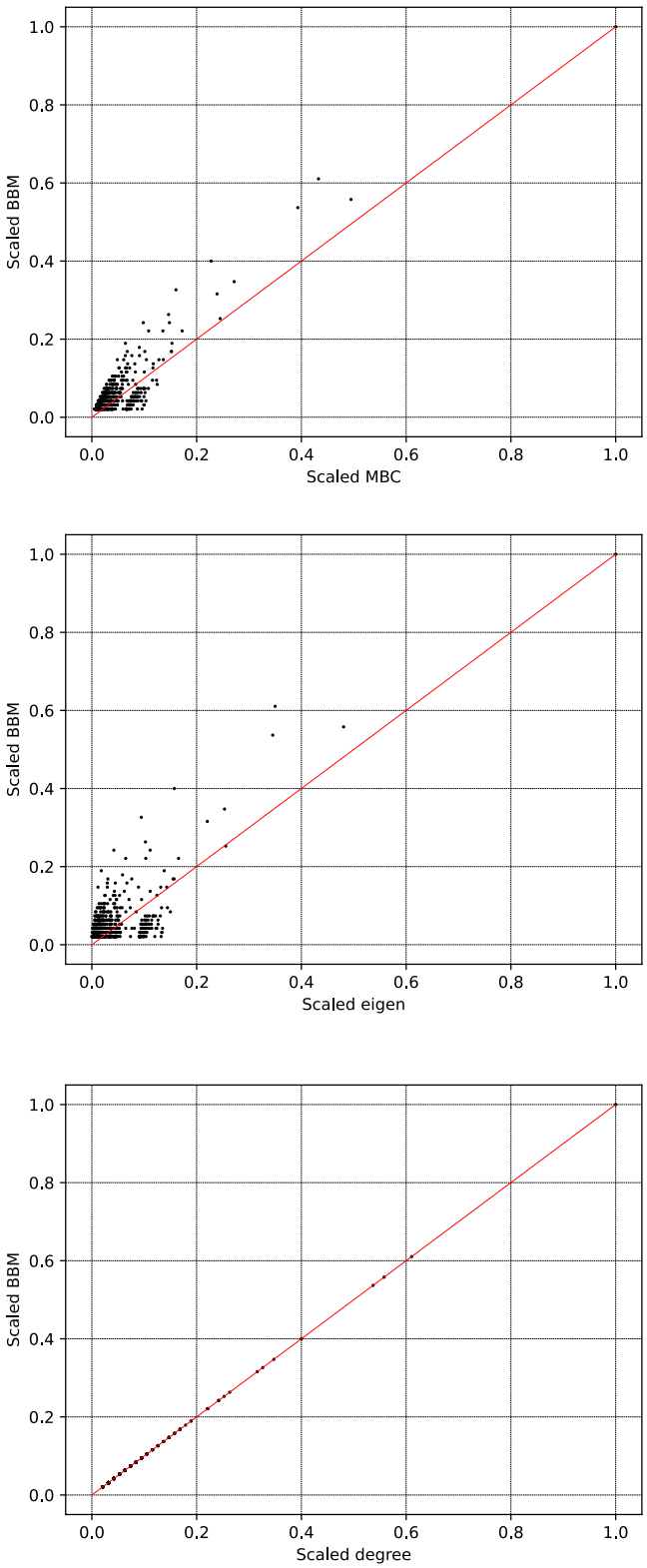

Figure 5.4: Scatter plots for the BBM given $\beta := 0$ against the MBC, eigencentrality and the degree centrality - each computed with respect to G_{BAM}.

5.2 Information Propagation Simulations

In this section, the IPM is applied on synthetic graphs which are generated according to the BAM-algorithm for $\ell = 2$ and $n = 10^3$ throughout; furthermore, the initial graph input to the BAM shall be defined as $\mathcal{S}_2$. In accordance with the construction of the IPM, the graph structure is stationary across intra-simulation runs. In modeling aspect, this reflects that communication takes place with much faster pace than the pace of any potential re-shaping of the graph structure as e.g., induced edges or nodes leaving the graph. Furthermore, the IPM simulation will be performed on the same graph, which will be referred to as $\mathbf{A} \propto G_{\text{BAM}}$, where $\mathbf{A}$ is the corresponding adjacency matrix. The reader is referred to Appendix B for principal code implementation.

The purpose of this simulation is to investigate the time until all nodes in the graph have received the information; in accordance with the IPM rules, the information propagation is induced by the transmissions of the nodes and as a whole initiated by a single node, which we refer to as the initiator node. In order to quantify the impact of centrality, the information propagation is compared against the centrality ranking of the initiator node. As shall be recalled, an *important* node is defined to be within the top 1% in centrality scores. Conversely, an *unimportant* node is any node which belong to the bottom 1%. The centrality metrics applied in this experiment are the following: the eigenvector centrality, Bonacich's β-measure and lastly, the MBC. As regards the β-measure, the parameters are set as $\beta := \pm 1/2\hat{\lambda}$, where

$$\hat{\lambda} := \max_i \lambda_i(\mathbf{A}).$$

The IPM is independently run 10^4 times on the same BAM-generated graph. Each simulation run is set to terminate before the next is initiated i.e., one simulation run is fully performed when the information has reached the entire graph. The status of the initiator node is divided into three categories: i) an important (top 1% in centrality rankings) ii) an unimportant (bottom 1%) and lastly iii) an arbitrary node; the latter is randomly selected from the node set corresponding G_{BAM}, irrespective of its centrality ranking. Furthermore, for each simulation run the initiator node is chosen randomly within the set of nodes which belong the same category. At the beginning of each simulation run, the attribute for each node is reset i.e., all nodes $v_i \neq v_{\text{init}}$ have of course not received the information, whereas v_{init} has as it is the source of the spreading. The susceptibility for each node at the beginning of the information spread is random according in accordance with equation (4.3),

$$p_{init}(v_i) := \mathbb{P}\text{r}(v_i \text{ is susceptible at time } t_0) := \frac{1}{2}, \quad v_i \neq v_{\text{initiator}}.$$

In this experiment, the transition probability from either susceptibility state to the other is set as equal i.e., for any node $v_i \in V \propto G_{\text{BAM}}$

$$p_{\text{sus}\to\text{insus}}(v_i) = p_{\text{insus}\to\text{sus}}(v_i) := \frac{1}{2}.$$

Hence, as motivated by equation equation (4.4), the expected time spent in either state is $\mathbb{E}[\tau] = 2$.

Results for the MBC

Figure 5.5 shows the simulation results for the MBC; the ordinate shows the mean of the number of nodes which have received the information over 10^4 simulations against time, which is displayed on the abscissa. Notably, the curves corresponding each category of initiator node do have the similar "s"-shape; the overall result output suggests that the centrality status of the initiator indeed node has impact on the pace of the transmission globally. This is particular evident in time-regions close to $t_0 := 0$; it can for instance be seen in figure that at time $t = 5$ the average number of informed nodes is 400 for the important initiator – whereas it is only 25 for the unimportant initiator. Furthermore, the information has reached the entire graph at $t = 20$ for the important initiator and at $t = 25$ for the unimportant.

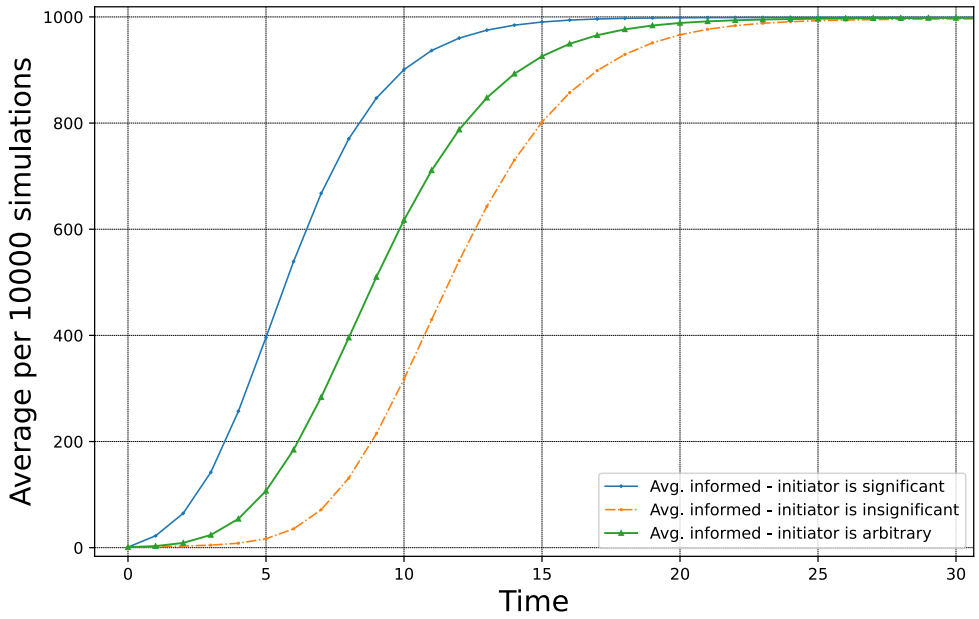

Figure 5.5: Curves of the mean number of informed nodes over 10^4 simulations for each category of initiator; the centrality scores are determined according to the MBC and the graph which the IPM is performed on is generated by the BAM for the intitial graph $\mathcal{S}_2$ and $\ell = 2$. The total number of nodes is $n = 1{,}000$.

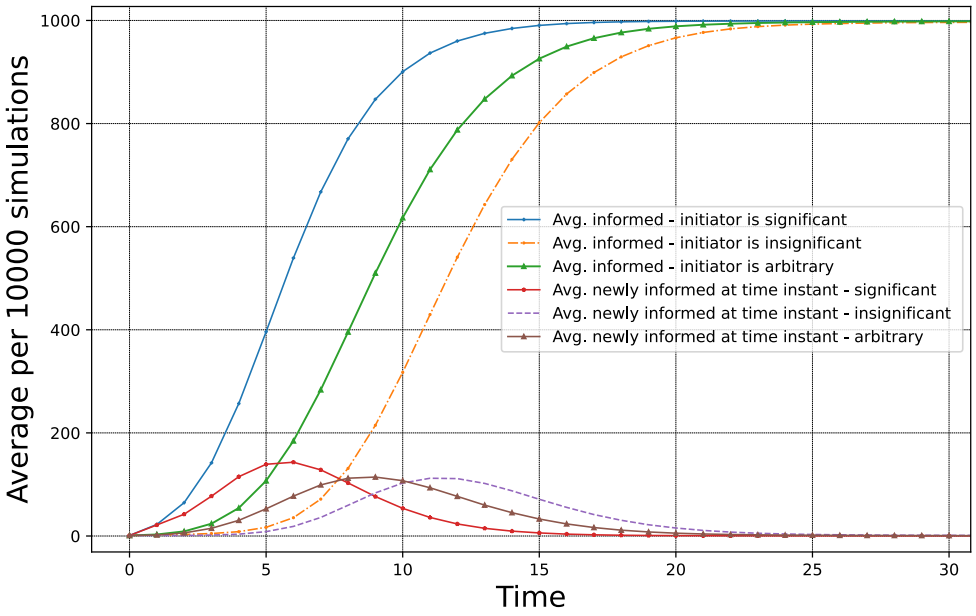

Figure 5.6: This plot shows add-on to Figure 5.5, namely curves of the average number of newly informed nodes per time instant over 10^4 simulations, for each category of initiator node.

Figure 5.6 contains the same curves as in Figure 5.5 and the average number of newly informed nodes per time instant. The peak of the number of newly informed nodes per time instant occurs sooner for the important initiator node. Naturally, the peaks also occur at the inflexion point(s) of the curves of the number of informed nodes. Furthermore, the peak width of the curve corresponding the important initiator is narrower compared the curve corresponding the unimportant initiator; altogether, from a dynamic point of view and irrespective of the centrality status of the initiator node, the rate of the propagation reaches a maxima from which it subsides definitely. It shall also be remarked that the spreading appears as more protracted for the unimportant node, as the peak is much more "smeared" over time.

Results for the Eigenvector centrality and Bonacich's β-centrality

The IPM is performed with regard to the eigenvector centrality and BBM for $\beta := 1/2\hat{\lambda}$. The results are shown in Figure 5.7. Principally, the outcomes are the same as when using the MBC (Figure 5.6) suggesting that the choice of measure, in mean-sense, has no significant impact in this experiment.

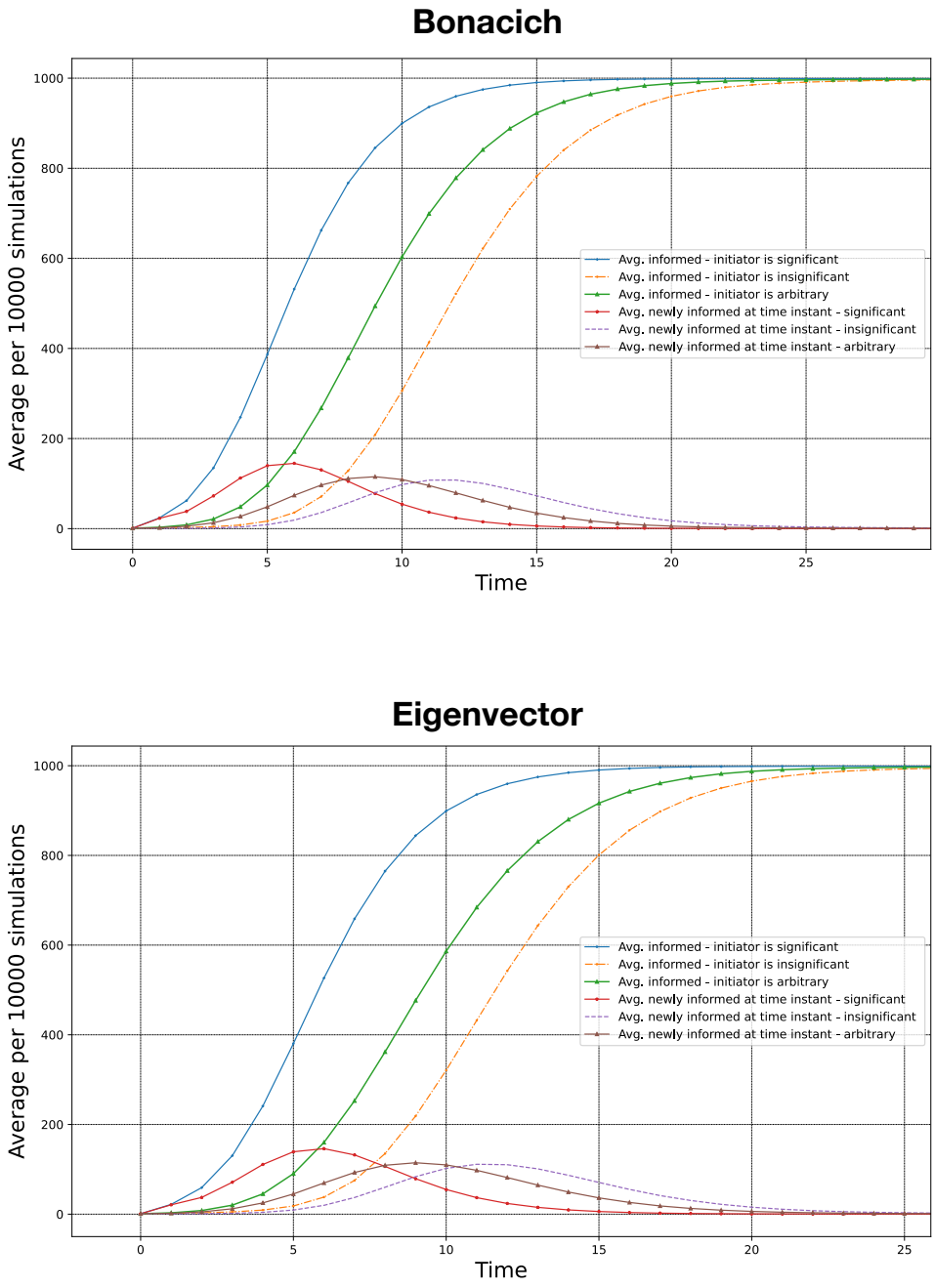

Figure 5.7: The plots show the curves of the average number of informed nodes over 10^4 simulations for each category of initiator; furthermore, the curves show the average number of newly informed nodes per time instant. The centrality rankings are determined by the eigenvector centrality and BBM for $\beta := 1/2\hat{\lambda}$.

5.3 Information Propagation Simulation on Real Network

In this section, the IPM is applied on a *Facebook* graph; it is undirected, connected and constituted by 4,039 nodes and 88,234 edges. A depiction of the graph can be seen in Figure 5.8.

Let $G_{\text{Facebook}} \propto \mathbf{A}_{\text{Facebook}}$. The IPM is performed for two cases where the MBC and the BBM (using $\beta := 1/2\hat{\lambda}$) are applied respectively;

$$\hat{\lambda} := \max_i \lambda_i(\mathbf{A}_{\text{Facebook}}).$$

The IPM is performed in total 10^3 times for each category of initiator node. Furthermore, as in previous simulations, the transition probabilities are set as,

$$p_{\text{sus}\to\text{insus}}(v_i) = p_{\text{insus}\to\text{sus}}(v_i) := \frac{1}{2}.$$

and the initiating probability as,

$$p_{init}(v_i) := \frac{1}{2}, \quad \forall v_i \neq v_{init}.$$

The results of the simulation for the MBC are presented in Figure 5.9; plot **(A)** shows the average number of nodes which have received the information against time, over 10^3 simulations. Moreover, the plot **(B)** is a zoom-in of **(A)** over a limited time range. Notably, the curve of newly informed nodes corresponding the *important* initiator node increases rapidly at $t_0 := 0$ and from there, it is smooth; for any other type of initiator node, the curve is smooth from directly from $t = 0$. This difference indicates that the *important* initiator nodes have high degree, whereas the *unimportant* do not.

The simulation results for this experiment share many similarities with the result obtained for the IPM simulation performed for synthetic graphs in Section 5.2; the curves are likewise "s"-shaped and also here the centrality of the initial sender has evident impact. A slight difference is however that the curves corresponding an *important* and arbitrary initiator are closer to each other compared to how they are when simulating the IPM for BAM-generated graphs in Section 5.2.

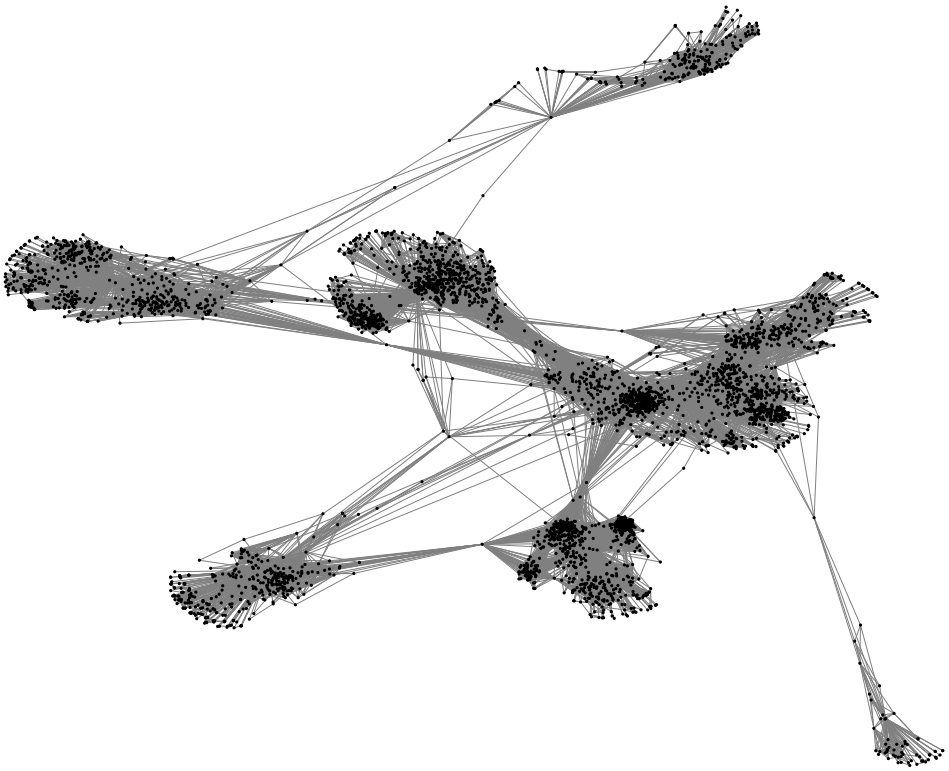

Figure 5.8: An overview of a *Facebook* graph on which the IPM is performed on.

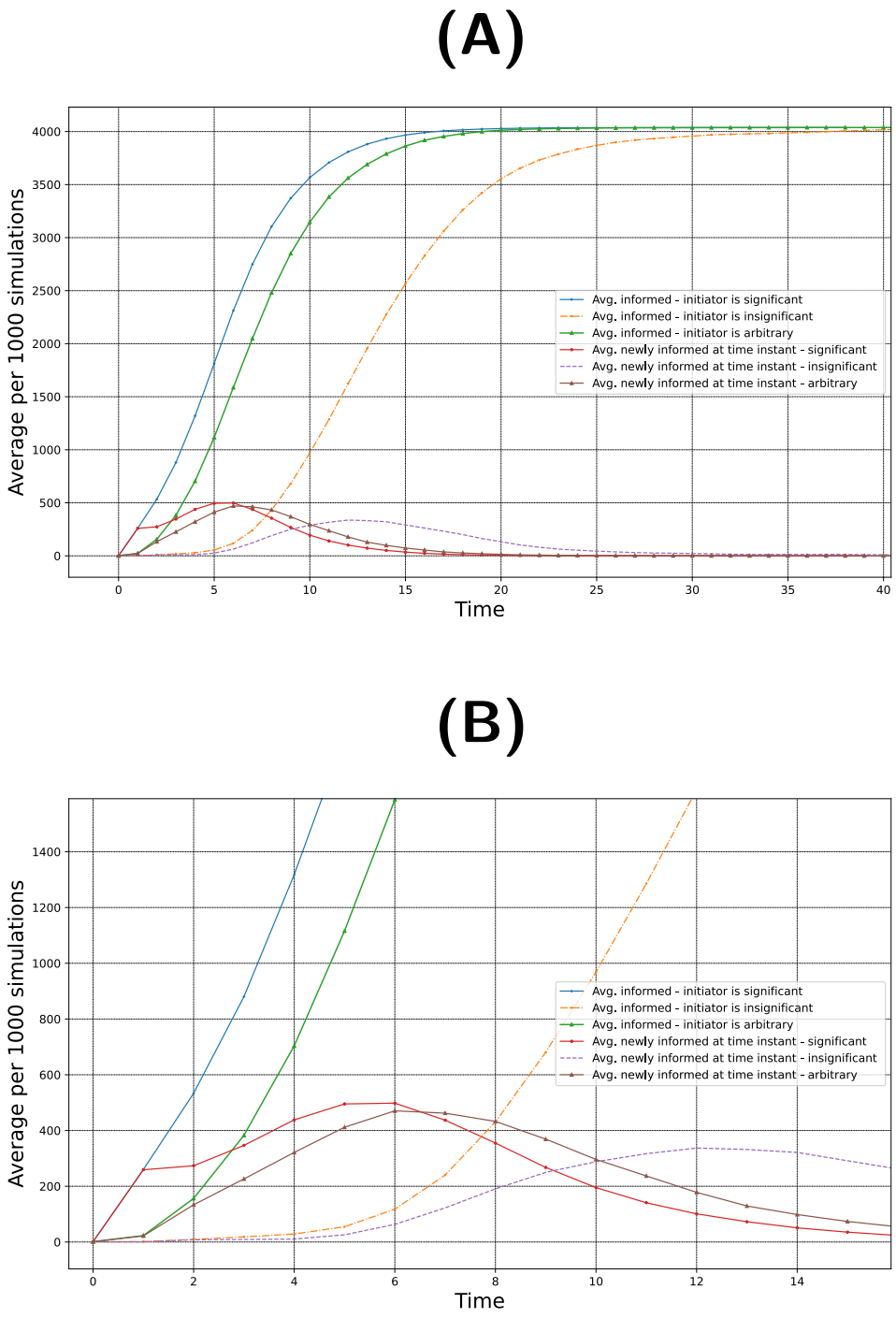

Figure 5.9: This figure shows the IPM simulation output when applying MBC; plot (**B**) is a zoom-in of plot (**A**)

.

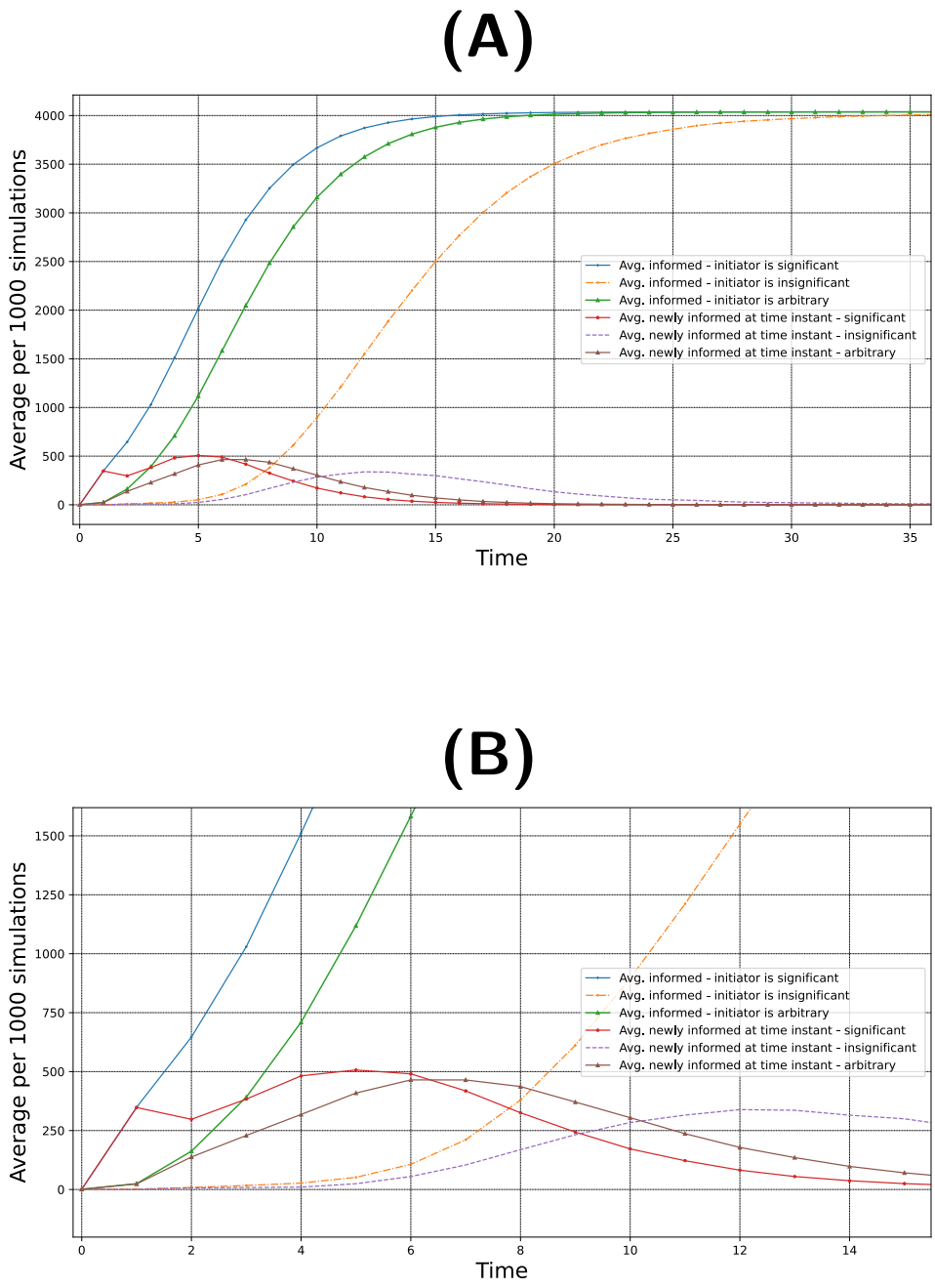

Figure 5.10: This figure shows the IPM simulation output when applying BBM for $\beta := 1/2\hat{\lambda}$; plot (**B**) is a zoom-in of plot (**A**)

.

Chapter 6

Discussion

In agreement with intuition, the outcome of the simulations indicate that the centrality of the initiator node – from which the information propagation is initiated according to the IPM – has evident impact on transmissions globally; the information is on average spread significantly faster if initiated by an *important* node. Altogether, the results suggest that the centrality measures are able to identify influential nodes in the graph in regard of how they enable fast spreading in mean-sense. It is furthermore demonstrated by Table 5.1 that there can indeed be certain dissimilarities between the measures; recall that the graph used in the qualitative study in Section 5.1 is generated by the BAM with the same parameters used for the graphs in the IPM simulations. Although for the IPM simulations, evidently these dissimilarities do not seem to affect much. A possible reason for this is that the measures principally determine the *same* most central nodes; this possibility is demonstrated by Table 5.2. There are however certain limitations which needs to be addressed. Firstly, the graphs on which the IPM do not change over time. Therefore, they do not resemble dynamics most likely to be observed in real networks. It remains to study dynamical graphs which emulate those of real networks in order to draw better conclusions of how the IPM acts as a propagation model; although, the IPM is applicable on dynamic graphs so this scope can easily be expanded. Moreover, the IPM is very rudimentary and it may at best only replicate basic dissemination mechanisms. In simulations, we have also defined a homogeneous population in the sense that all nodes have the same transition probabilities for their inherent MC:s, independent of time. In this sense, it might be worth studying how the information propagation is affected when implementing a time-dependent collective behaviour for various groupings of the nodes – as a suggestion, one could define the transition probabilities differently for each groups and let the probabilities be e.g., time-variant and dependent on various state(s) of the graph.

The measures treated in this thesis can be deemed to relate well to the real settings displayed, although they are somewhat redundant to one another as they yield similar rankings. In terms of their applicability, there is indeed a major limitation. If the centrality measures treated in this thesis were to be applied for a real network as e.g., a social network, one would need

to know the corresponding adjacency matrix. In addition, it is common for networks to be massive, which would furthermore lead to high computational cost. An intriguing thought which could overcome this problematic is to find approximate methods which could approximate the centrality without the need of knowing the full graph – which ideally should also be aimed to be less computationally expensive. Furthermore, if modeling information spreading on a real network using the principles of the IPM, one would need to know exactly the graph structure at all time.

The dynamics of the IPM as they appear in this thesis have similarities with the SIR compartmental model; most notably, the "s"-shape of the number of nodes which have received the information over time. In this analogy, the number of nodes which have received the information may be regarded as the number of infectious individuals at certain time, as according to the IPM, the informed nodes are those that are transmitting the information under the premise that they are *susceptible*. In the same interpretation, if the nodes represent individuals within a community, their inherent MC:s regulate their exposure to the network. Any node v_i which has low transition probability $p_{\text{sus}\to\text{insus}}(v_i)$ and high probability $p_{\text{insus}\to\text{sus}}(v_i)$ is expected to be more exposed to the information (disease) transmitted by adjacent nodes as it spends more time in an "active" role of the transmission. At the same time, once v_i receives the information (i.e., it becomes infected) and has high exposure still, it will transmit more information during the lifetime of the spreading compared to if it had low exposure. The curve which corresponds to the number of nodes which have received the information per time *instant* tells in the same context the number of infected individuals – recall however, according to the IPM they must also be *susceptible* to transmit the information, or disease in this case. A major flaw with the SIR-interpretation of the IPM is that the IPM does not account for nodes becoming "immune" to the information. Hypothetically, if $p_{\text{sus}\to\text{insus}}(v_i) = p_{\text{insus}\to\text{sus}}(v_i) \neq 0$, this can be interpreted as a scenario in which a disease spreads uncontrollably and the individuals display the same behaviour pattern irrespective of whether or not they are infected. In slight dispute with the theoretical arguments which adjudge that the IPM is a poor epidemiological model, there are notable similarities still with the curves obtained in this thesis and those of (Li et al., 2020) (in particular, Fig.1 in the paper); this referred paper conducts an analysis of the propagation of the spread of Covid-19 in Hubei Province, China. Most notably, the "s-shape" of number of infected for this real observation indeed has similar appearance to the number of *informed* of the IPM. In the same context, the results in this thesis also highlight that centrality measures could be applicable for assessing the importance of *patient zero*'s according to his or hers position in the community.

Lastly for this discussion, we emphasize that the MBC is derived without any deeper reflection in terms of its usefulness as a centrality metric; this is in itself a suggestion for further work. One clear benefit of the MBC, unlike the Bonacich β-measure, is that it produces rankings which do not depend on a hyper-parameter. In contrast, it is evident from Figure 1.1 that the choice of β has certain impact of the rankings, which raises the question of how the parameter should be specified. Regarding the Bonacich measure, the thesis straightforwardly conducted some treatment of its properties – it was for instance proven in Proposition 2.7

that the inequality

$$\|c(1,0)\|_1 = \sum_{v \in V} \deg v \leq \hat{\lambda}|V|,$$

which is well-recognized in literature, indeed can be derived from the BBM. Furthermore, the eigenvector centrality (usually) and the degree centrality can both be derived from the BBM. Overall, these aspects might justify further study of the measure; a suggestion is to analyze the dynamical properties of the measure, based on Proposition 2.4.

Bibliography

Albert, R. & Barabási, A. (2002), 'Statistical mechanics of complex networks', *Reviews of Modern Physics* **74**(1), 47–97.

Andrews, L. (1998), *Special functions of mathematics for engineers*, 2 edn, London: Oxford University Press.

Aprahamian, M. & Higham, N. (2014), 'The matrix unwinding function, with an application to computing the matrix exponential', *SIAM Journal on Matrix Analysis and Applications* **35**(1), 88–109.

Barabási, A.-L. & Albert, R. (1999), 'Emergence of scaling in random networks.', *Science* **286**(5439), 509–512.

Benzi, M. & Klymko, C. (2015), 'On the limiting behavior of parameter-dependent network centrality measures', *SIAM Journal on Matrix Analysis and Applications* **36**(2), 686–706.

Bonacich, P. (1987), 'Power and centrality: A family of measures', *American Journal of Sociology* **92**(5), 1170–1182.

Bonacich, P. (2007), 'Some unique properties of eigenvector centrality', *Social Networks* **29**(4), 555–564.

Borgatti, S. & Everett, M. (2006), 'A graph-theoretic perspective on centrality', *Social Networks* **28**(4), 466–484.

de Abreu, N. (2007), 'Old and new results on algebraic connectivity of graphs', *Linear Algebra and its Applications* **423**(1), 55–73.

Higham, N. (2008), *Functions of matrices: Theory and computation*, SIAM.

Horn, R. & Johnson, C. (2012), *Matrix Analysis*, Cambridge: Cambridge University Press.

Hunter, J. (2014), 'An introduction to real analysis'.
URL: *https://www.math.ucdavis.edu/∼hunter/intro_analysis_pdf/intro_analysis.pdf*

Klein, D. (2010), 'Centrality measure in graphs', *International Journal of Critical Infrastructure Protection* **47**(4), 1209–1223.

Li, L., Yang, Z., Dang, Z., Meng, C., Huang, J. & Meng, H. (2020), 'Propagation analysis and prediction of the covid-19', *Infectious Disease Modelling* **5**, 282–292.

Lukes, D. (1982), *Differential equations: Classical to controlled*, Elsevier Science & Technology.

Mcauley, J. & Leskovec, J. (2012), 'Learning to discover social circles in ego networks'.
URL: *https://www.math.ucdavis.edu/∼hunter/intro_analysis_pdf/intro_analysis.pdf*

Renardy, M. & Rogers, R. (2010), *An introduction to partial differential equations*, New York: Springer.

Serfozo, R. (2009), *Basics of applied stochastic processes. Probability and its applications*, New York: Springer.

Shao, J. (2014), *Mathematical Statistics*, New York: Springer.

SNAP: Network datasets: Social circles (n.d.).
URL: *https://snap.stanford.edu/data/ego-Facebook.html*

Stergiopoulos, G., Kotzanikolaou, P., Theocharidou, M. & Gritzalis, D. (2015), 'Risk mitigation strategies for critical infrastructures based on graph centrality analysis', *International Journal of Critical Infrastructure Protection* **10**, 34–44.

Stevanovic, D. (2014), *Spectral Radius of Graphs*, San Diego, CA: Academic Press.

Zhou, J., Cui, G., Hu, S., Zhang, Z., Yang, C. & Liu, Z. (2020), 'Graph neural networks: A review of methods and applications', *AI Open* **1**, 57–81.

Appendix A

Computes the Bonacich β-centrality over a range of defined values and plots the scores

```
import sys
import numpy as np
import matplotlib.pyplot as plt
import networkx as nx
from numpy import linalg as la

def computeBonacichMeasure(adjMatrix, beta):
    n = len(adjMatrix[0, :])
    return np.matmul(np.matmul(la.inv(np.add(np.identity(n),
    -beta*adjMatrix)), adjMatrix), np.ones((n, 1)))

nbrOfNodes = 10
G = nx.barabasi_albert_graph(nbrOfNodes, 2)
A = nx.adjacency_matrix(G).todense()
adjacencyMatrix = np.zeros((nbrOfNodes, nbrOfNodes))

for i in range(nbrOfNodes):
    for j in range(nbrOfNodes):
        adjacencyMatrix[i][j] = A[i,j]

eigenVals, _ = la.eig(adjacencyMatrix)
eigenVals.sort()
sensitivity = 1    # shall be > 0
lambdaAst = eigenVals[len(eigenVals) - 1] + sensitivity
```

```
25
26 resolution_N = 210
27 intv = np.linspace(-1 / lambdaAst, 1 / lambdaAst, resolution_N)
28 result = np.zeros((nbrOfNodes, resolution_N))
29
30 for i in range(resolution_N):
31     # strictly real, but format .0j so supress this:
32     temp = computeBonacichMeasure(adjacencyMatrix, intv[i]).real
33     for k in range(nbrOfNodes):
34         result[k][i] = temp[k]
35
36 for k in range(nbrOfNodes):
37     plt.plot(intv, result[k, :], linewidth = 0.5)
38 plt.grid(color='black', alpha=0.1, linestyle='dashed', linewidth=0.25)
39 plt.show()
```

Appendix B

Information propagation simulation over a Barabasi-Albert network

```
import math
import sys
import numpy as np
import matplotlib.pyplot as plt
import random
import networkx as nx
import statistics

from numpy import linalg as la
from scipy.sparse import csgraph
from scipy.linalg import logm, expm

initProbability = 0.5     # initial probability node is susceptible @init time

# Node class
class Node:
    def __init__(self, isImportant, p_OnOff, p_OffOn):
        self.hasReceivedInformation = False
        self.neighbors = []

        self.p_OnOff, self.p_OffOn = p_OnOff, p_OffOn  # transition probabilities
        self.isSusceptible = False

    def willTransmit(self):
        return True

    def willReceive(self):
```

```
28          return True
29
30      def setOnlineStatus(self):
31          if random.random() < initProbability:
32              self.isSusceptible = True
33          else:
34              self.isSusceptible = False
35
36      def regenerateOnlineStatus(self):
37          if self.isSusceptible:
38              if random.random() < self.p_OnOff:
39                  self.isSusceptible = False
40          else:
41              if random.random() < self.p_OffOn:
42                  self.isSusceptible = True
43
44  # Mean Bonacich Measure (MBC)
45  def getMBC(adjMatrix):
46      sensitivity = 0.01
47      eigen, _ = la.eig(adjMatrix)
48      n = len(adjMatrix[0, :])
49      return np.matmul(logm(
50          np.add(np.multiply(2, la.inv(np.add(np.identity(n),
51          (np.multiply(- 1 / (max(eigen) + sensitivity), adjMatrix))))),
52                 np.multiply(-1, np.identity(n)))), np.ones((n, 1)))
53
54  # Bonacich Beta-Measure Centrality
55  def getBonacich(adjMatrix):
56      # Computes the beta = 1 / 2 * lambda_hat
57      eigen, _ = la.eig(adjMatrix)
58      beta = 1/(2*max(eigen))
59      return np.matmul(np.matmul(la.inv(np.add(np.identity(len(adjMatrix[0, :])),
60          np.multiply(-beta, adjMatrix))), adjMatrix),
61                       np.ones((len(adjMatrix[0, :]), 1)))
62
63  # Eigen Centrality
64  def getEigenCentrality(adjMatrix):
65      eigVals, eigenVectors = la.eig(adjMatrix)
66      scores = eigenVectors[:, np.argmax(eigVals)]
67
68      if any(t < 0 for t in scores):
69          scores = (-1) * scores / la.norm(scores)
70      else:
```

```
71          scores = scores / la.norm(scores)
72      return scores
73
74  # Degree Centrality
75  def getDegree(adjMatrix):
76      eigen, _ = la.eig(adjMatrix)
77      beta = 0
78      return np.matmul(np.matmul(la.inv(np.add(np.identity(len(adjMatrix[0, :])),
79      np.multiply(-beta, adjMatrix))), adjMatrix),
80                      np.ones((len(adjMatrix[0, :]), 1)))
81
82  # GLOBAL PARAMETERS:
83  nbrOfNodes, nbrOfImportantNodes = 1000, 10
84  l = 2
85  G = nx.barabasi_albert_graph(nbrOfNodes, l)
86  A = nx.adjacency_matrix(G).todense()
87  adjacencyMatrix = np.zeros((nbrOfNodes, nbrOfNodes))
88
89  for i in range(nbrOfNodes):
90      for j in range(nbrOfNodes):
91          adjacencyMatrix[i][j] = A[i,j]
92
93  centrality = getBonacich(adjacencyMatrix)
94
95  populationSorted = sorted(range(len(centrality)),
96              key = lambda x: centrality[x], reverse = False)[-nbrOfNodes:]
97  important = populationSorted[-nbrOfImportantNodes:]
98  unimportant = populationSorted[0: nbrOfImportantNodes]
99
100 p_OnOff, p_OffOn = 0.5, 0.5
101 Nodes = [None] * nbrOfNodes
102 for i in range(nbrOfNodes):
103     if i in important:
104         Nodes[i] = Node(True, p_OnOff, p_OffOn)
105     else:
106         Nodes[i] = Node(False, p_OnOff, p_OffOn)
107
108     for j in range(nbrOfNodes):
109         if adjacencyMatrix[i][j] > 0:
110             Nodes[i].neighbors.append(j)
111
112 nbrOfSimulations = 1e2
113 timeLimit = 1000
```

```
114
115 print("run simulations...")
116 # RUNS SIMULATION FOR important NODES
117 simulationResults = dict()
118 for i in range(timeLimit):
119     simulationResults[i+1] = list()
120
121 informedResults = dict()
122 for i in range(timeLimit):
123     informedResults[i+1] = list()
124
125 run = 0
126 while run < nbrOfSimulations:
127     print(run)
128     run += 1
129     for i in range(nbrOfNodes):
130         Nodes[i].hasReceivedInformation = False
131         Nodes[i].setStatus()
132
133     initiator = random.choice(important)
134
135     Nodes[initiator].hasReceivedInformation = True
136     Nodes[initiator].isSusceptible = True
137     activeSenders = list([initiator])
138     generations = 0
139     total_informed = 1
140     breakSimulation = False
141     while total_informed < nbrOfNodes:
142         nbrOfInformed = 0
143         generations += 1
144         simulationResults[generations].append(total_informed)
145         activeSendersUpdate = []
146         for i in activeSenders:
147             if Nodes[i].isSusceptible:
148                 if Nodes[i].willTransmit():
149                     for neighbor in Nodes[i].neighbors:
150                         if Nodes[neighbor].isSusceptible:
151                             if not Nodes[neighbor].hasReceivedInformation:
152                                 if Nodes[neighbor].willReceive():
153                                     total_informed += 1
154                                     nbrOfInformed += 1
155                                     Nodes[neighbor].hasReceivedInformation = True
156                                     activeSendersUpdate.append(neighbor)
```

```
                                        if total_informed == nbrOfNodes:
                                            breakSimulation = True
                                            break
            informedResults[generations].append(nbrOfInformed)
            activeSenders = (activeSenders + activeSendersUpdate).copy()
            for i in range(nbrOfNodes):
                Nodes[i].regenerateStatus()

meanimportant = []
generations = []
ct = 0
for i in range(timeLimit):
    tempRecipients = simulationResults.get(i + 1)
    if len(tempRecipients) > 1:
        generations.append(i + 1)
        if i + 1 - ct > 1:
            sys.stdout.write("Error in simulation")
            break
        ct = i + 1
        tempMean = statistics.mean(tempRecipients)
        meanimportant.append(tempMean)

mean_info_important = []
generations_info = []
ct = 0
for i in range(timeLimit):
    tempRecipients = informedResults.get(i + 1)
    if len(tempRecipients) > 1:
        generations_info.append(i + 1)
        if i + 1 - ct > 1:
            sys.stdout.write("Error in simulation")
            break
        ct = i + 1
        tempMean = statistics.mean(tempRecipients)
        mean_info_important.append(tempMean)


# RUNS SIMULATION FOR unimportant NODES
simulationResults = dict()
for i in range(timeLimit):
    simulationResults[i+1] = list()

informedResults = dict()
```

```
200 for i in range(timeLimit):
201     informedResults[i+1] = list()
202 
203 run = 0
204 while run < nbrOfSimulations:
205     print(run)
206     run += 1
207     for i in range(nbrOfNodes):
208         Nodes[i].hasReceivedInformation = False
209         Nodes[i].setStatus()
210 
211     initiator = random.choice(unimportant)
212 
213     Nodes[initiator].hasReceivedInformation = True
214     Nodes[initiator].isSusceptible = True
215     activeSenders = list([initiator])
216     generations = 0
217     total_informed = 1
218     breakSimulation = False
219     while total_informed < nbrOfNodes:
220         nbrOfInformed = 0
221         generations += 1
222         simulationResults[generations].append(total_informed)
223         activeSendersUpdate = []
224         for i in activeSenders:
225             if Nodes[i].isSusceptible:
226                 if Nodes[i].willTransmit():
227                     for neighbor in Nodes[i].neighbors:
228                         if Nodes[neighbor].isSusceptible:
229                             if not Nodes[neighbor].hasReceivedInformation:
230                                 if Nodes[neighbor].willReceive():
231                                     total_informed += 1
232                                     nbrOfInformed += 1
233                                     Nodes[neighbor].hasReceivedInformation = True
234                                     activeSendersUpdate.append(neighbor)
235                                     if total_informed == nbrOfNodes:
236                                         breakSimulation = True
237                                         break
238         informedResults[generations].append(nbrOfInformed)
239         activeSenders = (activeSenders + activeSendersUpdate).copy()
240         for i in range(nbrOfNodes):
241             Nodes[i].regenerateStatus()
242 
```

```
meanINSIGNIFCANT = []
ct = 0
for i in range(timeLimit):
    tempRecipients = simulationResults.get(i + 1)
    if len(tempRecipients) > 1:
        if i + 1 - ct > 1:
            sys.stdout.write("Error in simulation")
            break
        ct = i + 1
        tempMean = statistics.mean(tempRecipients)
        meanINSIGNIFCANT.append(tempMean)

mean_info_unimportant = []
ct = 0
for i in range(timeLimit):
    tempRecipients = informedResults.get(i + 1)
    if len(tempRecipients) > 1:
        if i + 1 - ct > 1:
            sys.stdout.write("Error in simulation")
            break
        ct = i + 1
        tempMean = statistics.mean(tempRecipients)
        mean_info_unimportant.append(tempMean)

# RUNS SIMULATION FOR ARBITRARY NODES
simulationResults = dict()
for i in range(timeLimit):
    simulationResults[i+1] = list()

informedResults = dict()
for i in range(timeLimit):
    informedResults[i+1] = list()

run = 0
while run < nbrOfSimulations:
    print(run)
    run += 1
    for i in range(nbrOfNodes):
        Nodes[i].hasReceivedInformation = False
        Nodes[i].setStatus()

    initiator = random.choice(populationSorted)

```

```
286     Nodes[initiator].hasReceivedInformation = True
287     Nodes[initiator].isSusceptible = True
288     activeSenders = list([initiator])
289     generations = 0
290     total_informed = 1
291     breakSimulation = False
292     while total_informed < nbrOfNodes:
293         nbrOfInformed = 0
294         generations += 1
295         simulationResults[generations].append(total_informed)
296         activeSendersUpdate = []
297         for i in activeSenders:
298             if Nodes[i].isSusceptible:
299                 if Nodes[i].willTransmit():
300                     for neighbor in Nodes[i].neighbors:
301                         if Nodes[neighbor].isSusceptible:
302                             if not Nodes[neighbor].hasReceivedInformation:
303                                 if Nodes[neighbor].willReceive():
304                                     total_informed += 1
305                                     nbrOfInformed += 1
306                                     Nodes[neighbor].hasReceivedInformation = True
307                                     activeSendersUpdate.append(neighbor)
308                                     if total_informed == nbrOfNodes:
309                                         breakSimulation = True
310                                         break
311         informedResults[generations].append(nbrOfInformed)
312         activeSenders = (activeSenders + activeSendersUpdate).copy()
313         for i in range(nbrOfNodes):
314             Nodes[i].regenerateStatus()
315
316 meanARBITRARY = []
317 generations = []
318 ct = 0
319 for i in range(timeLimit):
320     tempRecipients = simulationResults.get(i + 1)
321     if len(tempRecipients) > 1:
322         generations.append(i + 1)
323         if i + 1 - ct > 1:
324             sys.stdout.write("Error in simulation")
325             break
326         ct = i + 1
327         tempMean = statistics.mean(tempRecipients)
328         meanARBITRARY.append(tempMean)
```

```
329
330 mean_info_ARBITRARY = []
331 ct = 0
332 for i in range(timeLimit):
333     tempRecipients = informedResults.get(i + 1)
334     if len(tempRecipients) > 1:
335         if i + 1 - ct > 1:
336             sys.stdout.write("Error in simulation")
337             break
338         ct = i + 1
339         tempMean = statistics.mean(tempRecipients)
340         mean_info_ARBITRARY.append(tempMean)
341
342 mean_info_important.insert(0, 1)
343 mean_info_unimportant.insert(0,1)
344 mean_info_ARBITRARY.insert(0,1)
345
346 plt1, = plt.plot(meanimportant, '+-', linewidth = 1, markersize = 2.25)
347 plt2, = plt.plot(meanINSIGNIFCANT, 'o-.', linewidth = 1, markersize = 1)
348 plt3, = plt.plot(meanARBITRARY,  '^-', linewidth = 1.2, markersize = 2.5)
349 plt4, = plt.plot(mean_info_important, '*-', linewidth = 1, markersize = 2.5)
350 plt5, = plt.plot(mean_info_unimportant,  '--', linewidth = 1, markersize = 2.5)
351 plt6, = plt.plot(mean_info_ARBITRARY, '^-', linewidth = 1, markersize = 2.5)
352
353 plt.grid(color='black', alpha=1, linestyle='dashed', linewidth = 0.25)
354 strIMP = "Avg. informed - initiator is important"
355 strNONIMP = "Avg. informed - initiator is unimportant"
356 strABTR = "Avg. informed - initiator is arbitrary"
357 strIMP_info = "Avg. newly informed at time instant - important "
358 strNONIMP_info = "Avg. newly informed at time instant - unimportant "
359 strABTR_info = "Avg. newly informed at time instant - arbitrary"
360 plt.legend([plt1, plt2, plt3, plt4, plt5, plt6],
361 [strIMP, strNONIMP, strABTR, strIMP_info, strNONIMP_info, strABTR_info])
362 plt.xlabel('Time', fontsize=18)
363 plt.ylabel('Average per ' + str(int(nbrOfSimulations)) + ' simulations',
364 fontsize = 18)
365 plt.show()
366
367 plt1, = plt.plot(meanimportant, '+-', linewidth = 1, markersize = 2.25)
368 plt2, = plt.plot(meanINSIGNIFCANT, 'o-.', linewidth = 1, markersize = 1)
369 plt3, = plt.plot(meanARBITRARY,  '^-', linewidth = 1.2, markersize = 2.5)
370 plt.grid(color='black', alpha=1, linestyle='dashed', linewidth = 0.25)
371 strIMP = "Avg. informed - initiator is important"
```

```
372 strNONIMP = "Avg. informed - initiator is unimportant"
373 strABTR = "Avg. informed - initiator is arbitrary"
374 plt.legend([plt1, plt2, plt3], [strIMP, strNONIMP, strABTR])
375 plt.xlabel('Time', fontsize=18)
376 plt.ylabel('Average per ' + str(int(nbrOfSimulations)) + ' simulations',
377 fontsize = 18)
378 plt.show()
```

Master's Theses in Mathematical Sciences 2023:E5
ISSN 1404-6342

LUTFMS-3464-2023

Mathematical Statistics
Centre for Mathematical Sciences
Lund University
Box 118, SE-221 00 Lund, Sweden

http://www.maths.lu.se/