2023

# Applications of Centrality Measures and Extremal Combinatorics

Hunter Dane Rehm
*University of Vermont*

# Applications of Centrality Measures and Extremal Combinatorics

A Dissertation Presented

by

Hunter Rehm

to

The Faculty of the Graduate College

of

The University of Vermont

In Partial Fulfillment of the Requirements
for the Degree of Doctor of Philosophy
Specializing in Mathematical Sciences

May, 2023

Defense Date: March 22nd, 2023
Dissertation Examination Committee:

Puck Rombach, Ph.D., Advisor
Peter Dodds, Ph.D., Chairperson
Chris Danforth, Ph.D.
James Bagrow, Ph.D.
Cynthia J. Forehand, Ph.D., Dean of the Graduate College

# Abstract

Centrality measures assign numbers or rankings to network nodes that reflect their importance. There are many types of centrality measures, each suitable for different types of networks and applications. In Chapter 2, we consider a model of astronaut health during a space mission. Katz centrality is commonly used to measure the influence of nodes in social and biological networks. We motivate its use in this application to estimate the expected quality time lost due to the progression of medical conditions. In Chapter 3, we find dominating sets in satellite networks. To do this, we use the Shapley value, a centrality measure that originates in game theory and is computed based only on local network information. We demonstrate that the Shapley value is an effective and time-efficient tool for finding small dominating sets in various random graph families and a set of real-world networks. In Chapter 4, we introduce a novel algorithm for categorizing which nodes are nearest the boundary, called boundary nodes, in a network that uses Chvátal's definition of a line in a graph. We test this algorithm on random geometric graphs and compare its effectiveness to other known methods for boundary node detection.

In Chapter 5, for certain sets $S$ and equations $eq$, we look for the smallest number of colors $\text{rb}(S, eq)$ such that for every surjective $\text{rb}(S, eq)$-coloring of $S$, there exists a solution to $eq$ where every element of the solution set is assigned a distinct color. We show that $\text{rb}([n], x_1 + x_2 = x_3) = \lfloor \log_2(m) + 2 \rfloor$ and $\text{rb}([m] \times [n], x_1 + x_2 = x_3) = m + n + 1$ for $m, n \geq 2$. In Chapter 6, a graph $G$ is $H$-semi-saturated if adding an edge $e$ to $G$ that is not currently in $G$ causes $H$ to appear as a subgraph in $G$ that contains $e$. We say that $G$ is $H$-saturated if $G$ does not contain $H$ as a subgraph before adding $e$. The smallest number of edges in an $H$-semi-saturated ($H$-saturated) graph is called the semi-saturation number of $H$ (saturation number of $H$). We show that the saturation and semi-saturation numbers differ by at least 1 for a disjoint union of paths called a linear forest. Additionally, we find graph families for which the saturation number is monotonic with respect to the subgraph relation.

# Acknowledgements

I want to express my deepest gratitude to my advisor, Dr. Puck Rombach, for supporting me throughout my time at UVM. I could not have completed this journey without her. This endeavor would not have been possible without everyone who advised and supported me at the NASA Glenn Research Center, especially Dr. Robert Kassouf-Short, Dr. Mona Matar, and Lauren McIntyre. Thanks to my committee members, Dr. James Bagrow, Dr. Chris Danforth, and Dr. Peter Dodds, for their valuable feedback on this work.

Words cannot express my gratitude to my undergraduate advisor, Dr. Nathan Warnberg. He took me under his wing early on in my undergraduate and motivated me to be the mathematician I am today. Thanks to my co-authors and friends, Kean Fallon, Colin Levaunt, Ethan Manhart, Joe Miller, Simon Wagner, and Laura Zinnel. In addition, I am grateful for the helpful discussion with my fellow graduate students, especially Marcus Elia, Calum Buchanan, and Anton Hilado.

I would be remiss in not mentioning my best friends, Sadegh Khodabandeh, Ben Price, and Johnny Ramsdell. Thank you so much for the endless joy and laughs. Lastly, I want to thank my Mom, Dad, Grandma, and Sister for their continued love and support, especially in the hard times.

# Table of Contents

# List of Figures

# List of Tables

# List of Symbols

| | |
|---|---|
| $C(\alpha)$ | $\alpha$-Katz centrality, 14 |
| $C(\alpha, \ell)$ | $(\alpha, \ell)$-Katz parameter, 14 |
| $A$ | adjacency matrix, 5 |
| AR | anti-Ramsey number, 63 |
| $B(v)$ | betweenness centrality of node $v$, 6 |
| $C(v)$ | closeness centrality of node $v$, 6 |
| $CC(v)$ | clustering coefficient of node $v$, 5 |
| $K_n$ | complete graph on $n$ vertices, 81 |
| $\deg(v)$ | degree of node $v$, 5 |
| $k_v$ | degree of node $v$, 5 |
| $d(u, v)$ | distance between $u$ and $v$, 6 |
| $\gamma(G)$ | domination number of a graph $G$, 31 |
| $E$ | edge set, 4 |
| $endpoints(u, v)$ | endpoints of $L(u, v)$, 53 |
| $\mathrm{ICC}(v)$ | inverse clustering coefficient of $v$, 36 |
| $G + H$ | join of $G$ and $H$, 88 |
| $\alpha$ | Katz parameter, 7 |
| $L(u, v)$ | line containing $u$ and $v$, 52 |
| $[m] \times [n]$ | $m \times n$ grid, 74 |
| $\delta(G)$ | minimum degree in $G$, 32 |
| ND | neighbor degree, 6 |
| $N(v)$ | neighborhood of $v$, 4 |
| $P_n$ | path on $n$ vertices, 84 |

| | |
|---|---|
| $\mathrm{rb}(S, eq)$ | rainbow number of $S$, 64 |
| $\mathrm{sat}^*(n, H)$ | rainbow saturation number of $H$ on $n$ vertices, 89 |
| $R(r, b)$ | Ramsey number for integers $r$ and $b$, 61 |
| RND | relative neighbor-degree centrality, 36 |
| $\mathrm{stress}(v, \delta)$ | restricted stress centrality, 51 |
| $\mathrm{sat}(n, H)$ | saturation number of $H$ on $n$ vertices, 81 |
| $\mathrm{sat}(n, \mathcal{F})$ | saturation number of $\mathcal{F}$ on $n$ vertices, 83 |
| $S(c)$ | Schur number, 62 |
| $\mathrm{ssat}(n, \mathcal{F})$ | semi-saturation number of $\mathcal{F}$ on $n$ vertices, 83 |
| $\mathrm{ssat}(n, H)$ | semi-saturation number of $H$ on $n$ vertices, 81 |
| SV | Shapley value, 5 |
| $\rho$ | spectral radius, 7 |
| $\mathrm{stress}(v)$ | stress centrality, 51 |
| ex | Turán number, 80 |
| $L_{\alpha,\epsilon}(N)$ | true length, 15 |
| $V$ | vertex set, 4 |

# Part I

# Network Theory: Applications of Centrality Measures

# Chapter 1

# Introduction

## 1.1 Overview

Networks are everywhere and appear in many scientific contexts, such as biology, social sciences, math, computer science, engineering, communication, health, and data science [1–4]. For example, a group of people and their social relationships may be viewed as a network. More precisely, a *network* is a set of objects, called *nodes*, connected by *links* that join pairs of nodes. Network analysis broadly describes a set of tools that are used to model and analyze systems based on their network structure. For example, community detection algorithms look for a natural division of a network into dense clusters to describe its high-level structure [5–7]. Such structure can then be used to map the network, classify nodes or better understand the interaction between network structure and dynamics, such as disease spread in a social network.

Another prevalent topic in network analysis, and the focus of Part I of this dissertation, is the problem of measuring the importance of each node in a network [1,8,9]. There are numerous ways of defining and measuring importance, called *centrality* *measures*, and it all depends on the application. For example, the number of links attached to a node could indicate its importance. Further importance can be given to a node if the nodes it is connected to have many connections themselves, and so on. This behavior can be captured by what is called *eigenvector centrality* [10] and is related to how Google measures the importance of its web pages [11]. Alternatively, an application may be such that we deem a node important if it is a part of many shortest paths between other nodes. This idea is captured by *betweenness centrality.* These two examples only scratch the surface of the vast number of measures available.

Some centrality measures only use information from the immediate neighborhood of a node, called a *local* centrality measure. If a measure requires information from the entire network, it is called *global.* In Section 1.2 and 1.3 we define many local and global centrality measures, respectively, used in the literature and throughout Part I of this dissertation. In future chapters, we describe uses for a few of these centrality measures.

Chapter 2 applies the study of centrality to a medical application. The Medical Extensible Dynamic Probabilistic Risk Assessment Tool (MEDPRAT) developed by NASA is an event-based risk modeling tool that assesses human health and medical risk during space exploration missions. One of its key features is the ability to represent and simulate the relationships between medical events. The Susceptibility Inference Network (SIN) captures these relationships in an internal data structure. We develop a model to capture the nuances of the SIN and assess the medical events using Katz centrality to understand their risk of progressing.

In Chapter 3, we worked with the Space Communications and Navigation team at NASA to create and develop a local algorithm for determining critical satellites in the communication network. This algorithm was created for dynamic satellite communication data (Starlink, Iridium, etc.) but has applications elsewhere. We test the performance of our algorithm on a variety of random graphs.

Chapter 4 describes a novel approach for identifying the points nearest the boundary of a shape using limited data on their relative proximity. We ultimately designed and implemented an algorithm in Python and tested it on simulated data.

These three applications are far apart regarding underlying processes, network models, and types of questions asked. Despite these differences, centrality measures are an indispensable tool in each case. These applications will showcase the importance of choosing the right centrality measure for the model and the problem at hand.

## 1.2 Local Centrality Measures

First, we define more precisely what we mean by a graph, or network. A *graph* is a tuple $G = (V, E)$, where $V$ is a collection of objects, called *nodes* or *vertices*, and $E \subseteq V \times V$ is a set of node pairs, called *edges*. If the edges are ordered pairs, each edge has a direction, and $G$ is called a *directed graph*.

A centrality measure $c$ is *local* if $c(v)$ only requires information from vertices $u$ such that $(u, v) \in E$; such vertices $u$ are said to be *adjacent* to $v$ and the set of all adjacent nodes is called the *neighborhood of* $v$, denoted $N(v)$. The number of vertices in $N(v)$ is called the *degree* or *degree centrality* of $v$, which is the quintessential example of a local centrality measure. If $u$ is adjacent to $v$, the $(u, v)$th entry of the *adjacency* *matrix*, typically denoted $A$, is 1; hence the degree of $v$ is

$$\deg(v) = k_v = \sum_{u \in V} A_{uv}.$$

Two notions of degree, or degree centrality, exist in a directed graph: the *in-degree* $k_v^+$ and *out-degree* $k_v^-$ centrality,

$$k_v^+ = \sum_{u \in V} A_{uv} \quad \text{and} \quad k_v^- = \sum_{u \in V} A_{vu}.$$

For example, the in-degree in the network of citations often determines a paper's influence, which represents the number of articles that cite it. The remainder of the section describes a few other local centrality measures: the clustering coefficient, Shapley value, and neighbor-degree centrality.

The *clustering coefficient* [12] of a node $v$ is

$$CC(v) = \frac{2t_v}{k_v(k_v - 1)} = \frac{t_v}{\binom{k_v}{2}}.$$

Here, $t_v$ is the number of triangles that node $v$ is a part of. The clustering coefficient measures how close the neighbors of a node are to forming a clique. Since being introduced in 1998, it has been used in brain networks [13], urban train line networks [14], and more to establish a measure of local connectivity [15].

The *Shapley value* (SV), initially defined in game theory [16], was introduced as a network centrality measure by Bozzo et al. [17]. We use a version defined in [18], namely.

$$S(v) = \sum_{u \in N(v)} \frac{1}{\deg(u)}.$$

A node with a high Shapley value is likely adjacent to many low-degree nodes. This definition of the Shapley value is simple and, in our case, instrumental when constructing dominating sets (see Chapter 3). Outside of this, the value appears many times in the literature [19–22].

Ai et. al. define the *neighbor-degree centrality* (ND) of $v$ by $\mathrm{ND}(v) = \frac{\sum_{u \in N(v)} k_u}{k_i}$ [23]. The value $\mathrm{ND}(v)$ is the average degree of the neighbors of $v$. We define a relative version of the neighbor-degree centrality in Chapter 3.

## 1.3 Global Centrality Measures

A *global centrality measure* requires the entire network to compute it. Here, we describe some of a few well-known global centrality measures.

The *closeness centrality* ($C$) [24–26] of a vertex $v$ is defined as

$$C(v) = \frac{n-1}{\sum_u d(u,v)}$$

where $d(u,v)$ is the length of a shortest path between $u$ and $v$ and $n$ the number of vertices in the graph. The closeness centrality helps us identify the vertices at the smallest average distance from other vertices in the network and has been used to study lexical processing and social networks [27, 28].

The *betweenness centrality* ($B$) [29] of a vertex $v$ is defined as

$$B(v) = \sum_{u \neq v \neq w} \frac{\sigma_{uw}(v)}{\sigma_{uw}}$$

where $\sigma_{uw}(v)$ is the number of shortest paths starting from vertex $u$ and ending at vertex $w$ passing through $v$ and $\sigma_{uw}$ is the total number of shortest paths from $u$ to $w$. This centrality measure gives a high score to the vertices contained in a large number of shortest paths. It has been used to estimate the interdisciplinary influence of a journal and to study traffic patterns [30, 31].

The *eigenvector centrality*, initially proposed by Bonicach [10], is defined as the solution $\vec{x}$ to the well-known equation $A\vec{x} = \rho\vec{x}$ where $A$ is the adjacency matrix of $N$, and $\rho$ is the largest eigenvalue of $A$. An alternative definition is

$$\vec{x}_u = \rho^{-1} \sum_{v \in V} \vec{x}_v A_{uv}. \tag{1.1}$$

Eigenvector centrality assigns more importance to a node if its neighbors are themselves important. The unique eigenvector corresponding to $\rho$ is guaranteed by the Perron-Frobenius theorem, which states that every square matrix with non-negative real entries will have a real, non-negative eigenvalue $\rho$ that is greater than or equal to all other eigenvalues [32, 33]. Consequently, the corresponding eigenvector $\vec{x}$ has real, non-negative entries, which defines the eigenvector centrality.

The *spectral radius* $\rho$ of $N$ (or $A$) is the maximum modulus of the eigenvalues of $A$, and a *walk of length* $k$ from node $u$ to $v$ is a sequence of $k$ edges $(v_i, v_{i+1}) \in E$, $i \in [1, k]$ such that $v_1 = u$ and $v_{k+1} = w$. The *Katz centrality* developed by Leo Katz in 1953 [34], counts the number of all possible walks of any length going from one vertex to another. When accounting for a walk, this centrality incorporates the weight of every edge traversed and an additional factor $\alpha$, called the *Katz parameter*, for each additional step in the walk. In short, for $\alpha \in \left(0, \frac{1}{\rho}\right)$, Katz centrality is defined by the matrix-vector product

$$\Big((I - \alpha A)^{-1} - I\Big) \cdot W,$$

where $I$ is the $n \times n$ identity matrix, and $W$ is the $n \times 1$ vector of non-negative vertex weights. The result of this computation is a vector of *Katz scores*. We mention predicting neuronal activity in neuroscience from the numerous successful applications of Katz centrality. The vertex importance in this setting is due to the influence of a firing neuron on the firing of adjacent neurons and the decaying effect on more distant neurons [35].

| Name | Notation | Formula | Locality | Applications |
|---|---|---|---|---|
| Degree Centrality | $k_v$ | $\sum_{u \in V} A_{uv}$ | Local | [10] |
| Local Clustering Coefficient [12] | $CC(v)$ | $\frac{2t_v}{k_v(k_v-1)} = \frac{t_v}{\binom{k_v}{2}}$ | Local | [36] |
| Neighbor-Degree Centrality [23] | $\mathrm{ND}(v)$ | $\frac{\sum_{u \in N(v)} k_u}{k_i}$ | Local | [23] |
| Shapley Value [16–18] | $\mathrm{SV}(v)$ | $\sum_{u \in N(v)} \frac{1}{\deg(u)}$ | Local | [19–22] |
| Closeness Centrality [24–26] | $C(v)$ | $\frac{n-1}{\sum_u d(u,v)}$ | Global | [27, 28] |
| Betweenness Centrality [29] | $B(v)$ | $\sum_{u \neq v \neq w} \frac{\sigma_{uw}(v)}{\sigma_{uw}}$ | Global | [30, 31] |
| Eigenvector Centrality [10] | $\vec{x}_v$ | $\vec{x}_v = \rho^{-1} \sum_{u \in V} \vec{x}_u A_{vu}$ | Global | [35, 37, 38] |
| $\alpha$-Katz Centrality [34] | $C(\alpha)_v$ | $(I - \alpha A)^{-1} = \left(\sum_{k=0}^{\infty} \alpha^k A^k\right)$ | Global | [35] |

*Table 1.1: Centrality Measures*

# Chapter 2

# The use of Katz centrality for a medical application

The work presented in this chapter results from a collaboration with Dr. Mona Matar, Lauren McIntyre, and Dr. Puck Rombach, most of which is documented in [39].

## 2.1 Introduction

Networks are structures that naturally appear in every aspect of life and are studied in a wide range of disciplines from sociology, biology, and engineering [1, 40–42]. One common question in network theory is how to rank the nodes according to importance, where importance can have many meanings depending on the application [43]. Many ranking algorithms are based on a, possibly weighted, count of walks in which a node is contained. Examples of such centrality measures are degree, betweenness [29], closeness [24–26], eigenvector [10, 44], PageRank [45], the Estrada index [46], and Katz centrality [34]. We argue that the latter is suitable for our application, hence the focus of this paper.

We look at a model component developed by the National Aeronautics and Space Administration's (NASA) Human Research Program (HRP) called the Susceptibility Inference Network (SIN). This network represents the probability that simulated medical conditions may occur and progress to subsequent, clinically related conditions. When integrated with MEDPRAT, these relationships will produce results that are a more appropriate analog to the real-world medical system than the standing assumption that conditions are probabilistically independent. The SIN is currently a prototype, as the data used to inform it does not have the necessary credibility required by NASA standards 7150.2D and 7009A for use in decision support tools [47, 48]. Given the time and financial costs associated with evidence collection at this scale, there is significant motivation to focus those efforts towards conditions, or groups of conditions, whose relationships most influence medical risk outcomes.

*Katz centrality*, developed by Leo Katz in 1953 [34], has been used in numerous applications [35, 49]. The Katz centrality of a node is a weighted count of all walks of any length starting at the node. Each walk of length $k$ is weighted by $\alpha^k$, where $\alpha$ is called the *Katz parameter*. We formally define Katz centrality in Section 2.1.1. Since the Katz parameter has a decaying effect, we can approximate the Katz centrality by ignoring the contribution of walks past a given length $L$. In [50] and [51], the authors numerically explore this approximation type. In Section 2.2, we give a lower bound on this value $L$ (in terms of $\alpha$) that guarantees a desired level of accuracy in terms of the Katz centrality and its node ranking.

The network we model we use in this section is a *Bayesian network*, where the nodes are variables, and their edges $(u, v)$ are weighted the conditional probability

$\mathbb{P}(v|u)$. We specifically use a *dynamic Bayesian network* (DBN), which is a Bayesian network where the probabilities depend on discrete time steps [52]. For every time step $t$, each edge is weighted by $\mathbb{P}(v_{t+1}|u_t)$. We consider a DBN of medical conditions where each link $(u, v)$ is weighted by the probability that condition $v$ occurs at time $t + 1$ given condition $u$ occurs at time $t$. This network is called the Susceptibility Inference Network (SIN). Each node in the SIN is weighted by the quality time one can expect to lose if they get that condition, called the *Quality Time Lost* (QTL).

This paper is organized as follows. Section 2.1.1 reviews useful graph theory concepts, definitions, and basic results. Here, we introduce the $\epsilon$-agreement of two centrality measures, which indicates their agreement regarding node rankings given an assumed $\epsilon$ margin of error in their node centrality scores. Section 2.1.2 introduces the SIN data set, which is the application of interest. In Section 2.2, we bound the error generated from approximating the Katz centrality by restricting the number of steps allowed in a walk, and develop a relationship between that number and the Katz parameter $\alpha$. An example of the relationship between $\alpha$ and the $\alpha$-Katz centrality node ranking is in Section 2.2 and our medical application in Section 2.3. Additionally, we assess the upper bound given in Section 2.2 to the true length in Section 2.4. Finally, the results and future work are addressed in Section 2.5.

### 2.1.1 Definitions

This section provides basic definitions for the graph-theoretical structures and tools used for the results in Section 2.2. We restate many of the definitions needed for this chapter here.

**Definition 1. (Weighted, directed network)** *Let $N = (V, E, w)$ be an edge-weighted, directed network consisting of $V$, the set of $n$ nodes, $E \subseteq V \times V$, the set of edges, and a weight function $w : E \to \mathbb{R}^+$.*

*We represent such a network by an adjacency matrix $A = A(N)$, where the entry $A_{ij}$ is the weight of the edge from node $i$ to node $j$, or $A_{ij} = 0$ if there is no edge from $i$ to $j$ in $N$. Let $W$ be an $n$-dimensional vector of non-negative node weights. In a setting where edges and/or nodes are unweighted, weights in $A$ and $W$ are set to 1.*

For example, in our application in Section 2.3, our weighted, directed network has nodes that represent medical conditions, and the node weights represent their severity, while edge weights represent the probability of one medical condition progressing to another.

As a reminder, the *spectral radius* $\rho$ of $N$ (or $A$) is the maximum modulus of the eigenvalues of $A$. A *walk of length $k$* from node $u$ to $v$ is a sequence of $k$ edges $(v_i, v_{i+1}) \in E$, $i \in [1, k]$ such that $v_1 = u$ and $v_{k+1} = w$. The *distance* from $u$ to $v$ is the length of a shortest walk from $u$ to $v$. The *$k$-hop neighborhood* of a node $v \in V$ is the set of nodes at a distance less than or equal to $k$ from $v$, denoted $N_k(v)$. A *centrality measure* is a function that assigns a real number to each node, to evaluate its relative importance to other nodes. Each centrality measure gives a (partial) *ranking* of the nodes, which reflects their relative importance.

Here, we define Katz centrality, a parameterized centrality measure whose parameter $\alpha$ takes in walk length considerations.

**Definition 2. (Katz centrality)** *Let $N$ be an edge-weighted, directed network with node weights $W$. Let $A = A(N)$ with spectral radius $\rho$, and let $\alpha \in (0, 1/\rho)$. The*

*$\alpha$-Katz centrality vector [34, 53, 54], is defined as*

$$C(\alpha) = \left( \sum_{i=1}^{\infty} \alpha^k A^k \right) \cdot W = \left( (I - \alpha A)^{-1} - I \right) \cdot W.$$

*The $\alpha$-Katz score of a particular node $i$ can then be expressed as*

$$C(\alpha)_i = \sum_{k=1}^{\infty} \sum_{j=1}^{n} W_j \alpha^k \left( A^k \right)_{ij}.$$

*The $(\alpha, \ell)$-Katz centrality vector [55–57] is*

$$C(\alpha, \ell) = \left( \sum_{i=1}^{\ell} \alpha^k A^k \right) \cdot W$$

*and for a particular node $i$, the $(\alpha, \ell)$-Katz score can be written as*

$$C(\alpha, \ell)_i = \sum_{k=1}^{\ell} \sum_{j=1}^{n} W_j \alpha^k \left( A^k \right)_{ij}.$$

Both $C(\alpha)$ and $C(\alpha, \ell)$ measure the *downstream* influence of nodes since they are weighted sums over outgoing walks. Replacing the matrix $A$ by its transpose $A^T$ reverses edge directions, taking weighted sums over incoming walks instead and measuring the *upstream* influence [1, 53].

Definition 3 provides a tool to compare centrality measures purely in terms of their relative node rankings. Intuitively, we may set a threshold $\epsilon$ for a centrality measure $C$, such that $|C_i - C_j| \geq \epsilon$ implies that $C$ provides a relative ranking of nodes $i$ and $j$. If $|C_i - C_j| < \epsilon$, we cannot reliably recover a ranking from $C$. For two centrality measures $C$ and $C'$, we compare their rankings and conclude that they agree on a ranking if they agree for every node pair where both rankings are reliable.

**Definition 3. ($\epsilon$-agreement)** *Let $N$ be a weighted, directed network, $\epsilon, \epsilon' > 0$, and $C$ and $C'$ be centrality measures. The nodes $i, j \in V(N)$ are $(\epsilon, \epsilon')$-properly ranked with respect to $C$ and $C'$ if the following holds:*

1. *$|C_i - C_j| < \epsilon$ or $|C'_i - C'_j| < \epsilon'$,*

2. *otherwise, $C_i - C_j$ and $C'_i - C'_j$ have the same sign.*

*We say that $C$ and $C'$ $(\epsilon, \epsilon')$-agree on $N$ if every pair of nodes in $N$ is $(\epsilon, \epsilon')$-properly ranked with respect to $C$ and $C'$. If $\epsilon = \epsilon'$, we simply say $\epsilon$-proper ranking and $\epsilon$-agreement.*

**Definition 4.** *Let $N$ be an edge-weighted, directed network, $\epsilon > 0$, $\alpha \in (0, 1/\rho)$. We let*

$$L_{\alpha,\epsilon}(N) = \min\{\ell \mid C(\alpha) \textit{ and } C(\alpha, \ell) \ \epsilon\textit{-agree on } N\}.$$

*When the parameters are clear from the context, we will let $L = L_{\alpha,\epsilon}(N)$.*

**Proposition 1.** *Let $C$ and $C'$ be two centrality measures on a network $N$. If*

$$\|C - C'\|_\infty < \epsilon,$$

*then $C$ and $C'$ $\epsilon$-agree.*

*Proof.* Suppose for the sake of contradiction that $C$ and $C'$ do not $\epsilon$-agree. Then, there exists a pair of nodes $u, v \in V(N)$ that is not $\epsilon$-properly ranked. By part (1) of Definition 3, we have $|C_u - C_v|> \epsilon$ and $|C'_u - C'_v|> \epsilon$. By part (2), without loss of generality, we have $C_u - C_v < 0$ and $C'_u - C'_v > 0$.

We have

$$C'_u - C_u = \underbrace{C'_u - C'_v}_{>\epsilon} + \underbrace{C'_v - C_v}_{>-\epsilon} + \underbrace{C_v - C_u}_{>\epsilon} > \epsilon,$$

which contradicts that $\|C - C'\|_\infty < \epsilon$. □

**Proposition 2.** *Let $C$ and $C'$ be two centrality measures on a network $N$ such that for all $v \in V(N)$, $0 \leq C_v - C'_v < 2\epsilon$, then $C$ and $C'$ $\epsilon$-agree.*

*Proof.* Suppose for the sake of contradiction that $C$ and $C'$ do not $\epsilon$-agree. Then, there exists a pair of nodes $u, v \in V(N)$ that is not $\epsilon$-properly ranked. By part (1) of Definition 3, we have $|C_u - C_v|> \epsilon$ and $|C'_u - C'_v|> \epsilon$. By part (2), without loss of generality, we have $C_u - C_v < 0$ and $C'_u - C'_v > 0$.

We have

$$C_u - C'_u = \underbrace{C_u - C_v}_{<-\epsilon} + \underbrace{C_v - C'_v}_{<2\epsilon} + \underbrace{C'_v - C'_u}_{<-\epsilon} < 0,$$

which contradicts that $C_u - C'_u \geq 0$. □

In Section 2.2, we use this notion to compare two closely related centrality measures, $C(\alpha)$ and $C(\alpha, \ell)$, and we therefore only use $\epsilon$-proper ranking and $\epsilon$-agreement. However, we state the definition here in a more general form. It can be used to compare any pair of centrality measures, even if their distributions of values differ significantly. The vector $C(\alpha, \ell)$ converges to $C(\alpha)$ as $\ell \to \infty$. In Theorem 1, we show that this implies that for all $\epsilon > 0$, there exists an $L$ so that for any $\ell > L$, $C(\alpha)$ and $C(\alpha, \ell)$ $\epsilon$-agree.

### 2.1.2 Susceptibility Inference Network

The Medical Extensible Dynamic Probabilistic Risk Assessment Tool (MEDPRAT) developed by NASA is an event-based risk modeling tool that assesses human health and medical risk during space exploration missions [58, 59]. One of its key features is the ability to represent and simulate the relationships between medical events. These relationships are captured in an internal data structure called the Susceptibility Inference Network (SIN).

The SIN is a directed network where nodes represent medical conditions. The data in this network is subject matter expert informed and is currently a prototype. There is an edge from $u$ to $v$ if medical condition $u$ can progress into medical condition $v$. This directed edge $(u, v)$ is weighted by the probability of such a progression. Note that a medical condition may progress to multiple other conditions simultaneously, or to no other conditions. Therefore, this matrix is not a transition matrix. The SIN currently has 99 nodes and 1078 edges. Medical conditions included are, for example, acute radiation syndrome, which has many outgoing edges towards other medical conditions. On the other hand, anxiety has many incoming edges.

This expert-informed data does not contain information about the time it takes for progressions. As a simplified model, we view the SIN as a Dynamic Bayesian Network [52]. Each node in the SIN has an associated weight that evaluates the severity of the condition regardless of the progression from or to that condition. This severity of a condition is quantified by Quality Time Lost (QTL), which we call *primary QTL* of that condition. The primary QTL measures the productive time a crew member is expected to lose and is equal to the $i$th entry $W_i$ of the weight vector

$W$.

In our condition progression networks, each edge $(i, j)$ is weighted by $A_{ij}$, which is the probability that condition $j$ is present at time $t + 1$ given that condition $i$ is present at time $t$. Then, this edge contributes $A_{ij}W_j$ to $\mathbb{E}[\mathrm{QTL}_i]$. We assume that progressions occur independently, and therefore a path of length two from $i$ to $j$ via a node $k$ contributes $A_{ik}A_{kj}W_j$ to $\mathbb{E}[\mathrm{QTL}_i]$. Under the assumption of uninterrupted progressions, we have, in general,

$$\mathbb{E}[\mathrm{QTL}_i] = \sum_{k=0}^{\infty} \sum_{j=1}^{n} W_j \left(A^k\right)_{ij} = W_i + C(1)_i.$$

Note that $W_i$ is the primary QTL of node $i$ and $C(1)_i$ the subsequent QTL under the assumption of uninterrupted progression. We highlight a few subtleties in this model. Note that we allow two types of cycles in our network: there may be multiple directed paths from a condition $i$ to a condition $j$, in which case $j$ contributes multiple times to the total expected QTL of $i$. There may also be directed cycles, wherein a condition $i$ contributes multiple ways to its total expected QTL. We motivate this in terms of the application later. First, we consider an illustration. Table 2.1 contains examples of small condition progression networks.

To make this estimate more realistic, we consider that the progression of conditions will likely be interrupted by medical interventions and time constraints on the mission. The parameter $\alpha$ provides a damping factor that decreases the weight of walks as they get longer. This application, therefore, illustrates the importance of using realistic walk lengths to guide the choice of $\alpha$. We provide a theoretic foundation for this in Section 2.2. In Section 2.3, we discuss how different values of $\alpha$ produce different rankings for the SIN due to subsequent QTL.

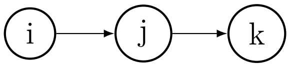
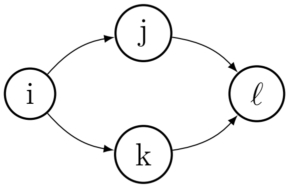
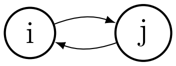
| Condition Progression Network | $\mathbb{E}(\mathrm{QTL}_i)$ |
|---|---|
|  | $W_i + A_{ij}W_j + A_{ij}A_{jk}W_k$ |
|  | $W_i + A_{ij}W_j + A_{ik}W_k + (A_{ij}A_{j\ell} + A_{ik}A_{k\ell})W_\ell$ |
|  | $\sum_{k=0}^{\infty}(A_{ij}A_{ji})^k W_i + A_{ij}\sum_{k=0}^{\infty}(A_{ij}A_{ji})^k W_j = \frac{W_i+A_{ij}W_j}{1-A_{ij}A_{ji}}$ |

*Table 2.1: Three networks with the expected QTL of node $i$.*

## 2.2 The Katz parameter and Walk Length

This section describes the relationship between maximum walk lengths $\ell$ and the parameter $\alpha$. In Theorem 1, we find a lower bound on $\ell$ that guarantees that $C(\alpha)$ and $C(\alpha, \ell)$ $\epsilon$-agree. This sheds light on the length of walks that decide the ranking provided by $C(\alpha)$. We provide a small, illustrative example of the effect of $\alpha$ on node rankings.

Lemma 1 gives an upper bound on the difference between values in $C(\alpha)$ and $C(\alpha, \ell)$.

**Lemma 1.** *(Absolute Error Tolerance) Let $p \in \{1, 2, \infty\}$ and $\alpha \in (0, 1/\rho)$. Then*

$$\|C(\alpha) - C(\alpha, \ell)\|_{\infty} \leq (\alpha \|A\|_p)^{\ell} \, \|C(\alpha)\|_p := \epsilon_{\ell}.$$

*Proof.* First, note that $\|V\|_{\infty} \leq \|V\|_2 \leq \|V\|_1$ for all vectors $V$. Furthermore, the $p$-norm is sub-multiplicative. We have

$$\begin{aligned}\|C(\alpha) - C(\alpha, \ell)\|_\infty &\leq \|C(\alpha) - C(\alpha, \ell)\|_p \\ &= \left\| \left( \sum_{i=\ell+1}^{\infty} \alpha^k A^k \right) \cdot W \right\|_p \\ &= \left\| \alpha^\ell A^\ell \left( \sum_{i=1}^{\infty} \alpha^k A^k \right) \cdot W \right\|_p \\ &\leq \alpha^\ell \|A^\ell\|_p \left\| \left( \sum_{i=1}^{\infty} \alpha^k A^k \right) \cdot W \right\|_p \\ &\leq \left(\alpha \|A\|_p\right)^\ell \|C(\alpha)\|_p .\end{aligned}$$

□

In Lemma 2, we show that there exists an $L$ so that for all $\ell > L$, the difference between the scores in $C(\alpha)$ and $C(\alpha, \ell)$ is small.

**Lemma 2.** *(Error Tolerance Guarantee) Let $p \in \{1, 2, \infty\}$, $\alpha \in (0, 1/\|A\|_p)$ and $\epsilon > 0$. If*

$$\ell > \log_{\alpha\|A\|_p} \left( \frac{2\epsilon}{\|C(\alpha)\|_p} \right) := L_{up}$$

*then $\|C(\alpha) - C(\alpha, \ell)\|_\infty < 2\epsilon$.*

*Proof.* Note that $\alpha < 1/\|A\|_p \leq 1/\rho$. By Lemma 1, it suffices to show that $\epsilon_\ell < 2\epsilon$ when $\ell > L_{up}$. We have

$$\begin{aligned}
\epsilon_\ell &= \|C(\alpha)\|_p \left(\alpha\|A\|_p\right)^\ell \\
&< \|C(\alpha)\|_p \left(\alpha\|A\|_p\right)^{L_{up}} \\
&= \|C(\alpha)\|_p \left(\alpha\|A\|_p\right)^{\log_{\alpha\|A\|_p}\left(\frac{2\epsilon}{\|C(\alpha)\|_p}\right)} \\
&= \|C(\alpha)\|_p \frac{2\epsilon}{\|C(\alpha)\|_p} \\
&= 2\epsilon.
\end{aligned}$$

□

Lemma 2 bounds the difference between $C(\alpha)$ and $C(\alpha, \ell)$ when $\ell$ is large enough. Theorem 1 shows that this also ensures that $C(\alpha)$ and $C(\alpha, \ell)$ $\epsilon$-agree.

**Theorem 1.** *Let $p \in \{1, 2, \infty\}$, $\alpha \in (0, 1/\|A\|_p)$, $\epsilon > 0$ and $L_{up}$ be as in Lemma 2. If $\ell > L_{up}$ then $C(\alpha)$ and $C(\alpha, \ell)$ $\epsilon$-agree.*

*Proof.* By Lemma 2 we have that $\|C(\alpha) - C(\alpha, \ell)\|_\infty < 2\epsilon$. Therefore, by Proposition 2, we have that $C(\alpha)$ and $C(\alpha, \ell)$ $\epsilon$-agree. □

We choose $\epsilon$ so that two nodes with Katz scores within $\epsilon$ of each other can be considered equivalent in terms of ranking. Of course, such a value must be chosen relative to the Katz scores. In Corollary 1, we suggest letting $\epsilon$ be some fraction of $\|C(\alpha)\|_p$, and $\alpha$ a fraction of $1/\|A\|_p$ for $p \in \{1, 2, \infty\}$.

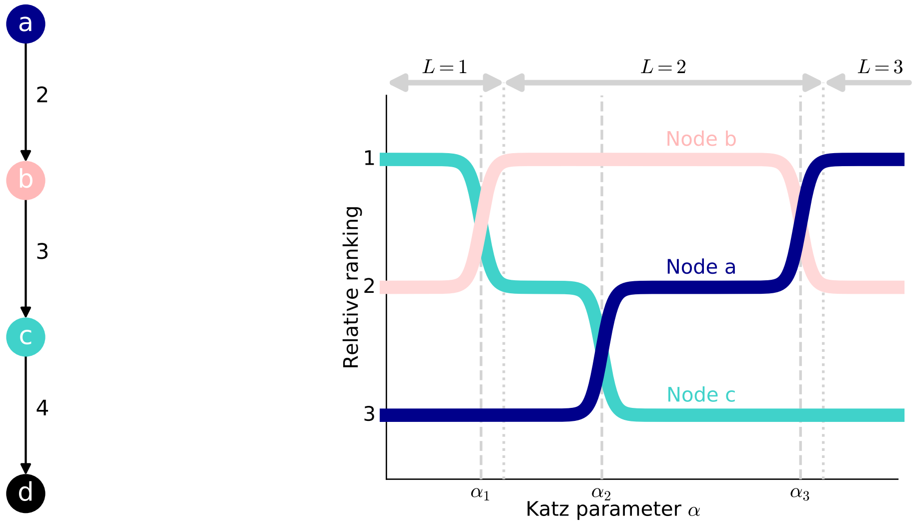
**Corollary 1** (Relative Error)**.** *Let $p \in \{1, 2, \infty\}$ and $\alpha_0, \epsilon_0 \in (0, 1)$. For $\alpha = \alpha_0/\|A\|_p$, and $\epsilon = \epsilon_0\|C(\alpha)\|_p$, if $\ell > \log_{\alpha_0}(\epsilon_0) = L_{up}$ then $C(\alpha)$ and $C(\alpha, \ell)$ $\epsilon$-agree.*

We present a small example illustrating the effect of $\alpha$ on node rankings. Consider the edge-weighted directed network $N$ in Figure 2.1a. Let $\alpha_1$ be the positive solution to $C(\alpha)_b = C(\alpha)_c$, let $\alpha_2$ be the positive solution to $C(\alpha)_a = C(\alpha)_c$, and let $\alpha_3$ be the positive solution to $C(\alpha)_a = C(\alpha)_b$. When $\alpha$ is small, the walks of length 1 determine the ranking. As $\alpha$ increases, walks of length 2 and later walks of length 3, become more important. Node $c$ has the most walks of length 1, node $b$ the most of length 2, and node $a$ the most of length 3, and they are each ranked on top for different ranges of $\alpha$.

Figure 2.1b also shows the value of $L = L_{\alpha,\epsilon}(N)$ as a function of $\alpha$. For example, at $\alpha_1$, the walks of length 2 become significant enough for node $b$ to overtake node $c$ in the ranking. Therefore, $L$ switches from 1 to 2. This switch happens at a value of $\alpha$ slightly greater than $\alpha_1$, once the difference in scores of $b$ and $c$ has exceeded $\epsilon$. Note that the ranking switch of node $a$ and $c$ at $\alpha_2$ is also due to paths of length 2 gaining significance and does not cause a change in $L$.

## 2.3 Application to the Susceptibility Inference Network

We use $\alpha$-Katz centrality to rank medical conditions in the SIN by their expected subsequent QTL. We inspect the distribution of the differences between Katz scores of nodes and set $\epsilon$ so that 5% of the differences are less than the chosen $\epsilon$. This implies that 95% of the pairs are ranked correctly. We run the analysis for $\alpha = 1$ and $\alpha = 0.4$. When $\alpha = 1$, $(\alpha, \ell)$-Katz centrality and $\alpha$-Katz centrality $\epsilon$-agree when $\ell \geq 5$, and when $\alpha = 0.4$, $(\alpha, \ell)$-Katz centrality and $\alpha$-Katz centrality $\epsilon$-agree



*(a) An example of a weighted directed network.*

*(b) The node rankings from the Katz scores of the network in Figure 2.1a.*

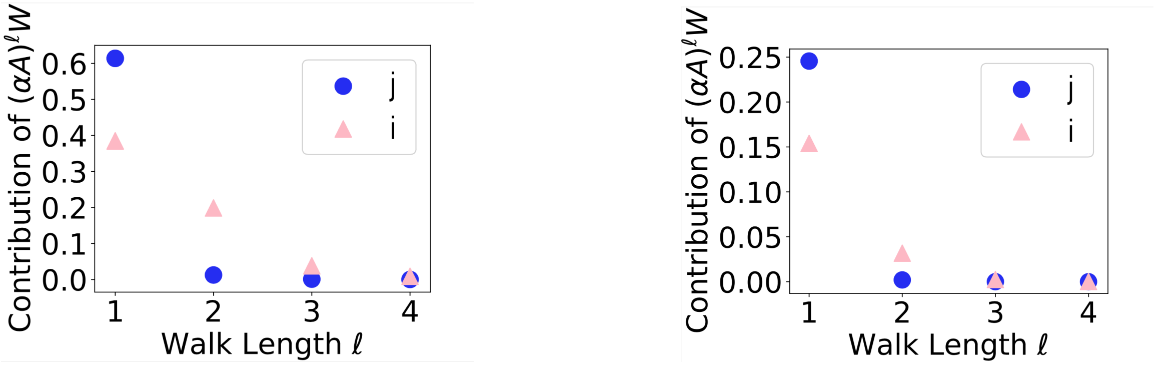
*Figure 2.1: The relationship between the Katz parameter $\alpha$ and the vertex ranking in a directed, edge-weighted example network.*

when $\ell \geq 3$. Table 2.2 shows a comparison of rankings of the top 10 conditions, and Figure 2.3 illustrates the subnetwork of the SIN containing the 12 nodes that appear in Table 2.2.

We note that one significant change in the ranking as $\alpha$ decreases from 1 to 0.4 is the one that swaps the positions of nodes $i$ and $j$. Figure 2.2 plots the contributions of walks of length $\ell$ to the Katz scores of these two nodes. When $\alpha = 1$, the contribution of walks of lengths 2 and 3 contribute enough to the score of node $i$ to place it above $j$. When $\alpha = 0.4$ these longer walks contribute significantly less, lowering the ranking of node $i$ to fall below that of $j$. In this and other applications, there is no universal value of $\alpha$ that is best. Here, one would have to consider the length of the space mission and other factors that affect how realistic any number of progressions is.

| Rank | $\alpha = 1$ | $\alpha = 0.4$ |
|---|---|---|
| 1 | $a$ | $a$ |
| 2 | $b$ | $b$ |
| 3 | $c$ | $d$ |
| 4 | $d$ | $c$ |
| 5 | $e$ | $e$ |
| 6 | $f$ | $f$ |
| 7 | $g$ | $g$ |
| 8 | $h$ | $j$ |
| 9 | $i$ | $h$ |
| 10 | $j$ | $l$ |

Table 2.2: The effect of the choice of $\alpha$ on the ranking of medical conditions in the SIN due to subsequent QTL.



*(a)* $\alpha = 1$

*(b)* $\alpha = 0.4$

*Figure 2.2: Contribution of walks of length* $\ell$ *to the scores of nodes* $i$ *and* $j$ *in* 1*-Katz centrality (a) and* 0.4*-Katz centrality (b).*

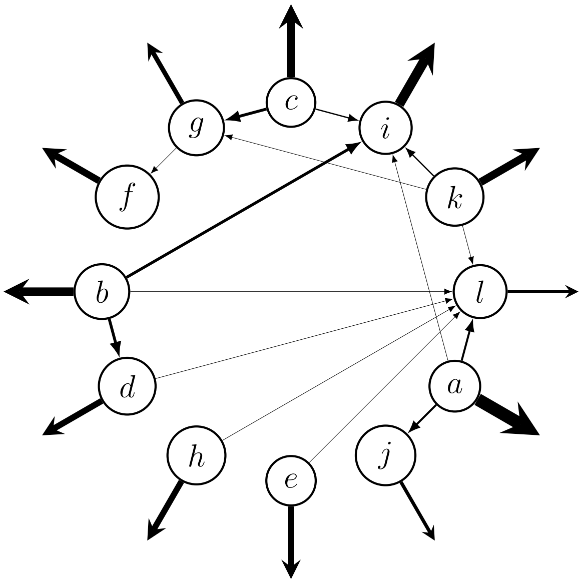


*Figure 2.3: Subnetwork of the SIN with 12 most influential nodes with weighted edges. The thickness of the edges that point outwards illustrates the total outgoing edge weight towards nodes in the remainder of the network.*

## 2.4 Testing the Upper Bound on Simulated Data

We would like to better understand for which values of $\ell$ that $C(\alpha)$ and $C(\alpha, \ell)$ $\epsilon$-agree. Theorem 1 gives an upper bound on $L_{\alpha,\epsilon}(N)$. In Figure 2.4, we compare the upper bound $L_{up}$ to $L_{\alpha,\epsilon}(N)$ on two families of undirected graphs. At each instance of $\alpha_0$ in Figures 2.4a and 2.4b, we sample 10 graphs from each specified graph family and plot the average upper bound and average $L_{\alpha,\epsilon}(N)$ across the samples.

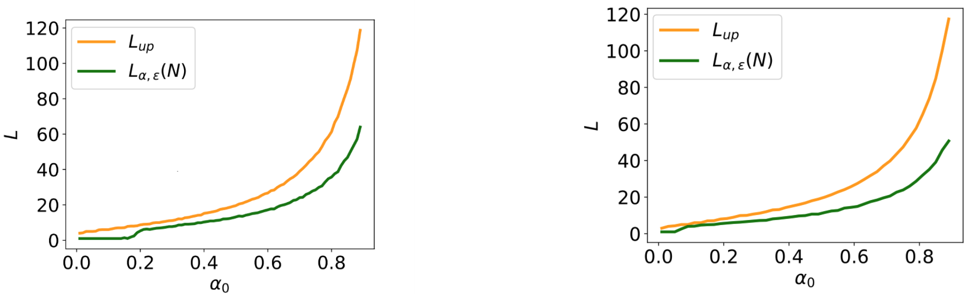
We use two different random graph models. The *Erdős-Rényi model* creates $n$ points $V$ and includes an edge between two points $u, v \in V$ if with probability $p$ to form the edge set $E$. We say that $G = (V, E)$ is sampled from $\mathcal{G}(n, p)$, denoted by $G \sim \mathcal{G}(n, p)$. The *Chung-Lu* model creates $n$ points and a length $n$ vector $D = (d_1, d_2, \ldots, d_n)$ containing the expected degree of each node in the network. The pair $(u, v) \in V \times V$ is included in the edge set $E$ with probability

$$p_{uv} = \frac{d_u d_v}{\sum_i d_i}.$$

With these probabilities, the expected degree distribution of the network is $D$.

In Figure 2.4a, we sample from the Erdős-Rényi model $G(n, p)$ where $n = 1000$ and $p = 0.008$. In Figure 2.4b we sample from the Chung-Lu model, which inputs a list of node degrees. We individually sampled 1000 numbers from the negative binomial distribution to create this list. This distribution takes a probability $p$ of success and a number $n$ of desired successes. We use $p = 0.1$ as the probability of success, and $n = 3$ as the number of desired successes.

The Chung-Lu model, as described here, produces graphs with a longer-tailed degree distribution than graphs sampled from the Erdős-Rényi model. In Figure 2.4, the value for $\epsilon$ that we use is calculated using the same technique as in Section 2.3 for each iteration. We create a list of pairwise differences between the Katz scores and set $\epsilon$ so that 5% of the differences are less than it.



*(a) Erdős-Rényi.*

*(b) Chung-Lu.*

*Figure 2.4: Comparing the upper bound to $L_{\alpha,\epsilon}(N)$ using the Erdős-Rényi model $G(1000, 0.008)$ in (a) and the Chung-Lu model with degrees sampled from the negative binomial distribution in (b).*

## 2.5 Discussion and Conclusions

This paper introduced a tool to help compare different centrality measures. We applied this to $\alpha$- and $(\alpha, \ell)$-Katz centrality to help better understand the effect of the $\alpha$-parameter on the walk lengths considered when it comes to the ranking of nodes. For a given $\alpha$, we provide an upper bound on the walk length $L$ so that, when $\ell > L$, the Katz scores in $\alpha$- and $(\alpha, \ell)$-Katz centrality are within $2\epsilon$ of each other. We show that two nodes with both centrality measures differing by at least $\epsilon$ are in the same order in both rankings when $\ell > L$.

If we want to find a minimal value of $L$ such that $\alpha$-Katz and $(\alpha, \ell)$-Katz centrality

$\epsilon$-agree for all $\ell \geq L$, then Theorem 1 provides an upper bound. We can find a stronger upper bound by iteratively increasing $L$ until for all $\ell \geq L$, $\|C(\alpha) - C(\alpha, \ell)\| < 2\epsilon$. This guarantees $\epsilon$-agreement, although $\epsilon$-agreement may still happen sooner, so care should be taken when interpreting these bounds.

All of the results in this paper require $\alpha$ to be in $(0, 1/\|A\|_2)$, a subset of the possible values that can be used as a Katz parameter. It may be possible to extend these results to all Katz parameters, namely, values in $(0, 1/\rho)$. These ranges match for undirected graphs, which applies to directed, edge-weighted graphs.

We showed the effect of changing the $\alpha$ parameter in the Susceptibility Inference Network. In this case, changes in the ranking were visible even among the top 10 nodes, and they may have significant implications for decision-making. It is, therefore, important that the choice of $\alpha$ is made carefully and tailored to each application. This also holds for $\epsilon$.

The work presented in this paper will be leveraged to identify subsets of the prototype SIN that are expected to produce the largest effect in MEDPRAT and thus act as a scalable road map for future work, which will be focused on collecting and validating higher credibility clinical evidence. Given the significant cost associated with evidence collection, it is critical to narrow the scope and focus the effort where it will be most valuable, which is to say where interactions between conditions produce the largest change in spaceflight medical risk.

# Chapter 3

# Generating Dominating Sets Using Locally-Defined Centrality Measures

The work presented in this chapter results from a collaboration with Dr. Robert Short and Dr. Puck Rombach. The Vermont Space Grant Consortium supported this work.

## 3.1 Introduction

A *dominating set* in a network is a set of nodes so that every node in the dominating set is either in the set or has a neighbor in the set. Dominating sets and their variations have many applications [60–62], and usually one objective is to minimize the size of such a set.

Deciding whether or not a graph has a dominating set of size $k$ is NP-complete [63]. The current fastest exact algorithm for identifying a minimal dominating set was introduced in 2011 by Van Rooij and Bodlaender [64] and runs in $O(1.4969^n)$ time. Instead of identifying a dominating set globally, there are also distributed algorithms where decisions are made at the level of individual nodes with limited rounds of communications [65–70]. In most of these distributed algorithms, it is assumed that each node has full global information regarding the structure of the network, but only local information regarding the dominating set assignment. In some applications, nodes may have none or limited global information. A pertinent example of such an application are ad-hoc and wireless sensor networks [71–74].

The Space Communication and Navigation (SCaN) program office manages NASA's space communication activities. One of their goals is to understand satellite communication, including routing, radio technology, and network modeling. Satellites in a swarm may play different roles during a mission: sending information, taking pictures, storing data, and more. However, whether a satellite is best suited for a particular task at any time depends on its position relative to other satellites and the ground. For instance, if a satellite is well-positioned to communicate, its primary role should be sending or receiving information. We aim to have the satellites selected in these important roles to form a dominating set.

Finding a minimum dominating set in these types of applications is not feasible. Instead, the objective is to find an approximately minimum dominating set in a distributed manner where nodes can access only local information in the network.

Let $G(V, E)$ be a simple, undirected graph on $n$ nodes, and let $\gamma(G)$ be the *domination number* of $G$; the cardinality of a minimum dominating set. One of the most fundamental results in the theory of dominating sets is an upper bound in terms of the minimum degree $\delta(G)$ of $G$:

$$\gamma(G) \leq \frac{1 + \ln(1 + \delta(G))}{1 + \delta(G)} n. \tag{3.1}$$

This result was proved independently by [75–78]. The proof given in [75] has become a standard example of the power of the probabilistic method. It gives a simple, two-step probabilistic algorithm that yields a dominating set whose cardinality is in expectation the upper bound in Inequality 3.1. The algorithm works as follows. In the first round, each vertex is assigned to a set $X$ with a fixed probability $p$, independently of other vertices. In the second round, each vertex not dominated by $X$ is added to a set $Y$. The output is an approximately minimum dominating set $X \cup Y$. This algorithm is both fast, as it is linear in the size of the graph, and highly local, as decisions are made at the vertex level with communication only to immediate neighbors.

Wu and Li [70], proposed a similar algorithm with a deterministic first step. It uses the local clustering coefficient to select a dominating set [12]. This algorithm assigns a node $v$ to the dominating set if it has a clustering coefficient less than 1. In this case, a connected dominating set is guaranteed, and there is no need for a second round.

Inspired by this style of algorithm, we propose an algorithm for generating dominating sets with the help of local centrality measures. Our algorithm is not guaranteed to generate a minimum dominating set, but we show that it performs well in practice in finding small dominating sets. In the first step, the algorithm selects vertices based on a centrality measure meeting a certain threshold. In the second step, more vertices are added as needed to guarantee a dominating set.

We compare several different centrality measures on synthetic and real-world networks. The best-performing measure in our analysis is the Shapley value, introduced in game theory [16]. It is used in various contexts to measure the importance of an actor in game theory [17,18,20,79]. There are various definitions of the Shapley value in the literature. Here we use the definition from [18], which defines the Shapley value of a node as the sum of the reciprocals of the degrees of its neighbors. (These authors then subtract 1 from each value, which we will not do.)

We are particularly interested in applying this algorithm to *wireless ad hoc network* (WANET), which is a computer network that shares resources through wireless data connections, called links. We also apply it to a *mobile ad hoc network* (MANET), which is a WANET where each node moves freely, and the links change accordingly. These networks are decentralized and are often used for forwarding data. For example, a collection of satellites orbiting a planet form a MANET, called a satellite network.

In Section 3.2, we define the two-step algorithm and a few centrality measures that appear to effectively select small dominating sets. We compare their performance in the two-step algorithm on various synthetic and real-world networks in Section 3.3 and 3.4, respectively. Section 3.5 provides counterexamples to a few conjectures about the two-step algorithm from the simulations.

## 3.2 Dominating Set Algorithms using Thresholds

In Section 3.2.1, we outline the two-step algorithm explicitly. In Section 3.2.2, we define the centrality measures considered in this study with a few basic observations that motivate their use in this context.

### 3.2.1 The Two-step Algorithm

Let $c(v)$ be a locally computed node centrality measure on the node set of a network $V(G)$, and $\tau \geq 0$ a threshold constant. The two-step algorithm has two rounds: the first round selects all nodes $v$ such that $c(v) > \tau$, and the second round selects all nodes $v$ such that neither $v$ nor any of its neighbors were selected in the first round. The second step ensures that the final set of selected nodes forms a dominating set in $G$. As a reminder, $N_1(v) = \{u \in V | (u, v) \in E(G)\}$.

**Two-step Algorithm**

1. Let $X$ be the set of nodes $v$ with $c(v) > \tau$.
2. Let $Y$ be the set of nodes $v$ such that $(N_1(v) \cup \{v\}) \cap X = \emptyset$.
3. Let $D = X \cup Y$.

This algorithm can be executed in a distributed manner. To perform the first step, each node sends the necessary information to its neighbors so that each of them can calculate $c(v)$. Then, each node communicates to its neighbors whether they are in the set and carries out the second step. All of the measures we consider in Section 3.2.2 can be computed using one round of communication with the immediate neighborhood of each node, supposing each node knows its degree. However, local or not, any centrality measure can be used in the algorithm.

This algorithm performs best when $X$ is close to a dominating set so that $Y$ is small. Thus, the goal of the measure $c(v)$ is to capture nodes in $X$ likely to appear in a minimum dominating set. In Section 3.2.2, we define a few centrality measures of interest. We compare their performance in the two-step algorithm on various synthetic and real-world networks in Section 3.3 and 3.4, respectively.

### 3.2.2 Centrality measures

Since we want our two-step algorithm to be local, we use centrality measures that can be determined locally. Specifically, we consider centrality measures that depend only on the immediate neighborhood of a node.

**Uniform random measure (URM):** Alon investigates which threshold minimizes the expected size of the set constructed in the two-step algorithm and gives an upper bound on the domination number [80]. Refer to Section 3.1 for more information.

**Clustering coefficient (CC and ICC):** In [70], Li and Wu show that the collection of all nodes with two non-adjacent neighbors is a connected dominating set. This is quantified by finding the proportion of non-adjacent neighbors, known as the *clustering coefficient* [12] of a node $v$

$$CC(v) = \frac{t_v}{\binom{k_v}{2}} = \frac{2t_v}{k_v(k_v - 1)}.$$

Here, $t_v$ is the number of triangles that node $v$ is a part of, and $k_v$ is the degree of node $v$.

Based on simulated data, we suspect that the nodes representative of stronger dominating set candidates tend to have lower values of $CC(v)$. For this reason, we let

$$\mathrm{ICC}(v) = 1 - CC(v) = \frac{\binom{k_v}{2} - t_v}{\binom{k_v}{2}}.$$

It is worth noting that ICC is not a new centrality measure but rather our adjustment of the clustering coefficient that allows for more convenient comparisons with the other centrality measures, as they all tend to correlate positively with the likelihood of appearing in minimum dominating sets.

Using the language of our paper, Li and Wu show that the first step of two-step algorithms constructs a connected dominating set with measure ICC and a threshold of 0.

**Relative neighbor degree (**RND**):** Ai, Li, Su, Jiang, and Xiong define the neighbor-degree centrality of $v$ as $ND(v) = \frac{\sum_{u \in N_1(v)} k_u}{k_i}$ [23]. We propose the *relative neighbor-degree centrality* of $v$ as

$$\mathrm{RND}(v) = \frac{k_v}{ND(v)} = \frac{k_v^2}{\sum_{u \in N_1(v)} k_u}$$

which measures how the average degree of nodes in $N_1(v)$ compares to the degree of $v$ itself. If a node $v$ has a relative neighbor degree of 1, then the average degree of the neighbors of $v$ equals the degree of $v$. If $v$ has a relative neighbor degree larger than 1, then $v$ makes for a good dominating set candidate as its degree is larger than its neighbor's degree on average.

**Shapley value (**SV**) [16,18]:** The Shapley value was originally proposed in [16], but was not introduce in network theory until later. Bozzo, Franceschet, and Rinaldi describe the Shapley value of a node $v$ in a network as the sum of the reciprocals of the neighbors' degrees,

$$SV(v) = \sum_{u \in N_1(v)} \frac{1}{deg(u)}.$$

A node with a high Shapley value is likely adjacent to many low-degree nodes.

For the remainder of the paper, we compare all four measures and specifically look at how SV and RND, as they seem to perform best in this setting. From some simple analysis, these two measures are related. For instance, Proposition 3 shows that the number of nodes selected in the first round of the two-step algorithm with SV is always greater than that with RND.

**Proposition 3.** *For node $v$ in a network $N$, if* $\mathrm{RND}(v) \geq \tau$*, then* $\mathrm{SV}(v) \geq \tau$*.*

*Proof.* Suppose that $\mathrm{RND}(v) \geq \tau$. Then the average degree of a node in $N_v$ is less than $k_v/\tau$. By Jensen's inequality, the average of the set $\{1/k_u | u \in N_v\}$ is at least $\tau/k_v$. Hence, $\mathrm{SV}(v) = \sum_{u \in N_v} 1/k_u \geq k_v \frac{\tau}{k_v} = \tau$. □

We give the expected value of SV in Proposition 4 below. This proposition shows that even though SV is a locally computed measure, we know how each node will compare to the average without knowing anything about the network's structure. The expected value of RND is not as easily described.

**Proposition 4.** *Let $G = (V, E)$ graph. The expected Shapley value of a node $v \in V$ is* 1*.*

*Proof.* Let $A$ be the adjacency matrix of a network $N = (V, E)$ with $n = |V|$. Let $M$ be the stochastic matrix with $M_{ij} = \frac{A_{ij}}{k_j}$ and $\mathbf{1}$ be the $n$-dimensional vector with 1 in each entry. Notice that $(M \cdot \mathbf{1})_i = \mathrm{SV}_i$ for each $i \in V$. Since the stochastic matrix preserves the average of vectors, then the expected Shapley value is 1. □

From Proposition 4, $\tau = 1$ is a natural choice of threshold if we use SV in our two-step algorithm. To round out this section, we give additional evidence suggesting that $\tau = 1$ is a reasonable choice of threshold in a vacuum.

**Observation 1.** *If* $\deg(v) = 1$*, then* $v$ *is adjacent to a node* $u$ *with* $\mathrm{SV}(u) \geq 1$*.*

**Observation 2.** *If* $\deg(v) \geq \deg(u)$ *for all* $u \in N_1(v)$*, then* $\mathrm{RND}(v) \geq 1$*.*

According to Observation 1, a threshold of $\tau = 1$ will, among others, select all nodes adjacent to degree 1 nodes in the first round of the algorithm. We will see in Section 3.3 and 3.5, other thresholds better find minimal dominating sets using the two-step algorithm.

## 3.3 Simulated Data

We test our algorithm on simulated graphs generated from three random graph families.

One way to randomly generate a network is by way of a *random geometric graph*, $G(n, r, X)$ where $n$ points are sampled uniformly from an object $X \subset \mathbb{R}^2$ and an edge $(u, v)$ is added if the euclidean distance $\|u-v\|_2$ is less than $r$. Most often, $X = [0, 1]^2$. In Chapter 4, we create an algorithm that identifies the nodes nearest the boundary of the sample space $X$ and assess this algorithm on various shapes $X$, orders $n$, and radii $r$.

We also consider two other random graph models in this section: the *Erdős-Rényi* and *Chung-Lu* model. As a reminder, we will define them here. The *Erdős-Rényi model* creates $n$ points $V$ and includes an edge between two points $u, v \in V$ if with probability $p$ to form the edge set $E$. We say that $G = (V, E)$ is sampled from $\mathcal{G}(n, p)$, denoted by $G \sim \mathcal{G}(n, p)$. The *Chung-Lu* model creates $n$ points and a length $n$ vector $D = (d_1, d_2, \ldots, d_n)$ containing the expected degree of each node in the network. The pair $(u, v) \in V \times V$ is included in the edge set $E$ with probability

$$p_{uv} = \frac{d_u d_v}{\sum_i d_i}.$$

With these probabilities, the expected degree distribution of the network is $D$. We will use all three of these random graph models to test this algorithm.

We generate 100 networks for each family with 100 nodes and an expected degree of 10. Below we describe the families and present the computational results.

**Erdős-Rényi Model** [81] In Figure 3.1, we compare the average size of a dominating set created by the two-step algorithm using a variety of thresholds over instances of Erdős-Rényi graphs. We sample each graph using 100 nodes and a probability $p = 0.1$.

**Random Geometric Graphs** [82] In Figure 3.2, we compare the size of the dominating set found by the two-step algorithm with different threshold values in random geometric graphs on the unit square with 100 nodes and a radius $r = 0.178$. The motivation for using this radius is that the average degree $100(0.178)^2$ is about 10, consistent across the other random graph families.

**Chung-Lu Model** [83–85] We use the Chung-Lu model to sample graphs with a longer-tailed degree distribution and create the degree sequence using a negative binomial degree distribution. Figure 3.3 compares the size of the dominating set for each threshold on 100 different graphs sampled using the Chung-Lu model. We create the degree sequence using the negative binomial distribution with $n = 1$ and $p = 0.1$.

In each of Figures 3.1, 3.2, and 3.3, SV and RND construct the smallest dominating sets across all thresholds, with SV slightly outperforming RND. Notice that the minimal dominating set in each family typically occurs at a threshold greater than 1 rather than precisely at 1. Because SV generally performed best, we use the Shapely value in many of the applications in Section 3.4.

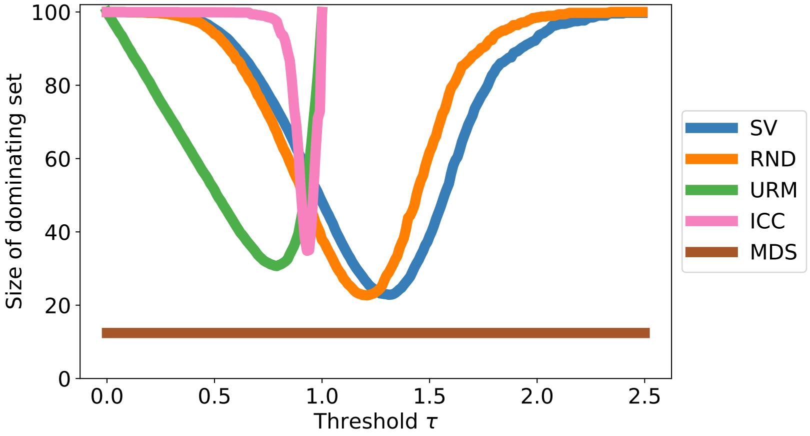


*Figure 3.1: Erdős-Rényi Model with 100 nodes and probability* 0.1*.*

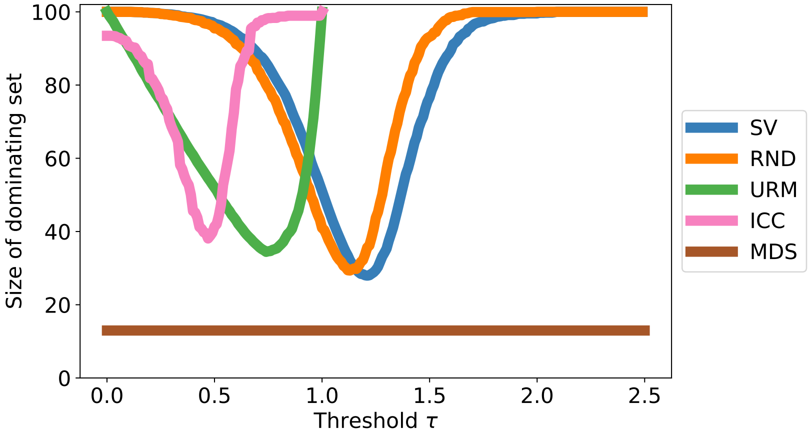


*Figure 3.2: A comparison of the centrality measures in the local domination algorithm for random geometric graphs* $G(100, 0.178)$*.*

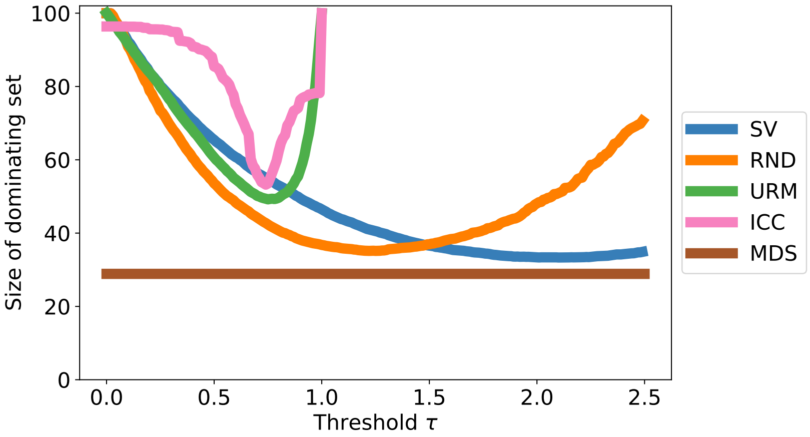


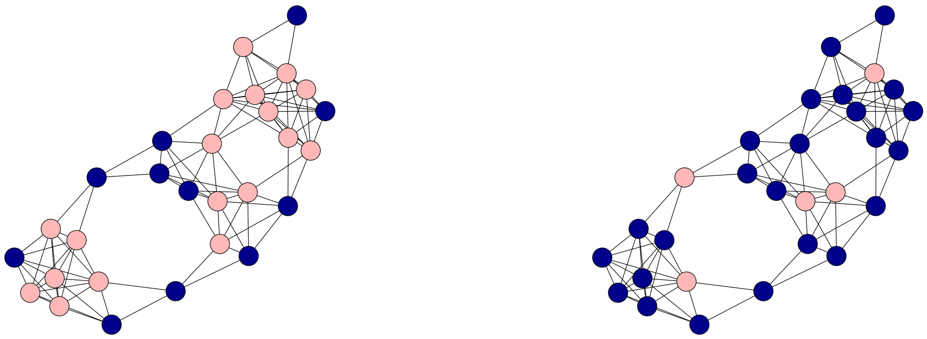
*Figure 3.3: Chung-Lu model with 100 nodes and a degree distribution sampled from a negative binomial distribution.*

## 3.4 Real World Data

We test the two-step algorithm using the Shapley value on two real-world networks: the Iridium Satellite network and the European Natural Gas Pipeline network. We first show how the algorithm performs with $\tau = 1$. The Iridium Satellite network shows which nodes are chosen in a real-world network by the algorithm, and that the algorithm finds a smaller dominating set when we use a different threshold.

### 3.4.1 Iridium Satellite Network

The Iridium satellites form a network in Low-Earth Orbit. Figure 3.4a indicates which nodes are selected in the two-step algorithm (in pink) with the Shapley value and threshold $\tau = 1$. Figure 3.4b shows the dominating set chosen by the two-step algorithm with the Shapley value when $\tau = 1.2$.



(a) $\tau = 1$

(b) $\tau = 1.2$

*Figure 3.4: Comparing thresholds in the two-step algorithm with* SV *centrality measure on an instance of the Iridium satellite network. The color pink indicates a node in the dominating set.*

### 3.4.2 European Natural Gas Pipeline Network

The International Energy Agency (IEA) data, analysis, and policy recommendations help countries provide a secure and sustainable future. They have collected data from 31 participating countries to create the European natural gas network (ENGN). We use the Shapley value in the two-step algorithm to find a dominating set. In Figure 3.5, the nodes colored pink are selected by the two-step algorithm with SV centrality measure and a threshold of $\tau = 1$.

Figure 3.6 is a subgraph of the ENGN where the Shapley value scales each node. This visual represents how the Shapley value can vary quite a bit from node to node.

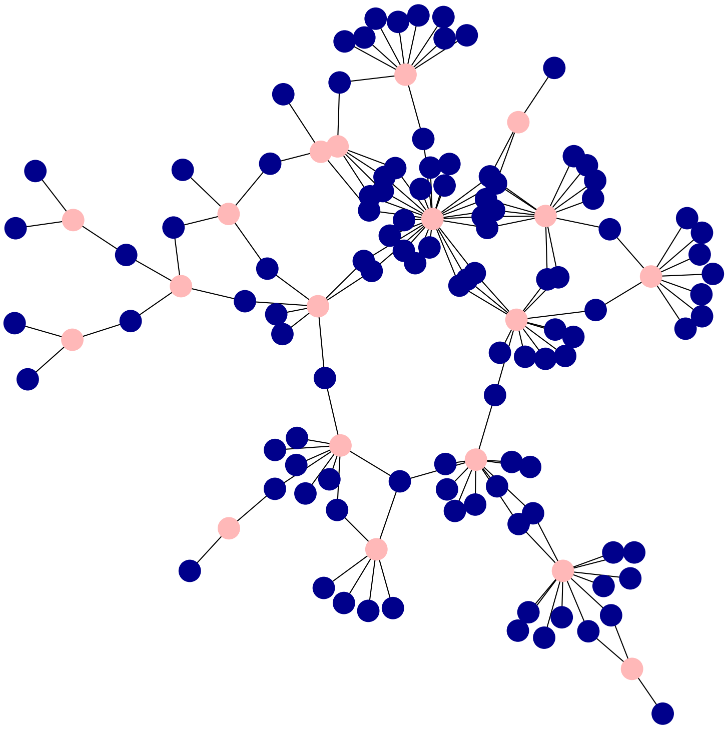


*Figure 3.5: European Natural Gas Pipeline Network with dominating set.*

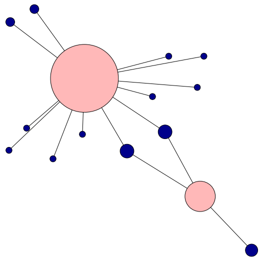


*Figure 3.6: Subgraph of the European Natural Gas Pipeline Network. Each node size is scaled to its Shapley value.*

## 3.5 Oddities

This section presents a set of counterexamples to natural conjectures regarding the two-step algorithm with the SV centrality measure. These counterexamples are enlightening, revealing exciting relationships between the measures we examined here and some general oddities surrounding the two-step algorithm.

With a threshold in the first filtering for the two-step algorithm, much of our questioning centered on choosing the best threshold. Inspired by Wu and Li [70], we initially sought a universal threshold for constructing a dominating set. If there were such a threshold for SV or RND, we could reduce our two-step algorithm to just one step: compute the measure and determine whether or not you are in the dominating set. If the threshold is 0, only one step is necessary, but perhaps a small non-zero threshold will also guarantee a dominating set in one step.

It turns out that there is no such non-zero threshold, but our construction to show this is unwieldy and unlikely to appear in graphs with relatively small numbers of nodes. We present this construction in the proof of the following proposition.

**Proposition 5.** *Given a threshold $\tau > 0$, let $S_\tau$ be the set of nodes such that* $\mathrm{SV}(v) > \tau$*. There exists a graph $G$ such that $S_\tau$ is not a dominating set of $G$.*

*Proof.* We construct a graph with node $a$ that satisfies the following properties:

1. For node $a$, $\mathrm{SV}(a) < \tau$.

2. For all nodes $b \in N_1(a)$, $\mathrm{SV}(b) < \tau$.

Given $\tau > 0$, there exists some $N_1 \in \mathbb{N}$ such that $\frac{1}{N_1} < \tau$. So, we can construct a set of $N_1$ nodes $\{b_i\}_{i=1}^{N_1}$ such that each $b_i$ is adjacent to $a$. Thus, $\deg(a) = N_1$, and $N_1(a) = \{b_i\}_{i=1}^{N_1}$.

Since $N_1, \tau > 0$, there exists $N_2 \in \mathbb{N}$ such that $N_2 > \frac{N_1}{\tau}$. For each $b_i$, construct $N_2 - 1$ nodes $\{c_{i,j}\}_{j=1}^{N_2-1}$ such that each is adjacent to $b_i$. Thus, $\deg(b_i) = N_2$, and $N_1(b_i) = \{a\} \cup \{c_{i,j}\}_{j=1}^{N_2-1}$ for each $i$. Then, by construction, $\mathrm{SV}(a) = \frac{N_1}{N_2} < \tau$.

Since $\frac{1}{N_1} < \tau$, $\tau - \frac{1}{N_1} > 0$, and since $N_2 \in \mathbb{N}$, $N_2 - 1 \geq 0$. Thus, there exists $N_3 \in \mathbb{N}$ such that $N_3 > \frac{N_2-1}{\tau - \frac{1}{N_1}}$. So, for each $c_{i,j}$, construct $N_3 - 1$ nodes $\{d_{i,j,k}\}_{k=1}^{N_3-1}$ such that each is adjacent to $c_{i,j}$. Thus, $\deg(c_{i,j}) = N_3$ for each $j$. So, by construction, for each $b_i$,

$$\mathrm{SV}(b_i) = \frac{1}{\deg(a)} + \sum_{j=1}^{N_2-1} \frac{1}{\deg(c_{i,j})} = \frac{1}{N_1} + \frac{N_2 - 1}{N_3} < \tau.$$

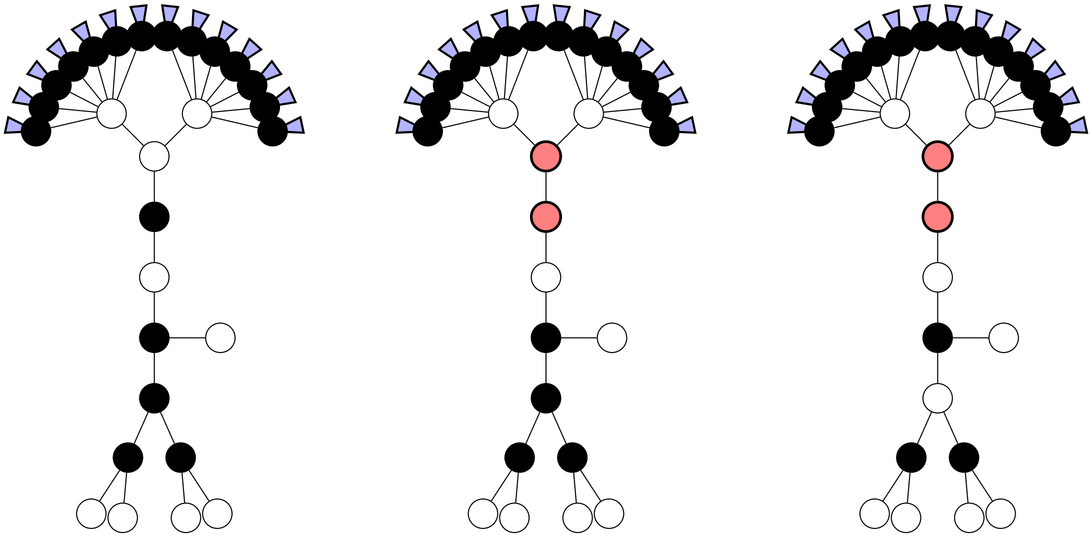
Thus, any graph containing this construction as a subgraph would satisfy the conditions above. □

From the construction in Proposition 5, and the relationship between RND and SV given in Proposition 3, we get the following result.

**Corollary 2.** *Given a threshold $\tau > 0$, let $R_\tau$ be the set of nodes such that $\mathrm{RND}(v) > \tau$. There exists a graph $G$ such that $R_\tau$ is not a dominating set of $G$.*

The choice of threshold then depends on the graph structure we are examining. However, in the plots we generated, the graphs appeared to have a consistent shape, including a single local minimum. For the minimum to be unique, one condition would be that for any three thresholds $\tau < \sigma < \rho$, one could never see $f(\tau) < f(\sigma) > f(\rho)$.

Unfortunately, this is not true in general. Figure 3.7 depicts one counterexample of this idea. In particular, when we use SV with the two-step algorithm at three different thresholds (5/6, 1, and 11/6), the size of the dominating set generated goes from 19 to 20 and back to 19.



*(a)* $\tau = 5/6$
$|D| = 19$

*(b)* $\tau = 1$
$|D| = 20$

*(c)* $\tau = 11/6$
$|D| = 19$

*Figure 3.7: The size of the dominating set formed by the two-step algorithm with the Shapley value does not achieve a unique local minimum. The purple triangles represent 100 leaf nodes. A black node suggests that it was chosen in the first round of the two-step algorithm, and red suggests it was chosen in the second round.*

We conclude our catalog of oddities by looking at how a threshold of 1 performs in SV and RND. Thanks to Proposition 4, we know that 1 holds a special place concerning the SV and RND values. Moreover, the following observation connects SV and RND to regularity.

**Observation 3.** *If $G$ is regular, then* $\mathrm{SV}(v) = \mathrm{RND}(v) = 1$ *for all* $v \in V$*.*

However, returning to the data represented in Figures 3.1, 3.2, and 3.3, there was one further counterexample to consider. When we set the threshold at 1, the dominating set generated with RND was generally larger than the dominating set developed with SV. However, Figure 3.8 shows an example where the number of nodes selected in the two-step algorithm is smaller in SV (Figure 3.8a) than in RND (Figure 3.8b) using the threshold $\tau = 1$. Here, the black nodes are chosen in the first round and the red nodes in the second.

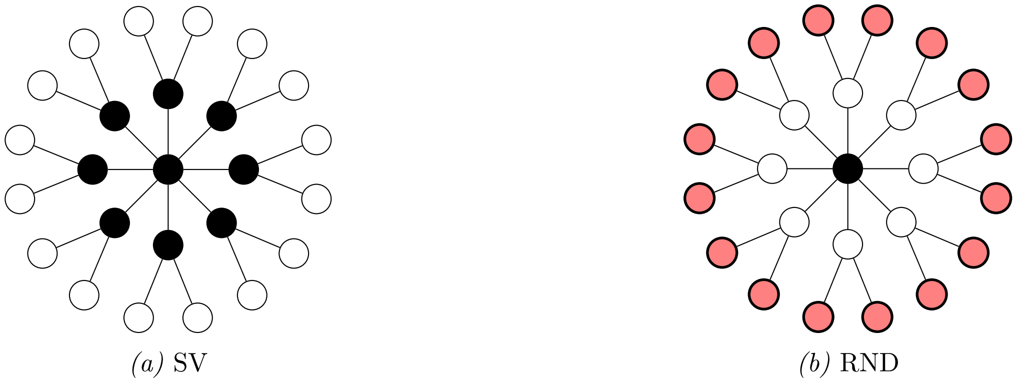


*Figure 3.8: An example where the dominating set generated using* RND *is greater than the dominating set generated using* SV*.*

There is undoubtedly more exploration in characterizing the function described in the plots. Both experimentally and theoretically, further investigation is warranted.

## 3.6 Discussion

Various algorithms aim to approximate a minimum dominating set for graphs [86]. Our approach of using a two-step algorithm allows for a larger variety of node measures than we propose for generating dominating sets. However, we compared four local measures to highlight some methods that could be practical in robot swarm and autonomous system communications.

We see several directions to take this work into the future. First, an analysis of different measures and their properties when plugged into the two-step algorithm may be an excellent method for comparing the effects of other measures. Included with this, a more in-depth theoretical analysis of the two-step algorithm is warranted to explore its effectiveness further.

One concept we have toyed with is constructing bounds for the thresholds in various situations. For example, one might wonder whether there are certain classes of networks that yield natural bounds, much like how Observation 3 holds for regular graphs. Predicting optimal thresholds based on network structure alone or time-varying network structure alone is a powerful tool in robotic mission design. Moreover, if the ability to find bounds analytically proves intractable, passing this information into a neural network may be an exciting problem to pursue. A Graph Neural Network that intakes graph structure and outputs a substantial threshold for achieving a dominating set could enable more optimizations, especially if this could be computed locally in some way.

# Chapter 4

# Boundary Detection

The work presented in this chapter is the result of a collaboration with Dr. Puck Rombach, with considerable guidance from Dr. Dominique Zosso.

## 4.1 Introduction

Many data sets have the form of a point cloud, where each data point represents a point in a feature space. When each data point is sampled from a bounded subset of the feature space, we try to understand the shape of the point cloud. Let $X \subset \mathbb{R}^k$ be a bounded subspace. Define an $(k-1)$-*ball* of radius $r$ centered at $p \in \mathbb{R}^k$ to be the set of points $x \in \mathbb{R}^k$ such that $\|p - x\| < r$ and let $S$ be the $(k-1)$-ball with the smallest radius that contains $X$. Suppose we sample points uniformly at random from $S$ and only take those contained in $X$. This chapter aims to understand the shape of $X$ by imposing a graph structure on the points we sample and using it to infer the boundary of $X$.

The boundary of an object $X$ is defined by the points on the closure of $X$ that are not in the interior of $X$ in $\mathbb{R}^k$, but a graph $G$ has no natural boundary. Hence, there is no agreed-upon definition of a boundary node, however many authors have created algorithms that identify them differently. We introduce an algorithm that categorizes each data point as either a boundary node or an interior node. In the remainder of this section, we will describe various methods already introduced in the literature.

In [87] and [88], the authors use the location of each point to determine which are nearest the boundary. The algorithm we propose only uses the graph structure to categorize the boundary nodes. The authors of [89–92] also only consider the graph structure in their algorithms, so we will describe those in more depth later in this section.

Each algorithm, including the one we introduce in this chapter, is tested on geometric graphs in Section 4.3. As a reminder, a *geometric graph* $G \sim \mathcal{G}(X, n, r)$ is a collection of $n$ points $V$ in a bounded subspace $X \subset \mathbb{R}^k$, where an edge connects two points $u, v \in V$ if $\|u - v\|_2 < r$. Note that $\mathcal{G}(X, n, r)$ is a distribution from which $G$ can be sampled, in which case $G$ is a random geometric graph. We use the notation $\mathcal{G}(n, r)$ if the bounded subspace $X$ is clear from the context.

Beghdad and Lamraoui in [89] propose an algorithm that starts by considering the neighborhood of a node $v$ in a graph $G$, denoted $N(v)$. Let $G[N(v)]$ be the induced subgraph of $G$ on the nodes in $N(v)$. If $G[N(v)]$ contains a maximal independent set of fewer than 3 nodes, then $v$ is categorized as a boundary node; otherwise, $v$ is an interior node. They repeat this process for every node.

Destino and Freitas de Abreu in [90] use the idea of clustering to categorize the boundary nodes in a graph. First, they partition the nodes into clusters. The nodes with the lowest betweenness centrality for each cluster are the *cluster boundary nodes*.

Then, they merge clusters that contain adjacent nodes. If a node labeled as a cluster boundary node is adjacent to other clusters, it is called an interior node. Otherwise, it is a boundary node.

Fekete et. al. [91] introduce two new centrality measures. Namely, they introduce stress centrality and restricted stress centrality. The *stress centrality* of a node $v$, denoted stress($v$) in a graph is defined as the number of shortest paths in $G$ containing $v$ over all pairs of endpoints. The *restricted stress centrality* stress($v, \delta$) is defined as the stress centrality of $v$ in the induced subgraph created by nodes in $N_\delta(v)$. In both cases, if a node's centrality measure is high, their algorithm assumes it is in the interior of $X$. If the measure is low, the algorithm assumes the node is a boundary node.

Li, et. al. proposed in [92] to identify a node $v$ as a boundary node if the betweenness centrality is larger than $\tau_b$ and the closeness is smaller than $\tau_c$. They investigate how well these two centrality measures perform when categorizing boundary nodes in geometric graphs. Another study by Giles, Georgiou, and Dettmann shows that betweenness centrality alone does well at predicting boundary nodes in dense random geometric graphs [93].

Huang, Wu, Gao, and Zhang in [94] construct a local boundary detection algorithm, with a 2-neighborhood of each node. In the first step, they say that if a node is not contained in a closed cycle within its 2-hop neighborhood, it is a boundary node candidate. In the second step, the set of nodes selected as candidates is refined by calculating the minimum hop count from a boundary node candidate to all non-boundary node candidates $f$. Finally, the nodes with maximal $f$ in each 2-hop neighborhood are chosen as boundary nodes. The algorithm's performance is tested on wireless sensor networks and is improved on lower-density graphs.

The remainder of this chapter describes a new boundary detection algorithm.

## 4.2 Boundary Detection using Lines in a Graph

This section contains our contribution, a geometrically motivated algorithm for boundary node detection. We will start by stating Chvátal's definition of a line in a graph.

**Definition 5.** *[95] Let $u$ and $v$ be nodes in $G$. A line $L(u, v)$ in $G$ is as follows: $L(u, v)$ is a set of nodes $w$ so that one of the three are satisfied*

1. $d(u, w) = d(u, v) + d(v, w)$
2. $d(v, w) = d(v, u) + d(u, w)$
3. $d(u, v) = d(u, w) + d(w, v)$

This definition holds for any metric space, including graphs. This is a natural generalization of a line in Euclidean space that uses the idea of betweenness originating as an axiom of Coxeter [96]. In a graph, we select the nodes which are furthest from either $u$ or $v$ in $L(u, v)$ to be the candidates for boundary nodes, called the endpoints.

**Definition 6.** *Let $L(u, v)$ be a line in a finite, simple graph $G$. The endpoints of $L(u, v)$ are the nodes in $L(u, v)$ of maximal distance from $u$ together with the nodes in $L(u, v)$ of maximal distance from $v$. We will denote this set of endpoints by* *endpoints*$(u, v)$. *The line centrality of a node* $v$ *is*

$$LC(v) = \sum_{u,w \in V} |endpoints(u, w) \cap \{v\}|.$$

Figure 4.1 gives an example of a line. The nodes in $E(u, v)$ are the boundary node candidates in our algorithm. This section outlines a deterministic approach for selecting which candidates are indeed boundary nodes.

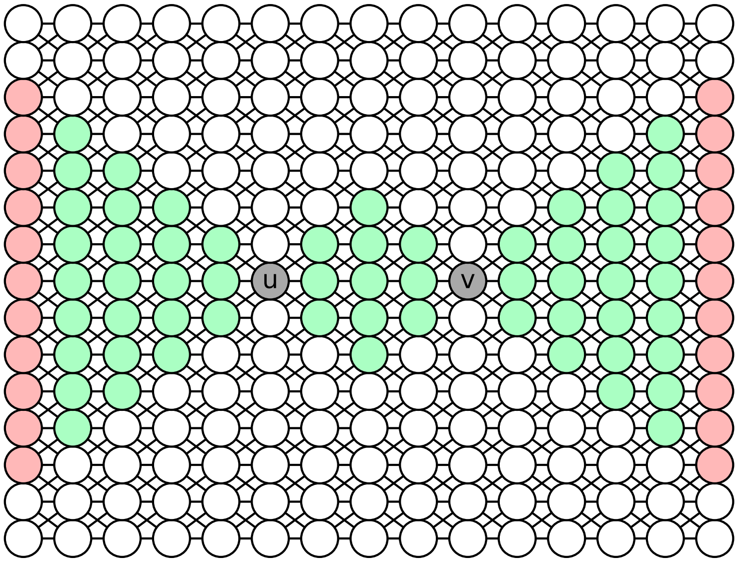


*Figure 4.1: Example of a line* $L(u, v)$*, where the green and red nodes are in* $L(u, v)$ *and* $E(u, v)$*, respectively.*

Algorithm 1, namely Endpoints$(G, u, v)$, finds the endpoints of a line through two nodes $u$ and $v$ in the graph and is used as a subroutine in Algorithm 2, namely LineCentrality$(G)$. LineCentrality$(G)$ runs through each pair of nodes in the graph and assigns to each node the number of times it was selected as an endpoint. Algorithm 3, called BoundaryDetection$(G, \tau)$, indicates that a node is a boundary node so long as it was chosen more than $\tau$ times in LineCentrality$(G)$. We often approximate the Line centrality for computational reasons by selecting only a subset of the node pairs.

Each algorithm can be used on any graph, but we test the algorithm on graphs where each node has a physical location. In Section 4.3 we test the effectiveness of Algorithms 3 on various sample spaces $X$ and graph densities. We test the performance of the line centrality for boundary detection by comparing it to the true distance from the boundary. We compare the performance of line centrality against other well-known centrality measures.

**Algorithm 1:** Endpoints(G,u,v)

```
Input: $G = (V, E)$ is a graph and $u, v \in V$;
Let $B$ be an empty set;
for $w$ in $L(u, v)$ do
    if $w$ is of maximal distance from $u$ or $v$ then
        add $w$ to $B$
    end
end
return $B$
```

**Algorithm 2:** LineCentrality(G)

```
Input: let $G = (V, E)$ be a graph;
Let $T$ be an array of size $|V|$;
for $u, v \in V$ do
    $P$ = Endpoints(G,u,v);
    for $i$ in $P$ do
        add 1 to the $i$th entry in $T$;
    end
end
return $T$
```

**Algorithm 3:** BoundaryDetection(G,$\tau$)

```
Input: let $G = (V, E)$ be a graph;
Let $B$ be an array;
$C$ = LineCentrality(G);
for $u$ in $V$ do
    if $C[u] \geq \tau$ then
        Append $u$ to $B$;
    end
end
return $B$
```

## 4.3 Application to Random Geometric Graphs

Every node in a random geometric graph was sampled from a bounded subspace $X$ of $\mathbb{R}^k$. We use the location information of each node to test the effectiveness of BoundaryDetection$(G, \tau)$ in Algorithm 3. In particular, we consider three different feature spaces $X$: a circle, a square, and a star. In Figures 4.2, 4.3, and 4.4, we create a random geometric graph $G \sim G(n, r, X)$ where $X$ is a circle, square, and star, and show which nodes are chosen as boundary nodes in BoundaryDetection$(G, \tau)$. We let $n = 300$, and set the radius $r$ so that the average degree is near 20.

We use our algorithm with betweenness centrality to select the corner nodes, which intuitively are the nodes nearest the corners of $X$. Of the nodes selected as boundary nodes in Algorithm 3, the ones with the lowest betweenness centrality are selected as *corner nodes*. These are displayed in Figure 4.5 as the blue nodes.

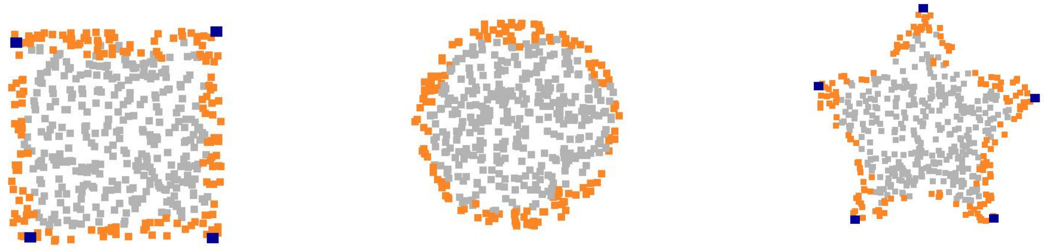


*Figure 4.2:* 300 *points in square with average degree of* 20*.*

*Figure 4.3:* 300 *points in circle with average degree of* 20*.*

*Figure 4.4:* 300 *points in star with average degree of* 20*.*

*Figure 4.5: Boundary nodes (orange) chosen from Algorithm 3 with* $\tau = 20$ *and corner nodes (blue).*

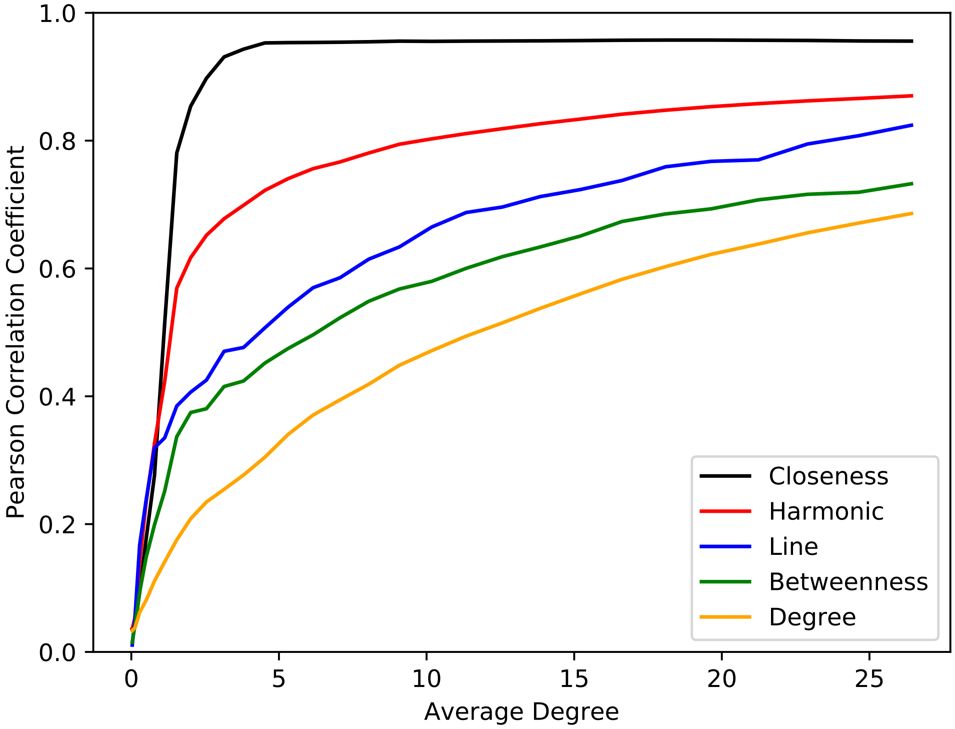


*Figure 4.6: Sample 100 graphs $G \sim \mathcal{G}(100, r, X)$ for a variety of radii $r$, and find the Pearson correlation coefficient with popular centrality measures and the distance from the boundary when $X$ is the unit square.*

The performance of BoundaryDetection$(G, \tau)$ depends on the edge density of the graph. Suppose the $i$th entry of the vector $D$ is the Euclidean distance of node $i$ to the true boundary of $X$. To test the performance of each algorithm, we find the Pearson correlation coefficient of the line centrality and $D$, and plot it for a range of graph densities. In Figure 4.6 we let $n = 100$ and test graphs with the expected average degree between 0 and 25 when $X$ is a square. In this figure, we compare the performance of the line centrality against other well-known centrality measures.

## 4.4 Discussion

Many authors have used their own definition for a boundary node in a graph, but there is no agreed-upon definition. This chapter proposes a new, geometrically motivated, definition for a boundary node in a graph. Chvátal introduced a definition for a line in a graph, and we propose to apply this definition to define a boundary node in a graph. In particular, we define a new centrality measure, called the line centrality, which counts the number of times a node $v$ is contained in the endpoints of a line $L(u, w)$ across all node pairs $u, w$ in the set of nodes. BoundaryDetection$(G, \tau)$ defines a node as a boundary node if the line centrality is larger than some predefined threshold $\tau$.

We tested the line centrality's correlation with the true distance to the boundary on various edge densities in random geometric graphs in Figure 4.6. The line centrality will outperform degree and betweenness centrality for certain densities and approach the harmonic and closeness centrality correlation coefficients as the density increases.

In Figure 4.6, the line centrality ranks in the top three in our analysis. In particular, line centrality correlates more with the distance from the boundary than betweenness centrality. In [92], the authors show that using closeness and betweenness centrality together does well at predicting the boundary nodes. In future work, we plan to see how the line centrality paired with other centrality measures, like closeness or harmonic centrality, performs in boundary node detection.

While boundary detection is one application of Chvátal's line in a graph, this work only scratches the surface of its potential. In basic geometry, many definitions come from a line: intersections, angles, and shapes. In future work, we plan to use Chvátal's line to characterize different graph objects like this. The ultimate goal of this work is to understand the ‘shape of the data,’ and determining how to categorize boundary nodes brings us one step closer.

# Part II

# Extremal Combinatorics: Semi-Saturation and Rainbow Numbers

# Chapter 5

# Rainbow Numbers

The work presented in this chapter is the result of a collaboration with Kean Fallon, Colin Giles, Ethan Manhart, Joe Miller, Dr. Nathan Warnberg, Simon Wagner, and Laura Zinnel, most of which is documented in [97] and [98].

## 5.1 Introduction

One of the most notable problems in extremal combinatorics is due to Frank Ramsey: the Ramsey number.

**Definition 7** (Ramsey number (2-color definition))**.** *[99] The Ramsey number for integers $r, b \geq 1$, written as $n = R(r, b)$, is the smallest integer $n$ such that every 2-edge-coloring of $K_n$, using the colors red and blue for edges, there is a red monochromatic subgraph $K_r$ or a blue monochromatic subgraph $K_b$.*

In Example 1, we give a classical proof that $R(3, 3) = 6$.

**Example 1.** *We will show that $R(3, 3) = 6$.*

*Figure 5.1 contains a* $2$*-edge-coloring of* $K_5$ *that does not contain a monochromatic* $K_3$ *subgraph. This shows that* $6 \leq R(3,3)$*. To show that* $R(3,3) \leq 6$*, we want to show that every 2-edge-coloring of* $K_6$ *will contain a monochromatic* $K_3$ *subgraph.*

*Start by considering a single vertex* $v$*, and notice that by the pigeonhole principle, at least three of the edges, say* $(v,x),(v,y),(v,z)$*, must be the same color, say red. Hence a monochromatic* $K_3$ *is present, and it will contain one or more of the edges* $(x,y),(y,z),(z,x)$*.*

*If any of the three edges are red, say* $(x,y)$*, then the triangle formed by vertices* $x,y,v$ *is monochromatic. However, if all three edges* $(x,y),(y,z),(z,x)$ *are blue, then a monochromatic triangle is formed by the vertices* $x,y,z$*. A monochromatic triangle exists in both cases, so* $R(3,3)$ *is at most* $6$*, therefore equal to* $6$*.*

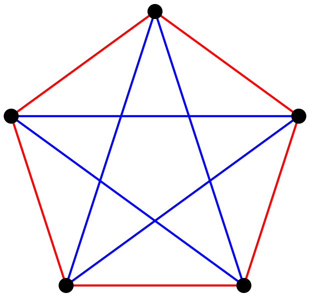


*Figure 5.1: A* $2$*-edge-coloring of* $K_5$ *that avoids a monochromatic* $K_3$ *subgraph.*

Ramsey theory is a branch of combinatorics. Other authors have contributed to the field of Ramsey theory by investigating subsets of the integers. A classic example of this are the Schur numbers, named after Issai Schur, which he defined in 1917 [100]. Note that $[n] = \{1,\ldots,n\}$ for $n \in \mathbb{Z}$.

**Definition 8** (Schur number)**.** *[100] For a given integer* $c$*, the Schur number, denoted* $S(c)$*, is the smallest integer so that every exact (surjective)* $c$*-coloring of* $[S(c)]$ *contains a monochromatic solution to the equation* $x + y = z$*.*

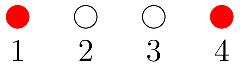


*Figure 5.2: Lower bound for $s(2)$.*

**Example 2.** *In this example, we show that $s(2) = 5$. Figure 5.2 contains a coloring of $[4] = \{1, 2, 3, 4\}$ that avoids monochromatic solutions to $x + y = z$. This shows that $5 \leq s(2)$. Now suppose $c$ is a coloring of $[5]$. Note that $(1, 1, 2)$ and $(2, 2, 4)$ are solutions to $x + y = z$. If neither of these solutions is monochromatic, then without loss of generality $c(1) =$ red, $c(2) =$ blue, and $c(4) =$ red. Because $(1, 3, 4)$ is a solution $c(3) =$ blue. Since $(1, 4, 5)$ is a solution $c(5) =$ blue. But then $(2, 3, 5)$ is monochromatic.*

We only know the Schur numbers for 1 through 5. The 5th Schur number, $S(5)$, was recently found by Heule in 2018 [101]. The proof was done by computer and took two petabytes of space to solve.

Each question in Ramsey theory has a corresponding question in Anti-Ramsey theory. Instead of aiming to guarantee monochromatic structures, Anti-Ramsey theory aims to guarantee structures where every element has a different color, called a *rainbow structure*. We say that $G$ contains a *rainbow copy of $H$* if $H$ is a subgraph of $G$ and for all $e_1, e_2 \in E(H)$, $c(e_1) = c(e_2)$ implies $e_1 = e_2$. As an example, we consider the anti-Ramsey numbers.

**Definition 9** (Anti-Ramsey numbers)**.** *[102] The anti-Ramsey number* $\mathrm{AR}(G, H)$ *is the smallest number of colors $k$ such that every $k$-edge-coloring of $G$ contains a rainbow copy of $H$.*

The Anti-Ramsey number $\mathrm{AR}(G, H)$ is known for a variety of graphs $G$ and $H$ [103–106]. Anti-Ramsey numbers and Schur numbers inspired studying rainbow numbers on $[n]$ with respect to the equation $x_1 + x_2 = x_3$, which we explore in Section 5.2.

## 5.2 Rainbow Numbers on $[n]$

Let $[n] = \{1, 2, \ldots, n\}$ and *eq* be any equation. Here, an *$r$-coloring* of a set $S$ is a function $c : S \to [r]$, where $[r] = \{1, 2, \ldots, r\}$, and an $r$-coloring is *exact* if it is surjective. If $X \subseteq S$, then $c(X) = \{c(x) : x \in X\}$. The *rainbow number* is the smallest number of colors $k$ so that for every exact $k$-coloring of $[n]$, there exists a solution to *eq* with every member of the solution set colored distinctly, denoted $\mathrm{rb}([n], eq)$. While any equation is valid in this definition, we investigate specific linear equations [97].

Let $c$ be an exact $r$-coloring on $[n]$. Define $\mathcal{C}_i = \{a \in [n] \,|\, c(a) = i\}$ and define $s_i \in \mathcal{C}_i$ such that $s_i$ is the smallest element of $\mathcal{C}_i$ for each color $i$. Note that for any exact $r$-coloring $c$, it is always possible to have $s_i < s_j$ for $i < j$. If that is not the case, say $s_i > s_j$ and $i < j$, an isomorphic coloring can be created by swapping the color of any number with color $i$ to have color $j$ and vice versa.

We begin the analysis with Observation 4. This observation establishes the proof structure for each Proposition, Lemma, and Theorem in this section.

**Observation 4.** *[97] If $c$ is an exact $(r-1)$-coloring of $[n]$ that avoids rainbow solutions for eq, then $r \leq \mathrm{rb}([n], eq)$. If every exact $r$-coloring of $[n]$ guarantees a rainbow solution to eq, then $\mathrm{rb}([n], eq) \leq r$.*

This section finds the rainbow number for the equation $\sum_{i=1}^{k-1} x_i = x_k$. To begin, assumed that *eq* is $x_1 + x_2 = x_3$. Also, if we say $(a, b, c)$ is a solution to *eq*, this means that $x_1 = a$, $x_2 = b$ and $x_3 = c$. Every exact $r$-coloring uses the color set $\{0, 1, \ldots, r-1\}$. Lemma 3 starts with a specific coloring that avoids rainbow solutions, thus giving a lower bound.

**Lemma 3.** *[97] For $n \geq 3$, $\lfloor \log_2(n) + 2 \rfloor \leq \mathrm{rb}([n], eq)$.*

*Proof.* The number of trailing zeros that an integer has is the number of zeros, starting from the right, before there is a nonzero digit. Define an exact $(\lfloor \log_2(n) + 1 \rfloor)$-coloring of $[n]$ as

$$c(x) = \text{the number of trailing zeros in the binary representation of } x$$

(see Example 3). Let $a, b \in [n]$. If $a$ and $b$ are odd, then $c(a) = c(b) = 0$ since their binary representation ends with a 1. This means $\{a, b, a+b\}$ is not a rainbow solution. If $a$ is odd and $b$ is even, then $a+b$ is odd so $c(a) = c(a+b) = 0$, thus $\{a, b, a+b\}$ is not a rainbow solution. Finally, consider the case where $a$ and $b$ are both even. If $a$ and $b$ have the same number of trailing zeros, then $c(a) = c(b)$ meaning $\{a, b, a+b\}$ is not a rainbow solution. If $c(a) \neq c(b)$ assume, without loss of generality, that $c(a) < c(b)$. Since the number of trailing zeros of $b$ exceeds the number of trailing zeros of $a$, it follows, via binary arithmetic, that $b + a$ has the same number of trailing zeros as $a$. Thus, $c(a + b) = c(a)$ and $\{a, b, a + b\}$ is not a rainbow solution. Therefore, no rainbow solutions exist with this coloring, and $\lfloor \log_2(n) + 2 \rfloor \leq \mathrm{rb}([n], eq)$. □

**Example 3.** *[97]* A table describing the coloring from Lemma 3.

| $x$ | $x$ *in binary* | $c(x)$ |
|---|---|---|
| 1 | 1 | 0 |
| 2 | 10 | 1 |
| 3 | 11 | 0 |
| 4 | 100 | 2 |
| 5 | 101 | 0 |
| 6 | 110 | 1 |
| 7 | 111 | 0 |
| 8 | 1000 | 3 |

Lemma 4 indicates that when attempting to color $[n]$ and avoid rainbow solutions, there are restrictions on how quickly new colors can be added.

**Lemma 4.** *[97] Let $c$ be an exact $r$-coloring of $[n]$, with color set $\{0, 1, \ldots, r-1\}$, that avoids rainbow solutions, then*

1. *if $s_i = \ell$, then $2\ell \leq s_{i+1}$ for $0 \leq i \leq r-2$,*
2. *$2^i \leq s_i$ for $0 \leq i \leq r-1$.*

*Proof.* For the first claim, if $\ell = 1$, then $i = 0$, so the smallest $m$ for which $s_2 = m$ is $m = 2$. If $\ell \geq 2$, let $1 \leq a < \ell$ with $a + \ell \leq n$. Then $c(a) \neq c(\ell)$, so $c(a+\ell) \in \{c(a), c(\ell)\}$. Thus, $c(a+\ell) \neq i+1$, hence $2\ell \leq s_{i+1}$.

Proving the second claim proceeds by induction on $k$. The base case is easily observed, in particular, $s_0 = 1 = 2^0$. For the induction hypothesis, assume $2^k \leq s_k$ for

$0 \leq k \leq r-2$. Applying part 1 to our induction hypothesis yields $2(2^k) = 2^{k+1} \leq s_{k+1}$ which completes the proof. □

Theorem 2 follows almost directly from Lemmas 3 and 4.

**Theorem 2.** *[97] For $n \geq 3$, $\mathrm{rb}([n], eq) = \lfloor \log_2(n) + 2 \rfloor$.*

*Proof.* By Lemma 3 $\lfloor \log_2(n) + 2 \rfloor \leq \mathrm{rb}([n], 3)$. Let $r = \lfloor \log_2(n) + 2 \rfloor$ and $c$ be an exact $r$-coloring, with color set $\{0, 1, \ldots, r-1\}$, of $[n]$ that avoids rainbow solutions. Then, by Lemma 4, $2^{r-1} \leq s_{r-1}$, so

$$2^{r-1} = 2^{\lfloor \log_2(n)+1 \rfloor} \leq s_{r-1}.$$

Thus, $n < s_{r-1}$. So if color $r-1$ appears in a coloring of $[n]$ there must be a rainbow solution, a contradiction that such a coloring exists. Therefore, $\mathrm{rb}([n], eq) = \lfloor \log_2(n) + 2 \rfloor$. □

This concludes the analysis for when *eq* is $x_1 + x_2 = x_3$. Next, we find the number for when *eq* is $x_1 + x_2 + x_3 = x_4$ in $[n]$ in Theorem 3. Lemma 5 provides a coloring that establishes a lower bound on the rainbow number for *eq*.

**Observation 5.** *[97]* Define $L = b_1$ with $\ell = 2$. Now, $L$ can be used to define a coloring on $[n]$ that avoids rainbow solutions to *eq*. In particular, $L = \left\lceil \frac{n-1}{2} \right\rceil$ gives the point in $[n]$ such that no solutions to *eq* exist if two of $x_1, x_2$ or $x_3$ are greater than or equal to $L$. This is useful in the proof of Lemma 5.

Note that Lemma 5 is just a specific case of a more general, namely Theorem 4. Lemma 5 is included here to help Section 3 stand alone.

**Lemma 5.** *[97] For $n \geq 5$,* $\left\lfloor \frac{1}{2}(n+7) \right\rfloor \leq \operatorname{rb}([n], eq)$.

*Proof.* Let $L = \left\lceil \frac{n-1}{2} \right\rceil$ and define

$$c(x) = \begin{cases} 1 & \text{if } 1 \leq x < L \\ i+2 & \text{if } x = L+i \end{cases}.$$

Note that if a solution to $eq$ is rainbow, it must contain three or more numbers greater than or equal to $L$. However, Observation 5 shows no such solutions. Hence, this coloring avoids rainbow solutions.

It remains to count how many colors are used. The number of elements that are not colored 1 is $n - L + 1 = n - \left\lceil \frac{n-1}{2} \right\rceil + 1 = \left\lfloor \frac{n+1}{2} \right\rfloor + 1$. Including the color 1 gives $\left\lfloor \frac{n+1}{2} \right\rfloor + 2$ colors. Since this coloring avoids rainbow solutions, Observation 4 implies $\left\lfloor \frac{n+1}{2} \right\rfloor + 3 = \left\lfloor \frac{1}{2}(n+7) \right\rfloor \leq \operatorname{rb}([n], eq)$. □

Lemma 6 serves as the base case for the inductive proof of Theorem 6.

**Lemma 6.** *[97] If $n = 7$, then* $\operatorname{rb}([n], eq) = 7$ *and there is a unique extremal coloring.*

*Proof.* The lower bound is a result of Lemma 5. The upper bound is trivial since the only coloring of [7] with seven colors is to color each number distinctly. Now let $c$ be an exact 6-coloring of [7]. Observe that $c(1) = 1$ and that if $s_6 \neq 7$, then $\{1, 2, 3, 6\}$ would be a rainbow. Thus, $s_6 = 7$. If $s_2 = 2$, then $1+2+4 = 7$ implies $c(4) \in \{1, 2\}$ so $c(3) = 3$, $c(5) = 4$ and $c(6) = 5$. However, this gives the rainbow solution $\{1, 2, 3, 6\}$. Therefore, the only extremal coloring of [7] is the coloring described in Lemma 5. □

**Proposition 6.** *[97] If $n$ is odd and $n \geq 5$, then* $\operatorname{rb}([n], eq) = \frac{1}{2}(n+7)$ *and there is a unique extremal coloring of* $[n]$.

*Proof.* When $n = 5$ the result is immediate and $n = 7$ is given by Lemma 6. For the induction hypothesis, assume that for all odd $k$ with $7 \leq k \leq n$, $n$ odd, that $\mathrm{rb}([k], eq) = \frac{1}{2}(k+7)$ and there is a unique extremal coloring of $[k]$, namely the coloring provided in Lemma 5. Define $r = \frac{1}{2}((n+2)+7)$, $L = \frac{n-1}{2}$ and let $c$ be an exact $r$-coloring of $[n+2]$ with color set $\{1, \ldots, r\}$. The induction hypothesis implies that if $r$ or $r-1$ colors appear in $[n]$, then there is a rainbow solution, so $s_r = n+2$ and $s_{r-1} = n+1$. The induction hypothesis also implies that if $r-2$ colors appear in $[n]$, a unique coloring of $[n]$ with $r-2$ colors avoids rainbow solutions. However, applying the unique coloring yields $c(1) = 1$, $c(L) = 2$, $c(L+1) = 3$ and $c(n+1) = r-1 > 3$, thus $\{1, L, L+1, n+1\}$ is a rainbow solution. Therefore, $\mathrm{rb}([n+2], eq) \leq \frac{1}{2}(n+9)$. Equality comes from the lower bound in Lemma 5.

Now let $c$ be an exact $(r-1)$-coloring of $[n+2]$, with color set $\{1, 2, \ldots, r-1\}$, that avoids rainbow solutions. If $r-1$ colors appear in $[n]$ the inductive hypothesis implies there is a rainbow solution, thus $n+1 \leq s_{r-1} \leq n+2$.

If $r-2$ colors appear in $[n]$, then $[n]$ has the coloring from Lemma 5. This means $s_2 = L$ and $s_3 = L+1$. Hence, either $\{1, L, L+1, n+1\}$ or $\{2, L, L+1, n+2\}$ is a rainbow solution since $n+1 \leq s_{r-1} \leq n+2$. Therefore, $n < s_{r-2}$ which implies $s_{r-2} = n+1$ and $s_{r-1} = n+2$.

Consider the case where $r-3$ colors appear in $[n]$. If $r-j-1$ colors appear in $[n-2j+2]$, for $2 \leq j \leq \frac{n+3}{4}$, then the coloring from Lemma 5 gives $L_j = \frac{n-2j+2-1}{2} = L-j+1 = s_2$, $L-j+2 = s_3$, $\ldots$ ,$n-2j+2 = s_{r-j-1}$. However, notice that

$$L-j+1 \leq L < L+1 \leq n-2j+2 < n+1,$$

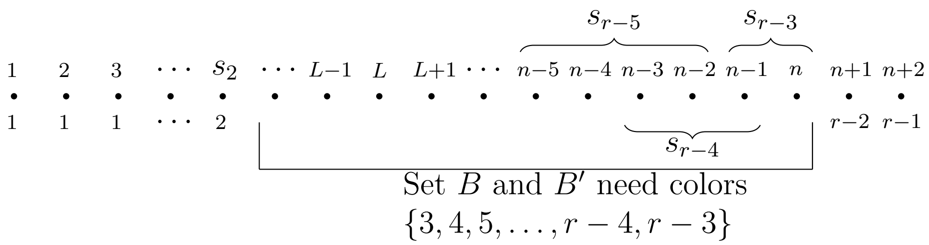
thus $\{1, L, L+1, n+1\}$ is a rainbow solution which implies $n-2j+3 \leq s_{r-j-1}$.

Combining this with $s_i < s_{i+1}$ gives

$$n - 2j + 3 \leq s_{r-j-1} \leq n - j + 2. \tag{*}$$

Note that Inequality (*) gives $s_2 = s_{r-(r-3)-1} \leq n - (r-3) + 2 = L + 1$ and (*) is also crucial in Case 3 below. The remainder of the proof will show that if $s_2 \neq L + 1$, there is a rainbow solution.

Case analysis will proceed by considering $s_2 = L - i$.



$s_{r-5}$
$s_{r-3}$
1 2 3 ⋯ $s_2$ ⋯ $L-1$ $L$ $L+1$ ⋯ $n-5$ $n-4$ $n-3$ $n-2$ $n-1$ $n$ $n+1$ $n+2$
1 1 1 ⋯ 2
$r-2$ $r-1$
$s_{r-4}$
Set $B$ and $B'$ need colors
$\{3, 4, 5, \ldots, r-4, r-3\}$

*Figure 5.3: The situation for Cases* 1,2, *and* 3. *The top row is elements in* $[n+2]$, *and the bottom is their corresponding colors.*

*Case 1:* $\dfrac{L}{2} < s_2 \leq L$, i.e. $0 \leq i < L/2$
Notice that $n = 2L + 1$, since $L = \dfrac{n-1}{2}$. For $1 \leq \alpha \leq i$, the equation

$$\alpha + [L - i] + [2L + 2 - (L - i) - \alpha] = 2L + 2 \tag{1}$$

and

$$1 + [L - i] + [2L + 3 - (L - i) - 1] = 2L + 3 \tag{2}$$

imply $\{\alpha, L-i, 2L+2-(L-i)-\alpha, 2L+2\}$ and $\{\alpha, L-i, 2L+3-(L-i)-1, 2L+3\}$ are nondegenerate solutions since

$$L - i < L + 2 + L + i - L - i \leq 2L + 2 - (L - i) - \alpha$$

and

$$2L + 2 - (L - i) - \alpha < 2L + 3 - (L - i) - 1 \leq 2L + 1.$$

Further, $c(\alpha) = 1$, $c(L - i) = 2$, $c(2L + 2) = r - 2$ and $c(2L + 3) = r - 1$. Define

$$A = \{2L + 2 - (L - i) - \alpha \,|\, 1 \leq \alpha \leq i\} \cup \{2L + 3 - (L - i) - 1\},$$

i.e. $A = \{L + 2, \ldots, L + 2 + i\}$. If any element of $A$ is not color 1 or 2, then Equation(s) (1) or (2) give a rainbow solution, thus $c(A) \subseteq \{1, 2\}$. Now define $B = \{L - i + 1, \ldots, 2L + 1\}$ and notice that elements in the set $B\backslash A$ have not yet been assigned a color and there are $|\{3, 4, \ldots, r - 3\}| = L$ colors left to be assigned (see Figure 5.3). However, $|B\backslash A| = |B| - |A| = (L + i + 1) - (i + 1) = L$. So every element in $B\backslash A$ is colored distinctly. Observe, $L + 1 \in B\backslash A$ and $L \in B\backslash A$ or $L = s_2$. Therefore, $\{1, L, L + 1, 2L + 2\}$ is rainbow solution.

*Case 2:* $\dfrac{L}{2} = s_2$, i.e. $i = \dfrac{L}{2}$ is an integer

In this case, Equation (1) is valid for $1 \leq \alpha \leq i - 1$ and Equation (2) is valid. Define $A' = \{L + 3, L + 4, \ldots, 3L/2 + 2\}$ and notice $c(A') \subseteq \{1, 2\}$. Letting $B' = \{L/2 + 1, \ldots, 2L + 1\}$ gives that $|B'\backslash A'| = L + 1$ elements need to be assigned one of $|\{3, 4, \ldots, r - 3\}| = L$ colors. This means either, one of the colors $\{1, 2\}$ appear in $c(B'\backslash A')$ exactly once or there exists exactly one pair of elements $\ell, \ell' \in B'\backslash A'$ with $c(\ell), c(\ell') \notin \{1, 2\}$ and $c(\ell) = c(\ell')$. Consider $c(L/2 + 1)$ and

$c(L+2)$. Notice that $\mathcal{S} = \{1, L/2, L/2+1, L+2\}$ is a solution, and if it is rainbow there is a contradiction. If $\mathcal{S}$ is not rainbow, then either $c(L/2+1) = c(L+2)$ or $c(L/2+1) \in \{1,2\}$ or $c(L+2) \in \{1,2\}$. In these scenarios, $|c(\{L, L+1\}) \backslash \{1,2\}| = 2$ and $\{1, L, L+1, 2L+2\}$ is a rainbow solution.

*Case 3:* $2 \leq s_2 < \dfrac{L}{2}$, i.e. $L/2 < i \leq L-2$

Consider the equations

$$\beta + [L-i] + [n+1-(L-i)-\beta] = n+1 \tag{3}$$

and

$$1 + [L-i] + [n+2-(L-i)-1] = n+2 \tag{4}$$

for $1 \leq \beta \leq L-i-1$. To assure that Equations (3) and (4) are valid, i.e. non-degenerate, consider the following inequalities:

$$L/2 < i$$

$$L < 3L/2 + 3 < 3i + 3$$

$$L - i < 2i + 3 = n - 2L + 2i + 2$$

and

$$i \leq L - 2$$

$$n - L + i + 2 \leq n$$

$$n - L + i + 1 \leq n - 1.$$

Now, if $A'' = \{n-2L+2i+2, n-2L+2i+3, \ldots, n-L+i+1\}$ and recalling that $c(\beta) = 1$, $c(L-i) = 2$, $c(n+1) = r-2$ and $c(n+2) = r-1$, gives $c(A'') \subseteq \{1, 2\}$.

Using Inequality (*), and it's restrictions on $j$, observe that if $j' = L - i + 1$, then $L/2 < i \leq L - 2$ gives $3 \leq j' < \dfrac{n+3}{4}$. Thus,

$$n - 2L + 2i + 1 \leq s_{r-j'-1} \leq n - L + i + 1 \text{ (see Figure 5.3).}$$

However, all possible values for $s_{r-j'-1}$ are in $A''$ except for $n - 2L + 2i + 1$, so $s_{r-j'-1} = n - 2L + 2i + 1$.

A similar argument using $j' + 1 = L - i$ and the upper bound of Inequality (*) gives $s_{r-j'} = n-L+i+2$. However, since $1+s_2+s_{r-j'-1} = s_{r-j'}$, $\{1, s_2, s_{r-j'-1}, s_{r-j'}\}$ is a rainbow solution.

A rainbow solution has been found in all cases with $s_2 \leq L$. Therefore, $s_{2+i} = L + 1 + i$ for $0 \leq i \leq r - 3$ is the only way to color $[n+2]$ with $r - 1$ colors which is the coloring described in Lemma 5. □

**Lemma 7.** *[97] If* $n \geq 5$ *is odd, then* $\text{rb}([n+1], eq) \leq \dfrac{1}{2}(n+7)$.

*Proof.* Let $r = \dfrac{1}{2}(n+7)$ and $c$ be an exact $r$-coloring of $[n+1]$. If $r$ colors appear in $[n]$, then, by Proposition 3, there exists a rainbow solution. Thus $c(n+1) = r$ and $r - 1$ colors appear in $[n]$. Note that there is unique extremal coloring of $[n]$ with $r-1$ colors, thus $\left\{1, \dfrac{n-1}{2}, \dfrac{n+1}{2}, n+1\right\}$ is a rainbow solution since $c(1) = 1$, $c((n-1)/2) = 2$, $c((n+1)/2) = 3$ and $c(n+1) = r$. Therefore, $\text{rb}([n+1], eq) \leq \dfrac{1}{2}(n+7)$. □

Theorem 3 follows Proposition 6 and Lemma 7.

**Theorem 3.** *[97] For $n \geq 5$,* $\mathrm{rb}([n], eq) = \left\lfloor \frac{1}{2}(n+7) \right\rfloor$.

This concludes our analysis when $eq$ is $x_1 + x_2 + x_3 = x_4$. For a more general linear equation, a similar coloring to the one established in Lemma 5 can be used to avoid rainbow solutions and arrive at Theorem 4. We skip the proof here, but refer to [97] for more details.

**Theorem 4.** *[97] Let* $eq : \sum_{i=1}^{k-1} x_i = x_k$*,* $k \geq 4$ *and define* $L = \left\lceil \frac{2n - \ell(\ell-1)}{2\ell} \right\rceil$ *for* $\ell = k-2$*. Then*

$$\mathrm{rb}([n], eq) \geq \begin{cases} n+1 & \text{if } n < \frac{(k-1)k}{2} \\ n - L + 3 & \text{otherwise} \end{cases} .$$

# 5.3 Rainbow Numbers on $[m] \times [n]$

In Section 5.2, we outline the results in [97] to find $\mathrm{rb}([n], eq)$ for certain equations $eq$. In this section, we consider the same question but with $eq$ fixed as $x_1 + x_2 = x_3$, and instead, we use the set

$$[m] \times [n] = \{(i,j) : i, j \in \mathbb{Z}, 1 \leq i \leq m \text{ and } 1 \leq j \leq n\}$$

with $1 \leq m \leq n$. In short, we determine that $\mathrm{rb}([m] \times [n], x_1 + x_2 = x_3) = m + n + 1$. Note that addition is component-wise in $[m] \times [n]$, and if an exact (surjective) $r$-coloring $c : [m] \times [n] \to [r]$ does not contain a rainbow solution, then $c$ is said to be *rainbow-free.*

The analysis in [98] is rather lengthy, so we outline many important results instead of rewriting the whole paper. To enable this dialog, we define

$$D_k = \Big\{(i,j) \in [m] \times [n] \,:\, m - k = i - j\Big\}$$

as the *kth-diagonal* in $[m] \times [n]$. The $m$th-diagonal, $D_m$, is called the *main diagonal*, and every other diagonal is an *off-diagonal*. On a high level, we categorize each coloring by the number of colors in the main diagonal. That is, we analyze when $|c(D_m)| < 3$, $|c(D_m)| = 3$, and $|c(D_m)| > 3$. The final section in [98] ties it together and states the desired result. The remainder of this discussion gives a high-level outline of the paper.

Firstly, we establish a lower bound on $\mathrm{rb}([m] \times [n], eq)$ in Lemma 8.

**Lemma 8.** *[98] For $2 \le m \le n$, $m + n + 1 \le \mathrm{rb}([m] \times [n], eq)$.*

*Proof.* Let $c : [m] \times [n] \to [m+n]$ be defined by

$$c((i,j)) = \begin{cases} 1 & \text{if } i < m \text{ and } j < n, \\ i+1 & \text{if } i < m \text{ and } j = n, \\ j+m & \text{if } i = m. \end{cases}$$

Note that $(1,n)$ and $(m,1)$ are not in any solution to $eq$. If $\alpha, \beta, \gamma \in [m] \times [n]$ such that $\alpha + \beta = \gamma$, then $c(\alpha) = c(\beta) = 1$. Therefore, $c$ is rainbow-free for $eq$ and $m + n + 1 \le \mathrm{rb}([m] \times [n], eq)$. □

Lemma 9 indicates that the color set of an off-diagonal cannot differ from the color set of the main diagonal by more than one color.

**Lemma 9.** *[98] If $c$ is a rainbow-free coloring of $[m] \times [n]$ for $eq$ with $m \le n$, then, for all $D_x$ with $x \neq m$, $\Big|c(D_x) \setminus c\Big(D_m\Big)\Big| \le 1$.*

*Proof.* If $m = 1$ each diagonal has one element and the result follows. Thus, assume $2 \leq m$. Since $|D_1| = |D_{m+n-1}| = 1$, it is certainly true that $|c(D_1) \setminus c(D_m)| \leq 1$ and $|c(D_{m+n-1}) \setminus c(D_m)| \leq 1$.

For the purpose of contradiction, assume $1 < x < m+n-1$ and $|c(D_x) \setminus c(D_m)| \geq 2$. Then there exists $\beta$, $\gamma \in D_x$ such that $c(\beta), c(\gamma) \in c(D_x) \setminus c(D_m)$. However, there exists $\alpha \in D_m$ such that $\{\alpha, \beta, \gamma\}$ or $\{\alpha, \gamma, \beta\}$ is a rainbow solution, a contradiction. Thus, $\left|c(D_x) \setminus c\Big(D_m\Big)\right| \leq 1$. □

Of the collection of off-diagonals that contain a color, not in the main diagonal, the one with the smallest index is called a *contributing diagonal*. As an immediate consequence of Lemma 9, Corollary 3 indicates that for an $(m + n + 1)$-coloring $c$ to be rainbow-free, $|c(D_m)| \geq 3$.

**Corollary 3.** *[98] If $c$ is an exact, rainbow-free $(m+n+1)$-coloring of $[m] \times [n]$ for eq with $3 \leq m \leq n$, then $|c(D_m)| \geq 3$.*

Let $c$ be an $(m + n + 1)$-coloring of $[m] \times [n]$ and suppose $|c(D_m)| = 3$. The motivation for many of the following results is derived from the notable staircase pattern in the extremal $m + n$-colorings. A description of this pattern, which we call the staircase pattern, is regarding consecutive contributing diagonals. This begins with Lemma 10 stating that each off-diagonal must contribute if $|c(D_m)| = 3$.

**Lemma 10.** *[98] If $c$ is an exact, rainbow-free $(m + n + 1)$-coloring of $[m] \times [n]$ for eq with $3 \leq m \leq n$ and $|c(D_m)| = 3$, then each off-diagonal $D_k$ contributes exactly one color $c_k$ such that $c_k \notin c([m] \times [n] \setminus D_k)$.*

If $\alpha \in D_a, \beta \in D_{a+1}$ where $D_a$ and $D_{a+1}$ are contributing, and $c(\alpha), c(\beta) \notin c(D_m)$, then $\{\alpha, \beta\}$ are a *consecutive contributing pair of elements.* If $\{\alpha, \beta\}$ is a consecutive contributing pair of elements with $\alpha = (a_1, a_2)$ and $\beta = (a_1, a_2 + 1)$, then $\{\alpha, \beta\}$ is a *horizontal pair* and if $\alpha = (a_1, a_2)$ and $\beta = (a_1 - 1, a_2)$, then $\{\alpha, \beta\}$ is a *vertical pair*. Let $P_v$ be a vertical pair, $P_h$ be a horizontal pair, and define $P = P_v \cup P_h$. If $P_v \cap P_h = \emptyset$, $P_v \cap W \neq \emptyset$, and $P_h \cap W \neq \emptyset$, $P$ is called a *contributing disjoint corner*.

Many of the results to come are leveraged to force certain structures in $[m] \times [n]$. For example, Lemma 11 indicates that the elements in the diagonals that are colored distinctly must be next to each other in one of two ways, as either a horizontal or vertical pair as pictured in Figure 5.4, taken from [98].

**Lemma 11.** *[98] If $c$ is an exact, rainbow-free $(m + n + 1)$-coloring of $[m] \times [n]$ for eq with $3 \leq m \leq n$ and $|c(D_m)| = 3$, then there are no jumps from $\alpha$ to $\beta$ with $c(\alpha), c(\beta) \notin c(D_m)$ and $c(\alpha) \neq c(\beta)$.*

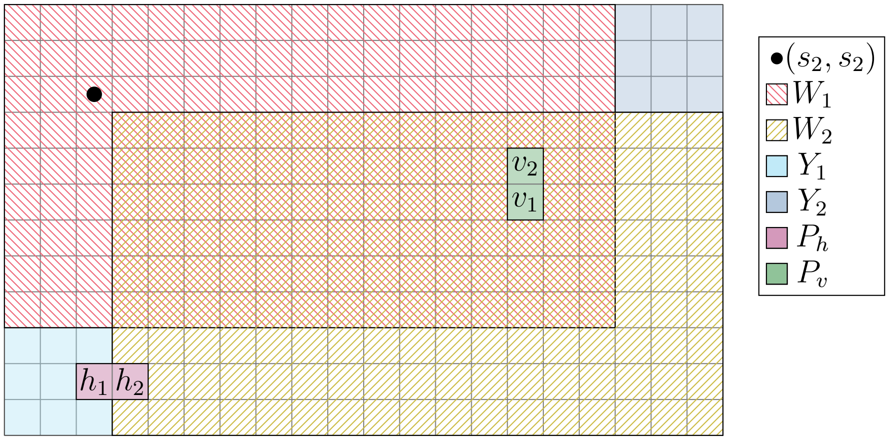


*Figure 5.4: An example of $[m] \times [n]$ when $(s_2, s_2) = (3, 3)$ with corresponding $Y_1$, $Y_2$, $W_1$ and $W_2$ highlighted. Further, a contributing disjoint corner, $P_h \cup P_v$, with $c(h_1), c(h_2), c(v_1)$ and $c(v_2)$ pairwise distinct, is shown.*

Lemma 12 shows that if both a horizontal and vertical pair are in $[m] \times [n]$, there must also be a rainbow solution.

**Lemma 12.** *[98] If $c$ is a rainbow-free coloring of $[m] \times [n]$ for eq with $m \leq n$, then there are no contributing disjoint corners $P = P_v \cup P_h$ in $[m] \times [n]$.*

To rule out the case where $|c(D_m)| = 3$, in Theorem 5, we show that every $(m+n+1)$-coloring with $|c(D_m)| = 3$ must contain a horizontal and vertical pair, and hence a rainbow solution.

**Theorem 5.** *[98] If $c$ is an exact $(m+n+1)$-coloring of $[m] \times [n]$ with $3 \leq m \leq n$ and $|c(D_m)| \leq 3$, then $[m] \times [n]$ contains a rainbow solution to eq.*

Now that we have determined that every exact $(m+n+1)$-coloring $c$ with $|c(D_m)| \leq 3$ must have a rainbow solution, we investigate colorings where $|c(D_m)| \geq 4$. We show that if two elements $\alpha, \beta \in [m] \times [n]$ are distinctly colored with colors that are not in the main diagonal, then there cannot exist a solution of the form $\alpha + \delta = \beta$ (called a *jump*) for any $\delta = (d_1, d_2)$. We arrive at this result by bounding the size of $d_1 + d_2$ above and below, which leads to a contradiction and ultimately Theorem 6, indicating that no jumps can exist.

**Theorem 6.** *[98] Let $c$ be an exact, rainbow free $(m+n+1)$-coloring of $[m] \times [n]$ for eq with $3 \leq m \leq n$, and suppose there is a jump from $\alpha \in D_a$ to $\beta \in D_b$ such that $\alpha + \delta = \beta$ for some $\delta = (d_1, d_2) \in D_t$. Then $c(\alpha) \in c(D_m)$ or $c(\beta) \in c(D_m)$ or $c(\alpha) = c(\beta)$.*

Now that we have eliminated all jumps, we conclude with the analysis of $|c(D_m)| \geq 4$. We use the limitations imposed by Theorem 6 to show in Lemma 13 that only a certain number of horizontal and vertical pairs can exist.

**Lemma 13.** *[98] Let $c$ be an exact, rainbow-free $(m+n+1)$-coloring of $[m] \times [n]$ for eq with $4 \leq m \leq n$. If there is a horizontal pair $P_h$ intersecting $W$, then there* *are at most $2s_2 - 2$ vertical pairs in $[m] \times [n]$. Likewise, if there is a vertical pair $P_v$ intersecting $W$, then there are at most $2s_2 - 2$ horizontal pairs in $[m] \times [n]$.*

Finally, in the proof of Theorem 7, we contradict Lemma 13, giving us the final result. That is, $\mathrm{rb}([m] \times [n], x_1 + x_2 = x_3) = m + n + 1$.

**Theorem 7.** *[98] If $c$ is an exact $(m+n+1)$-coloring of $[m] \times [n]$ with $8 \leq m \leq n$, then $[m] \times [n]$ contains a rainbow solution to eq.*

In [107], the authors found the rainbow number for $x_1 + x_2 = ax_3$ in the group $\mathbb{Z}_n$. In [108], the authors found more general results for the linear equations $a_1x_1 + a_2x_2 + a_3x_3 = b$ on the group $\mathbb{Z}_n$. The rainbow number has not been studied for other equations *eq* or sets $S$ beyond this. In future work, we plan to investigate non-abelian groups, such as the dihedral groups, with various linear equations.

# Chapter 6

# Graph Saturation

## 6.1 Introduction

The forbidden subgraph problem is a question of the form: find the maximal number of edges $\mathrm{ex}(n, H)$ so every n-vertex graph with $\mathrm{ex}(n, H)$ edges does not have a subgraph isomorphic to $H$. The value $\mathrm{ex}(n, H)$ is called the *extremal number* or the *Turán number*.

The history of extremal numbers goes back to 1906 when the following question was posed by Mantel: What is the maximum number of edges in a $K_3$-saturated graph? This was answered in [109]. Wythoff showed that a complete bipartite graph with the same size partite sets (or sizes off by one) maximizes the number of edges in a $K_3$-saturated graph on $n$ vertices. It was not until 1945 that this result was extended to $K_r$-saturated graphs ($r \geq 3$) by Turán in [110]. Turán proved that a unique graph maximizes the number of edges over all $K_r$-saturated graphs. The graph that realizes the extremal number $\mathrm{ex}(n, K_r)$ is the complete $(r-1)$-partite graph on $n$ vertices where each partite set is the same size or within 1 of each other, known as *balanced*. Since then, many authors have found the extremal number for various graph families [111].

There are a large number of questions that arise from the forbidden graph problem. Typically, we aim to characterize the set of graphs that do not contain a subgraph $H$. In particular, we ask, what is the smallest number of edges in a graph $G$ so that adding any edge $e$ to $G$ forces a subgraph isomorphic to $H$ containing $e$? If $G$ is a graph sp that adding any edge creates a copy of $H$ containing that edge, then $G$ is *$H$-semi-saturated*. The fewest number of edges in an $H$-semi-saturated graph on $n$ vertices is the *semi-saturation number* of $H$, denoted $\mathrm{ssat}(n, H)$. If an $H$-semi-saturated graph $G$ does not contain a subgraph isomorphic to $H$, called *$H$-free*, then we say $G$ is *$H$-saturated*. The smallest number of edges in such a graph on $n$ vertices is called the *saturation number* of $H$, denoted $\mathrm{sat}(n, H)$. To give an example of saturation, we start with a few definitions.

A *complete graph on $n$ vertices*, denoted $K_n$, is a graph where an edge connects each pair of the $n$ vertices. A *complete bipartite graph* $G = (V, E)$, denoted $K_{m,n}$, is a graph where $V$ can be partitioned into two sets $V_1$ and $V_2$, with $|V_1| = n$ and $|V_2| = m$, where every edge of the form $(v_1, v_2)$ with $v_1 \in V_1$ and $v_2 \in V_2$ is in $E$, and no other edges are present. Each set $V_1$ and $V_2$ is called a *partite set*, a maximal independent set. We sometimes call $K_{1,n}$ a *star graph on $n$ vertices*. A *$k$-partite graph* is a graph whose vertices are (or can be) partitioned into $k$ different partite sets. Next, we present an example of saturation that uses $K_{1,5}$ and $K_3$.

Consider the star graph $K_{1,5}$ in Figure 6.1. A *triangle* is a complete graph on 3 vertices, denoted $K_3$. $K_{1,5}$ does not contain any triangles. However, as Figure 6.2 demonstrates, replacing any non-edge with an edge creates a triangle. Thus, adding an edge to $K_{1,5}$ forms $K_3$ as a subgraph. Therefore $K_{1,5}$ is $K_3$-saturated. This shows that $sat(6, K_3) \leq 5$.

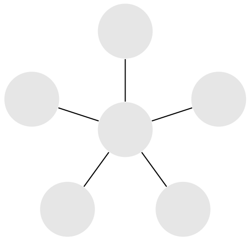


*Figure 6.1: Star ($K_{1,5}$)*

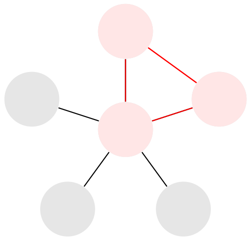


*Figure 6.2: Add edge to $K_{1,5}$*

Before continuing our example, we go through a few definitions. A graph $G = (V, E)$ is *connected* if there is a path between every pair of vertices in the graph. Otherwise, $G$ is *disconnected*. A *subgraph* of $G$ is a subset of the vertex and edge sets. An *induced subgraph*, denoted $G[V']$, is the subgraph on vertex set $V'$ where the edge set is all edges in $E$ with both endpoints contained in $V'$. If a subset $V'$ of $V$ is a maximal set so that the induced subgraph $G[V']$ is connected, then $G[V']$ is a *connected component* of $G$. Connected graphs with the smallest number of edges are called *trees*. That is, if removing any edge disconnects the graph, then it is a tree.

Now, continuing our example on $K_3$-saturated graphs, we show that $K_{1,5}$ is a $K_3$-saturated graph with the fewest edges over all graphs on 6 vertices. That is, we show that $\text{sat}(6, K_3) \geq 5$.

Suppose $G$ is a graph on 6 vertices with fewer than 5 edges that is $K_3$-saturated. Notice that $K_{1,5}$ is a tree. Therefore, $G$, having one less edge than $K_{1,5}$, must have at least two connected components, call them $G_1$ and $G_2$ with vertex sets $V_1$ and $V_2$, respectively. Adding any edge $(v_1, v_2)$ for $v_1 \in V_1$ and $v_2 \in V_2$ does not form a cycle, and, thus, does not form a $K_3$, contradicting the assumption that $G$ was

$K_3$-saturated. Therefore, the smallest number of edges in a $K_3$-saturated graph on 6 vertices is 5. Thus, $\text{sat}(6, K_3) = 5$. A similar argument applies to $K_{1,n-1}$. Thus, $K_{1,n}$ is a $K_3$-saturated graph on $n$ vertices with the fewest possible edges. This shows that $\text{sat}(n, K_3) = n - 1$.

The semi-saturation and saturation numbers can be defined on a family of graphs, as in Definitions 11 and 10.

**Definition 10.** *For a family of graphs $\mathcal{F}$, a graph $G = (V, E)$ is $\mathcal{F}$-semi-saturated if for all $e \notin E$, $G + e$ contains a subgraph isomorphic to some $H \in \mathcal{F}$ which contains $e$. The semi-saturation number of $\mathcal{F}$ on $n$ vertices, denoted* $\text{ssat}(n, \mathcal{F})$*, is the smallest number of edges in an $\mathcal{F}$-semi-saturated graph on $n$ vertices.*

**Definition 11.** *Let $G = (V, E)$ be a simple undirected graph. If $v, w \in V$, and $e = (v, w)$ not contained in $E$, then $G + e := (V, E \cup \{e\})$. For a family of graphs $\mathcal{F}$, a graph $G$ is $\mathcal{F}$-saturated if $G$ is $H$-free for all $H \in \mathcal{F}$, and for all $e \notin E(G)$, the graph $G + e$ contains a subgraph isomorphic to some $H \in \mathcal{F}$. The saturation number of $\mathcal{F}$ on $n$ vertices, denoted* $\text{sat}(n, \mathcal{F})$*, is the smallest number of edges in an $\mathcal{F}$-saturated graph on $n$ vertices.*

Kászonyi and Tuza found a general upper bound for the saturation number of a graph $H$, $\text{sat}(n, \mathcal{F})$, which is linear in $n$ [112].

In summary, we have

$$\begin{aligned}
\text{ssat}(n, H) &= \min\{||G|| : |G| = n, G \text{ is } H\text{-semi-saturated}\} \\
\text{sat}(n, H) &= \min\{||G|| : |G| = n, G \text{ is } H\text{-saturated}\} \\
\text{ex}(n, H) &= \max\{||G|| : |G| = n, G \text{ is } H\text{-saturated}\}
\end{aligned}$$

where $\text{sat}(n, H) \leq \text{ex}(n, H)$. We say the *saturation spectrum* of $H$ is a set of integers, the minimum of which is $\text{sat}(n, H)$ and the maximum $\text{ex}(n, H)$.

A graph that is $H$-saturated is already $H$-semi-saturated. Therefore, $\text{ssat}(n, H) \leq \text{sat}(n, H)$. There are examples of $(n, H)$ pairs where the saturation and semi-saturation numbers are unequal. For example, Burr, in their Master's thesis [113], describes that $\text{ssat}(n, H) < \text{sat}(n, H)$ when $H$ is a path graph. Additionally, Furëdi and Kim showed a similar result for cycles of any length [114]. We find graphs where the saturation and semi-saturation numbers differ.

For example, define a *path on $n$ vertices* $P_n$ is a collection of $n$ vertices $V = (v_1, v_2, \ldots, v_n)$ so that $(v_i, v_i + 1)$ for $i = 1, \ldots, n - 1$ are all the edges. A disjoint union of graphs $G$ and $H$, denoted $G \cup H$, has vertex set $V = V(G) \cup V(H)$ and edge set $E = E(G) \cup E(H)$. A disjoint union of paths is called a *linear forest*. The graph in Figure 6.3 is the smallest $P_5$-semi-saturated graph [113]. Any edges you add to this graph forms a copy $P_5$. This graph contains a copy of $P_5$, so it is not $P_5$-saturated but rather is $P_5$-semi-saturated. Figure 6.4 is the smallest $P_5$-saturated graph [112]. This shows that $\text{ssat}(8, P_5) < \text{sat}(8, P_5)$. Thus relaxing the $H$-free condition on the saturation number changes the question. Burr [113] proved that for big enough $n$, $\text{ssat}(n, P_k) < \text{sat}(n, P_k)$. We use a result from [115] in Section 6.2 to show that a linear forest has a similar property.

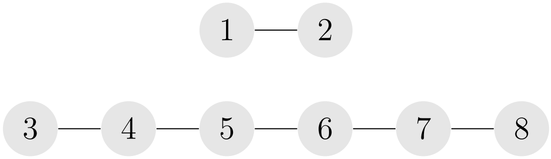


*Figure 6.3: $P_5$-semi-saturated graph*

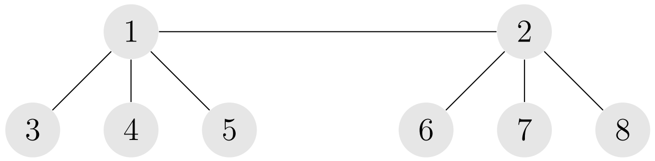


*Figure 6.4: $P_5$-saturated graph*

## 6.2 Linear Forests

In this section, we show that for large enough $n$, $\mathrm{ssat}(n, F) < \mathrm{sat}(n, F)$ where $F$ is a linear forest. We will reference results on saturation of linear forests from [115] and results on semi-saturation of paths from [113]. Suppose that $P_k$ denotes a path graph on $k$ vertices with $k-1$ edges for the following propositions.

**Proposition 7.** *Let $F = P_{k_1} \cup \cdots \cup P_{k_t}$ with $k_1 \geq \cdots \geq k_t = k$. Then* $\mathrm{ssat}(n, F) \leq \mathrm{ssat}(n-\ell, P_k) + c(F)$ *where $c(F) = \binom{\ell}{2}$ with $\ell = -1 + \sum_{i=1}^{t} k_i$.*

*Proof.* Let $\ell = -1 + \sum_{i=1}^{t} k_i$ and consider the complete graph on $\ell$ vertices, $K_\ell$. Let $T$ be a graph on $n-\ell$ vertices which is $P_k$-semi-saturated. We want to show that $G = K_\ell \cup T$ is $F$-semi-saturated. First note that $K_\ell$ contains a copy of $P_{k_1} \cup \cdots \cup P_{k_{t-1}}$. Therefore, if the addition of an edge in $T$ creates a copy of $P_k$, it creates a copy of $F$ in $G$. Consider adding an edge $e$ with one endpoint in $K_\ell$ and one in $T$. Then this edge forms a path of length $k$ by considering $e$ and $k-1$ vertices in $K_\ell$. $K_\ell$ will contain $P_{k_1} \cup \cdots \cup P_{k_2}$ which creates a copy of $F$ in $G$. Thus $G$ is $F$-semi-saturated. Notice that $G$ has $\mathrm{ssat}(n-\ell, P_k)$ edges in $T$. Additionally, there are $\binom{\ell}{2}$ edges in $K_\ell$. This shows that $ssat(n, F) \leq \mathrm{ssat}(n-\ell, P_k) + c(F)$ where $c(F) = \binom{\ell}{2}$ with $\ell = -1 + \sum_{i=1}^{t} k_i$ for large enough $n$. □

**Corollary 4.** *If $F = P_{k_1} \cup \cdots \cup P_{k_t}$ with $k_1 \geq \cdots \geq k_t = k$, then* $\mathrm{ssat}(n, F) < \mathrm{sat}(n, F)$

*for large enough $n$.*

*Proof.* Define $\ell = -1+\sum_{i=1}^{t} k_i$. In [113], it is shown that $\mathrm{ssat}(n, P_k) < \mathrm{sat}(n, P_k)$ and that the difference between them grows with $n$. Therefore, there exists an integer $N$ so that $\mathrm{ssat}(n-\ell, P_k) + c(F) < \mathrm{sat}(n-\ell, P_k)$ for every $n \geq N$. Since saturation is monotonic in $n$ for paths [112], we have $\mathrm{sat}(n-\ell, P_k) \leq \mathrm{sat}(n, P_k)$. Therefore, $\mathrm{ssat}(n-\ell, P_k)+c(F) < \mathrm{sat}(n-\ell, P_k) \leq \mathrm{sat}(n, P_k)$ for all $n$ greater than $N$. In [115], the authors show that $\mathrm{sat}(n, P_k) \leq \mathrm{sat}(n, F)$. Thus, by these results and Proposition 7, we get that

$$\mathrm{ssat}(n, F) \leq \mathrm{ssat}(n-\ell, P_k) + c(F) < \mathrm{sat}(n, P_k) \leq \mathrm{sat}(n, F)$$

which gives us our result. □

## 6.3 Monotonicity

The results in this section are motivated by the monotonicity of the extremal number. The extremal number has three monotonic properties, as described in Lemma 14.

**Lemma 14.** *[111] For $H$ a subgraph of $H'$ and $\mathcal{F} \subset \mathcal{F}'$,*

1. $\mathrm{ex}(n, H) \leq \mathrm{ex}(n, H')$
2. $\mathrm{ex}(n-1, H) \leq \mathrm{ex}(n, H)$
3. $\mathrm{ex}(n, \mathcal{F}) \geq \mathrm{ex}(n, \mathcal{F}')$

All three types of monotonicity that hold for the extremal numbers do not generally hold for saturation. In Example 4, we show that monotonicity does not hold with respect to the subgraph relation (analogous to part 1 in Lemma 14).

**Example 4.** *Consider the stars $K_{1,s}$ and $K_{1,t}$ for $s < t$. Notice that $K_{1,s}$ is a subgraph of $K_{1,t}$. In this case, it is true that $\operatorname{sat}(n, K_{1,s}) \leq \operatorname{sat}(n, K_{1,t})$ for all $s < t$.*

*On the contrary, consider the graph $K_{1,m}$ and the graph $H_m = K_{1,m} + e$ (examples given in Figure 6.5 and 6.6 for when $m = 3$). Notice that each non-edge is isomorphic, so the choice of $e$ is arbitrary. We will show that even though $K_{1,m}$ is a subgraph of $H_m$, $\operatorname{sat}(n, H_m) \leq \operatorname{sat}(n, K_{1,m})$. Note that $K_{1,n-1}$ is $H$ saturated so*

$$\operatorname{sat}(n, H_m) = n - 1.$$

*To show that $\operatorname{sat}(n, K_{1,m})$ is greater than $\operatorname{sat}(n, H_m)$, we will show that at least $n-1$ vertices in a $K_{1,m}$-saturated graph $G$ must have degree $m-1$ or greater. Suppose two vertices $u$ and $v$ in $G$ have degrees less than $m-1$. Then the addition of the edge $(u, v)$ to $G$ does not create a vertex with degree $m$ and thus does not create a subgraph isomorphic to $K_{1,m}$. Therefore,*

$$\operatorname{sat}(n, K_{1,m}) \geq \frac{(n-1)(m-1)}{2}.$$

*This shows that $\operatorname{sat}(n, H_m) \leq \operatorname{sat}(n, K_{1,m})$ for $m \geq 3$.*

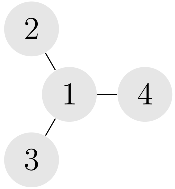


*Figure 6.5: $K_{1,3}$*

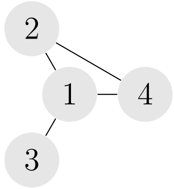


*Figure 6.6: $H_3 = K_{1,3} + e$*

Example 4 shows that, in general, saturation is not monotonic with respect to the subgraph relation. For which graph families is it monotonic? Let $G+H$ denote the *join* of two graphs. That is, $G+H$ has a vertex set and edge set

$$V(G+H)=V(G)\cup V(H)$$

and

$$E(G+H)=E(G)\cup E(H)\cup\{(g,h)|g\in V(G)\text{ and }h\in V(H)\},$$

respectively. In [116], Cameron and Puleo show that if $H$ is a graph and $H'=H+K_1$ then $\mathrm{sat}(H,n)\leq \mathrm{sat}(H',n-1)+n-1$. Here, we show that $\mathrm{ssat}(n,H)\leq \mathrm{ssat}(n,H')$.

**Proposition 8.** *Let $H$ be a graph and $H'=H+K_1$. For all $e\in E(H')$, there exists a copy of $H$ as a subgraph of $H'$ so that $e\in E(H)$.*

*Proof.* Recall that $H'=H+K_1$. Partition the edge set as $E(H')=E(H)\cup E(v)$ where $E(v)$ is every edge in $E(H')$ incident to the dominating vertex $v$. Choose $e$ arbitrarily from $E(H')$. We have our result if $e$ is already contained in $E(H)$. Now suppose that $e\in E(v)$. Let $e=(v,w)$ where $v$ is the dominating vertex of $H'$, and $w$ is contained in $H$. Let $u$ be a vertex in $H$ incident to $w$, and $N(u)$ denote the collection of vertices incident to $u$. Consider the subgraph with vertices $V(H')\setminus u$ and edges $E(H)\setminus E(u)\cup A$ where $A$ is the collection of edges incident to $v$ and every vertex in $N(u)\cap V(H)$. This subgraph is isomorphic to $H$, and since the subgraph contains $e$, we have our result. □

Figure 6.7 and 6.8 show an example of Proposition 8.

**Proposition 9.** *If $H$ is a nonempty graph and $H'=H+K_1$, then* $\mathrm{ssat}(n,H)\leq \mathrm{ssat}(n,H')$.

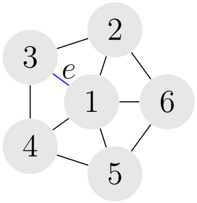


*Figure 6.7:* $W_5 = C_5 + v$.

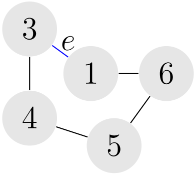


*Figure 6.8: A* $C_5$ *in* $W_5$ *using* $e$.

*Proof.* Let $G$ be an $H'$-semi-saturated graph. Consider $G + e$ for some $e \notin V(G)$. Since $G$ is $H'$-saturated then $G + e$ contains a copy of $H'$ in $G$. By Proposition 8, $H'$ contains a copy of $H$ which contains $e$ as an edge. Therefore, $G + e$ has a subgraph isomorphic to $H$ that contains $e$. Therefore, $G$ is $H$-semi-saturated. □

## 6.4 Discussion

Bushaw, Johnston, and Rombach define that a graph $G$ is *rainbow $H$-saturated* if there is some proper edge coloring of $G$, which is rainbow $H$-free, but where the addition of any edge makes such a rainbow $H$-free coloring impossible [117]. The minimum number of edges among all rainbow $H$-saturated graphs is called the *rainbow $H$-saturation number*, denoted by $\text{sat}^*(n, H)$.

The authors in [117] say it is unknown how the saturation and rainbow-saturation numbers relate since an $H$-saturated graph may not be rainbow $H$-saturated. That is, there could be a proper edge coloring of $G$ that avoids rainbow copies of $H$, even though $G$ has $H$ as a subgraph.

One way to find such a coloring is to find examples where the $H$-saturation and $H$-semi-saturation numbers differ. These the candidates of $H$ to consider when investigating if $\text{sat}^*(n, H) < \text{sat}(n, H)$. In Section 6.2, we show that linear forests $F$

have the property that $\mathrm{ssat}(n, F) < \mathrm{sat}(n, F)$. In future work, we plan to find other examples of this, with the ultimate goal of showing that $\mathrm{sat}^*(n, H) < \mathrm{sat}(n, H)$.

# Index

# Bibliography

[1] Mark Newman. *Networks.* Oxford University Press, 2018.

[2] Stephen P Borgatti, Ajay Mehra, Daniel J Brass, and Giuseppe Labianca. Network analysis in the social sciences. *science*, 323(5916):892–895, 2009.

[3] Björn H Junker and Falk Schreiber. *Analysis of biological networks.* John Wiley & Sons, 2011.

[4] Lidong Zhou and Zygmunt J Haas. Securing ad hoc networks. *IEEE network*, 13(6):24–30, 1999.

[5] Rui Xu and Donald Wunsch. Survey of clustering algorithms. *IEEE Transactions on neural networks*, 16(3):645–678, 2005.

[6] Aristidis Likas, Nikos Vlassis, and Jakob J Verbeek. The global k-means clustering algorithm. *Pattern recognition*, 36(2):451–461, 2003.

[7] John A Hartigan, Manchek A Wong, et al. A k-means clustering algorithm. *Applied statistics*, 28(1):100–108, 1979.

[8] Linton C Freeman et al. Centrality in social networks: Conceptual clarification. *Social network: critical concepts in sociology. Londres: Routledge*, 1:238–263, 2002.

[9] Akrati Saxena and Sudarshan Iyengar. Centrality measures in complex networks: A survey. *arXiv preprint arXiv:2011.07190*, 2020.

[10] Phillip Bonacich. Power and centrality: A family of measures. *American Journal of Sociology*, 92(5):1170–1182, 1987.

[11] Amy N. Langville and Carl D. Meyer. *Google's PageRank and Beyond: The Science of Search Engine Rankings.* Princeton University Press, Princeton, 2011.

[12] Duncan J Watts and Steven H Strogatz. Collective dynamics of ‘small-world’ networks. *nature*, 393(6684):440–442, 1998.

[13] Naoki Masuda, Michiko Sakaki, Takahiro Ezaki, and Takamitsu Watanabe. Clustering coefficients for correlation networks. *Frontiers in neuroinformatics*, 12:7, 2018.

[14] Katherine A Seaton and Lisa M Hackett. Stations, trains and small-world networks. *Physica A: Statistical Mechanics and its Applications*, 339(3-4):635–644, 2004.

[15] Vito Latora and Massimo Marchiori. Is the boston subway a small-world network? *Physica A: Statistical Mechanics and its Applications*, 314(1-4):109–113, 2002.

[16] Lloyd S Shapley. Cores of convex games. *International journal of game theory*, 1(1):11–26, 1971.

[17] N Rama Suri and Yadati Narahari. Determining the top-k nodes in social networks using the Shapley value. In *Proceedings of the 7th international joint conference on Autonomous agents and multiagent systems-Volume 3*, pages 1509–1512, 2008.

[18] Enrico Bozzo, Massimo Franceschet, and Franca Rinaldi. Vulnerability and power on networks. *Network Science*, 3(2):196–226, 2015.

[19] Holger Ingmar Meinhardt. Disentangle the florentine families network by the pre-kernel. *Munich Personal RePEc Archive*, 2021.

[20] Tomasz P Michalak, Karthik V Aadithya, Piotr L Szczepanski, Balaraman Ravindran, and Nicholas R Jennings. Efficient computation of the Shapley value for game-theoretic network centrality. *Journal of Artificial Intelligence Research*, 46:607–650, 2013.

[21] Ramasuri Narayanam and Yadati Narahari. A Shapley value-based approach to discover influential nodes in social networks. *IEEE Transactions on Automation Science and Engineering*, 8(1):130–147, 2010.

[22] Karthik V Aadithya, Balaraman Ravindran, Tomasz P Michalak, and Nicholas R Jennings. Efficient computation of the Shapley value for centrality in networks. In *Internet and Network Economics: 6th International Workshop, WINE 2010, Stanford, CA, USA, December 13-17, 2010. Proceedings 6*, pages 1–13. Springer, 2010.

[23] Jun Ai, Linzhi Li, Zhan Su, Linhua Jiang, and Naixue Xiong. Node-importance identification in complex networks via neighbors average degree. In *2016 Chinese Control and Decision Conference (CCDC)*, pages 1298–1303. IEEE, 2016.

[24] Alex Bavelas. Communication patterns in task-oriented groups. *The Journal of the Acoustical Society of America*, 22(6):725–730, 1950.

[25] Murray A Beauchamp. An improved index of centrality. *Behavioral science*, 10(2):161–163, 1965.

[26] Linton Freeman. Centrality in social networks conceptual clarification. *Social Networks*, 1(3):215–239, 1978.

[27] Rutherford Goldstein and Michael S Vitevitch. The influence of closeness centrality on lexical processing. *Frontiers in psychology*, 8:1683, 2017.

[28] Kazuya Okamoto, Wei Chen, and Xiang-Yang Li. Ranking of closeness centrality for large-scale social networks. *Lecture Notes in Computer Science*, 5059:186–195, 2008.

[29] Linton C Freeman. A set of measures of centrality based on betweenness. *Sociometry*, pages 35–41, 1977.

[30] Loet Leydesdorff. Betweenness centrality as an indicator of the interdisciplinarity of scientific journals. *Journal of the American Society for Information Science and Technology*, 58(9):1303–1319, 2007.

[31] Aisan Kazerani and Stephan Winter. Can betweenness centrality explain traffic flow. In *12th AGILE international conference on geographic information science*, pages 1–9, 2009.

[32] Georg Frobenius, Ferdinand Georg Frobenius, Ferdinand Georg Frobenius, Ferdinand Georg Frobenius, and Germany Mathematician. Über matrizen aus nicht negativen elementen. 1912.

[33] Oskar Perron. Zur theorie der matrices. *Mathematische Annalen*, 64(2):248–263, 1907.

[34] Leo Katz. A new status index derived from sociometric analysis. *Psychometrika*, 18(1):39–43, 1953.

[35] Jack McKay Fletcher and Thomas Wennekers. From structure to activity: Using centrality measures to predict neuronal activity. *International Journal of Neural Systems*, 28(02):1750013, 2018.

[36] Yangyang Liu, Chengli Zhao, Xiaojie Wang, Qiangjuan Huang, Xue Zhang, and Dongyun Yi. The degree-related clustering coefficient and its application to link prediction. *Physica A: Statistical Mechanics and its Applications*, 454:24–33, 2016.

[37] Cesi Cruz, Julien Labonne, and Pablo Querubin. Politician family networks and electoral outcomes: Evidence from the Philippines. *American Economic Review*, 107(10):3006–3037, 2017.

[38] Matthew O Jackson et al. *Social and Economic Networks*, volume 3. Princeton University Press, 2008.

[39] Hunter Rehm, Mona Matar, Puck Rombach, and Lauren McIntyre. The effect of the Katz parameter on node ranking, with a medical application. *arXiv preprint arXiv:2210.06392*, 2022.

[40] Linton Freeman. The development of social network analysis. *A Study in the Sociology of Science*, 1(687):159–167, 2004.

[41] Sambor Guze. Graph theory approach to transportation systems design and optimization. *TransNav: International Journal on Marine Navigation and Safety of Sea Transportation*, 8:57–62, 2014.

[42] Georgios A Pavlopoulos, Maria Secrier, Charalampos N Moschopoulos, Theodoros G Soldatos, Sophia Kossida, Jan Aerts, Reinhard Schneider, and Pantelis G Bagos. Using graph theory to analyze biological networks. *BioData Mining*, 4(1):1–27, 2011.

[43] Kousik Das, Sovan Samanta, and Madhumangal Pal. Study on centrality measures in social networks: a survey. *Social Network Analysis and Mining*, 8(1):1–11, 2018.

[44] Phillip Bonacich. Technique for analyzing overlapping memberships. *Sociological Methodology*, 4:176–185, 1972.

[45] Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. The PageRank citation ranking: Bringing order to the web. Technical report, Stanford InfoLab, 1999.

[46] Ernesto Estrada. Characterization of 3D molecular structure. *Chemical Physics Letters*, 319(5-6):713–718, 2000.

[47] NASA. Nasa software engineering requirements (npr 7150.2d). *Office of the Chief Engineer.*

[48] NASA. Nasa space flight human-system standard volume 1, revision a: Crew health. *NASA Technical Standard*.

[49] Justin Zhan, Sweta Gurung, and Sai Phani Krishna Parsa. Identification of top-$k$ nodes in large networks using Katz centrality. *Journal of Big Data*, 4(1):1–19, 2017.

[50] Eisha Nathan and David A Bader. Approximating personalized Katz centrality in dynamic graphs. In *International Conference on Parallel Processing and Applied Mathematics*, pages 290–302. Springer, 2017.

[51] Eisha Nathan, Geoffrey Sanders, James Fairbanks, David A Bader, et al. Graph ranking guarantees for numerical approximations to Katz centrality. *Procedia Computer Science*, 108:68–78, 2017.

[52] Paul Dagum, Adam Galper, and Eric Horvitz. Dynamic network models for forecasting. In *Uncertainty in Artificial Intelligence*, pages 41–48. Elsevier, 1992.

[53] Omar De la Cruz Cabrera, Mona Matar, and Lothar Reichel. Analysis of directed networks via the matrix exponential. *Journal of Computational and Applied Mathematics*, 355:182–192, 2019.

[54] Ernesto Estrada and Desmond J Higham. Network properties revealed through matrix functions. *SIAM review*, 52(4):696–714, 2010.

[55] Evrim Acar, Daniel M Dunlavy, and Tamara G Kolda. Link prediction on evolving data using matrix and tensor factorizations. In *2009 IEEE International Conference on Data Mining Workshops*, pages 262–269. IEEE, 2009.

[56] Ferenc Béres, Róbert Pálovics, Anna Oláh, and András A Benczúr. Temporal walk based centrality metric for graph streams. *Applied Network Science*, 3(1):1–26, 2018.

[57] Zhengdong Lu, Berkant Savas, Wei Tang, and Inderjit S Dhillon. Supervised link prediction using multiple sources. In *2010 IEEE International Conference on Data Mining*, pages 923–928. IEEE, 2010.

[58] Lauren McIntyre, Lawrence Leinweber, and Jerry G Myers. Dynamic medical risk assessment supported by inference networks. In *Human Research Program Investigators' Workshop (HRP IWS 2020)*, number GRC-E-DAA-TN77298, 2020.

[59] Lauren McIntyre, Jerry G Myers, Lawrence Leinweber, Matthew Prelich, Clara Gasiewski, Michael Lovell, and Raj Prabhu. A model based approach to estimating human spaceflight medical risk. In *Committee on Space Research (COSPAR)*, 2022.

[60] Mark Sh Levin. On combinatorial optimization for dominating sets (literature survey, new models). *arXiv preprint arXiv:2009.09288*, 2020.

[61] Zhuo Liu, Bingwen Wang, and Lejiang Guo. A survey on connected dominating set construction algorithm for. *Inf. Technol. J.*, 9(6):1081–1092, 2010.

[62] Jiguo Yu, Nannan Wang, Guanghui Wang, and Dongxiao Yu. Connected dominating sets in wireless ad hoc and sensor networks–a comprehensive survey. *Computer Communications*, 36(2):121–134, 2013.

[63] Michael R Garey. A guide to the theory of np-completeness. *Computers and Intractability*, 1979.

[64] Johan MM Van Rooij and Hans L Bodlaender. Exact algorithms for dominating set. *Discrete Applied Mathematics*, 159(17):2147–2164, 2011.

[65] Leonid Barenboim, Michael Elkin, and Cyril Gavoille. A fast network-decomposition algorithm and its applications to constant-time distributed computation. *Theoretical Computer Science*, 751:2–23, 2018.

[66] Fei Dai and Jie Wu. An extended localized algorithm for connected dominating set formation in ad hoc wireless networks. *IEEE transactions on parallel and distributed systems*, 15(10):908–920, 2004.

[67] Mohsen Ghaffari and Fabian Kuhn. Derandomizing distributed algorithms with small messages: Spanners and dominating set. In *32nd International Symposium on Distributed Computing (DISC 2018)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2018.

[68] Lujun Jia, Rajmohan Rajaraman, and Torsten Suel. An efficient distributed algorithm for constructing small dominating sets. *Distributed Computing*, 15(4):193–205, 2002.

[69] Ben Liang and Zygmunt J Haas. Virtual backbone generation and maintenance in ad hoc network mobility management. In *Proceedings IEEE INFOCOM 2000. Conference on Computer Communications. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies (Cat. No. 00CH37064)*, volume 3, pages 1293–1302. IEEE, 2000.

[70] Jie Wu and Hailan Li. A dominating-set-based routing scheme in ad hoc wireless networks. *Telecommunication Systems*, 18:13–36, 2001.

[71] Jennifer Yick, Biswanath Mukherjee, and Dipak Ghosal. Wireless sensor network survey. *Computer networks*, 52(12):2292–2330, 2008.

[72] Carlos F García-Hernández, Pablo H Ibarguengoytia-Gonzalez, Joaquín García-Hernández, and Jesús A Pérez-Díaz. Wireless sensor networks and applications: a survey. *IJCSNS International Journal of Computer Science and Network Security*, 7(3):264–273, 2007.

[73] Manish Kumar Singh, Syed Intekhab Amin, Syed Akhtar Imam, Vibhav Kumar Sachan, and Amit Choudhary. A survey of wireless sensor network and its types. In *2018 international conference on advances in computing, communication control and networking (ICACCCN)*, pages 326–330. IEEE, 2018.

[74] Shafiullah Khan, Al-Sakib Khan Pathan, and Nabil Ali Alrajeh. Wireless sensor networks: Current status and future trends. 2016.

[75] Noga Alon and Joel H Spencer. *The probabilistic method*. John Wiley & Sons, 2016.

[76] Vladimir I Arnautov. Estimation of the exterior stability number of a graph by means of the minimal degree of the vertices. *Prikl. Mat. i Programmirovanie*, 11(3-8):126, 1974.

[77] László Lovász. On the ratio of optimal integral and fractional covers. *Discrete mathematics*, 13(4):383–390, 1975.

[78] Charles Payan. Sur le nombre d'absorption d'un graphe simple. 1975.

[79] Enrico Bozzo and Massimo Franceschet. A theory on power in networks. *Communications of the ACM*, 59(11):75–83, 2016.

[80] Noga Alon. Transversal numbers of uniform hypergraphs. *Graphs and Combinatorics*, 6(1):1–4, 1990.

[81] Paul Erdos. On random graphs. *Mathematicae*, 6:290–297, 1959.

[82] Brent N Clark, Charles J Colbourn, and David S Johnson. Unit disk graphs. *Discrete mathematics*, 86(1-3):165–177, 1990.

[83] William Aiello, Fan Chung, and Linyuan Lu. A random graph model for power law graphs. *Experimental mathematics*, 10(1):53–66, 2001.

[84] Fan Chung and Linyuan Lu. The average distances in random graphs with given expected degrees. *Proceedings of the National Academy of Sciences*, 99(25):15879–15882, 2002.

[85] Fan Chung and Linyuan Lu. Connected components in random graphs with given expected degree sequences. *Annals of combinatorics*, 6(2):125–145, 2002.

[86] Gang Lu, Ming-Tian Zhou, Yong Tang, Zhen-Qiang Wu, Guo-Yong Qiu, and Liu Yuan. A survey on exact algorithms for dominating set related problems in arbitrary graphs. *Chinese Journal of Computers*, 33(6):1073–1087, 2010.

[87] Qing Fang, Jie Gao, and Leonidas J Guibas. Locating and bypassing holes in sensor networks. *Mobile Networks and Applications*, 11(2):187–200, 2006.

[88] Prasan Kumar Sahoo, Jang-Ping Sheu, and Kun-Ying Hsieh. Target tracking and boundary node selection algorithms of wireless sensor networks for internet services. *Information Sciences*, 230:21–38, 2013.

[89] Rachid Beghdad and Amar Lamraoui. Boundary and holes recognition in wireless sensor networks. *Journal of Innovation in Digital Ecosystems*, 3(1):1–14, 2016.

[90] Giuseppe Destino and Giuseppe Thadeu Freitas de Abreu. Network boundary recognition via graph-theory. In *2008 5th Workshop on Positioning, Navigation and Communication*, pages 271–275. IEEE, 2008.

[91] Sándor P Fekete, Michael Kaufmann, Alexander Kröller, and Katharina Lehmann. A new approach for boundary recognition in geometric sensor networks. *arXiv preprint cs/0508006*, 2005.

[92] Xu Li, Shibo He, Jiming Chen, Xiaohui Liang, Rongxing Lu, and Sherman Shen. Coordinate-free distributed algorithm for boundary detection in wireless sensor networks. In *2011 IEEE Global Telecommunications Conference-GLOBECOM 2011*, pages 1–5. IEEE, 2011.

[93] Alexander P Giles, Orestis Georgiou, and Carl P Dettmann. Betweenness centrality in dense random geometric networks. In *2015 IEEE International Conference on Communications (ICC)*, pages 6450–6455. IEEE, 2015.

[94] Baoqi Huang, Wei Wu, Guanglai Gao, and Tao Zhang. Recognizing boundaries in wireless sensor networks based on local connectivity information. *International Journal of Distributed Sensor Networks*, 10(7):897039, 2014.

[95] Vasek Chvátal. Sylvester–gallai theorem and metric betweenness. *Discrete & Computational Geometry*, 31:175–195, 2004.

[96] Harold Scott Macdonald Coxeter. Introduction to geometry. 1961.

[97] Kean Fallon, Colin Giles, Hunter Rehm, Simon Wagner, and Nathan Warnberg. Rainbow numbers of $[n]$ for $\sum_{i=1}^{k-1} x_i = x_k$. *Australian Journal of Combinatorics*, 77(1):1–8, 2020.

[98] Kean Fallon, Ethan Manhart, Joe Miller, Hunter Rehm, Nathan Warnberg, and Laura Zinnel. Rainbow numbers of $[m] \times [n]$ for $x_1 + x_2 = x_3$. *arXiv preprint arXiv:2301.10349*, 2023.

[99] Frank P Ramsey. On a problem of formal logic. In *Classic Papers in Combinatorics*, pages 1–24. Springer, 2009.

[100] Issai Schur. Über kongruenz x...(mod. p.). *Jahresbericht der Deutschen Mathematiker-Vereinigung*, 25:114–116, 1917.

[101] Marijn Heule. Schur number five. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018.

[102] Paul Erdős, Miklós Simonovits, and Vera T Sós. Anti-ramsey theorems. 1975.

[103] Xueliang Li, Jianhua Tu, and Zemin Jin. Bipartite rainbow numbers of matchings. *Discrete Mathematics*, 309(8):2575–2578, 2009.

[104] Ruth Haas and Michael Young. The anti-ramsey number of perfect matching. *Discrete Mathematics*, 312(5):933–937, 2012.

[105] Izolda Gorgol. Anti-ramsey numbers in complete split graphs. *Discrete Mathematics*, 339(7):1944–1949, 2016.

[106] Maria Axenovich, Heiko Harborth, Arnfried Kemnitz, Meinhard Möller, and Ingo Schiermeyer. Rainbows in the hypercube. *Graphs and Combinatorics*, 23(2):123–133, 2007.

[107] Erin Bevilacqua, Samuel King, Jürgen Kritschgau, Michael Tait, Suzannah Tebon, and Michael Young. Rainbow numbers for $x_1 + x_2 = kx_3$ in $\mathbb{Z}_n$. *arXiv preprint arXiv:1809.04576*, 2018.

[108] Houssein El Turkey, Jessica Hamm, Anisah Nu'Man, Nathan Warnberg, and Michael Young. Rainbow numbers of $\mathbb{Z}_n$ for $a_1x_1+a_2x_2+a_3x_3 = b$. *INTEGERS*, 20:2, 2020.

[109] Willem Mantel. Problem 28. *Wiskundige Opgaven*, 10(60-61):320, 1907.

[110] Paul Turán. On an external problem in graph theory. *Mat. Fiz. Lapok*, 48:436–452, 1941.

[111] Jill R Faudree, Ralph J Faudree, and John R Schmitt. A survey of minimum saturated graphs. *The Electronic Journal of Combinatorics*, 1000:DS19, 2011.

[112] László Kászonyi and Zs Tuza. Saturated graphs with minimal number of edges. *Journal of Graph Theory*, 10(2):203–210, 1986.

[113] Erika Burr. *A study of saturation number.* PhD thesis, 2017.

[114] Zoltán Füredi and Younjin Kim. Cycle-saturated graphs with minimum number of edges. *Journal of Graph Theory*, 73(2):203–215, 2013.

[115] Guantao Chen, Jill R Faudree, Ralph J Faudree, Ronald J Gould, Michael S Jacobson, and Colton Magnant. Results and problems on saturation numbers for linear forests. *Bulletin of the Institute for Combinatorics and its Applications*, 75:29–46, 2015.

[116] Alex Cameron and Gregory J Puleo. A lower bound on the saturation number, and graphs for which it is sharp. *Discrete Mathematics*, 345(7):112867, 2022.

[117] Neal Bushaw, Daniel Johnston, and Puck Rombach. Rainbow saturation. *Graphs and Combinatorics*, 38(5):166, 2022.