



Graduate Theses, Dissertations, and Problem Reports

2017

A Study of Arc Strong Connectivity of Digraphs

Janet Anderson

Follow this and additional works at: <https://researchrepository.wvu.edu/etd>

Recommended Citation

Anderson, Janet, "A Study of Arc Strong Connectivity of Digraphs" (2017). *Graduate Theses, Dissertations, and Problem Reports*. 5104.

<https://researchrepository.wvu.edu/etd/5104>

This Dissertation is protected by copyright and/or related rights. It has been brought to you by the The Research Repository @ WVU with permission from the rights-holder(s). You are free to use this Dissertation in any way that is permitted by the copyright and related rights legislation that applies to your use. For other uses you must obtain permission from the rights-holder(s) directly, unless additional rights are indicated by a Creative Commons license in the record and/ or on the work itself. This Dissertation has been accepted for inclusion in WVU Graduate Theses, Dissertations, and Problem Reports collection by an authorized administrator of The Research Repository @ WVU. For more information, please contact researchrepository@mail.wvu.edu.

A STUDY OF ARC STRONG CONNECTIVITY OF DIGRAPHS

Janet Anderson

Dissertation submitted
to the Eberly College of Arts and Sciences
at West Virginia University

in partial fulfillment of the requirements for the degree of

Doctor of Philosophy in
Mathematics

Hong-Jian Lai, Ph.D., Chair
Marjorie Darrah, Ph.D.
Elaine Eschen, Ph.D.
Rong Luo, Ph.D.
Kevin Milans, Ph.D.

Department of Mathematics

Morgantown, West Virginia
2017

Keywords: Arc-strong connectivity, subdigraph arc connectivity
extremal digraphs, minimax theorems
Copyright 2017 Janet Anderson

ABSTRACT

A Study of Arc-Strong Connectivity of Digraphs

Janet Anderson

My dissertation research was motivated by Matula and his study of a quantity he called the strength of a graph G , $\bar{\kappa}'(G) = \max\{\kappa'(H) : H \subseteq G\}$. For an integer $k > 0$, a simple graph G with $|V(G)| \geq k+1$ is k -maximal if $\bar{\kappa}'(G) \leq k$ but for any edge $e \in E(G^c)$, $\bar{\kappa}'(G + e) > k$. Mader considered the extremal problem on k -maximal graphs and solved this problem for undirected graphs: If

$$|E(G)| \leq (n - k) + \binom{k}{2}$$

for $n > k \geq 1$ vertices, then G is a k -maximal graph. A natural question arises: Do these ideas hold for directed graphs? In my dissertation, we investigated these ideas for directed graphs and found that these ideas do hold for directed graphs. The main result states: If

$$|A(D)| \leq k(2n - k - 1) + \binom{n - k}{2}$$

for $n \geq k + 1$ vertices, then D is k -maximal.

Matula studied edge connectivity by considering $\kappa'(G) = \min\{|C| : C \text{ is a cut of } G\}$. His work focused on finding dualities between connectivity and cuts. In my dissertation, we extended these ideas to directed graphs and proved two main results:

$$\begin{aligned}\bar{\lambda}(D) &= \max\{\lambda(H) : H \subseteq D\} = \min\{\omega'(S) : S \text{ is a slicing of } D\} \\ \bar{\delta}^+(D) &= \max\{\delta(H) : H \subseteq D\} = \min\{\omega'(X) : X \text{ is a } \delta^+ \text{- slicing}\}.\end{aligned}$$

The quantities $\bar{\delta}^+(D)$, $\bar{\delta}^-(D)$, $\bar{\lambda}(D)$, and $\bar{\kappa}(D)$ can all be computationally determined in polynomial time.

Ariannejad and Tusserkani studied a connectivity measure based on the deletion of some spanning trees from a finite simple graph G . This connectivity measure, *st*-robustness, measures the probability of whether a graph would be disconnected by the

random deletion of spanning trees. They further classify spanning trees of G based on the number of edge-disjoint spanning trees that exist after their deletion. Their main results focus on traditional connectivity measures and the existence of spanning tree types. In my disseration, we develop the terminology for directed graphs and begin to investigate st -robustness for digraphs and the removal of arbitrary spanning arborescences and present this as future work.

DEDICATION

To my parents,

and

Aunt Kay

ACKNOWLEDGEMENTS

First and foremost, I want to thank my advisor Dr. Hong-Jian Lai. From the very first class as an undergraduate, you have encouraged and pushed me to be a better student while being patient and understanding when issues arose. You were exactly what I needed in order to tackle a Ph. D. thesis. Without you, I wouldn't be half the student that I am and this thesis would not exist.

Mom and Dad, thank you for all the support over the years. I love you both. This degree is as much an accomplishment for you as it is me. You instilled in me the values of hard work, dedication, and perseverance that was needed for me to succeed. Thank you for all the encouragement, surprise visits, late night calls while I walked home from campus, financial support, and the time and energy spent packing, moving, and unpacking during this endeavor.

JR, Rachael, Jerry, Brooke, and Chloe, thank you for your love and support. JR, thank you for all the encouragement over the years, I am blessed to call you my brother. Rachael and Jerry, thank you for being the best sister and brother-in-law that anyone could every ask for. You both have supported me over the years with an open home, a hot meal, and reminding me that I'd make it through. Also thank you for my nieces Brooke and Chloe. Brooke and Chloe, thank you for constantly reminding me of the important things in life, bringing laughter and silliness into almost any situation, and the bonding time that I will cherish forever. I love you all.

Renee, thank you for being my best friend. I seriously would not have made it through college and graduate school without you. You have always been there for me, constantly encouraging me to stick with my goals, and to also have a life outside of school and work. Thank you for all the visits, listening to me talk about anything and everything, being my travel companion for many trips abroad, and helping me to be a better person in life.

Brittany, John, Kristy, Kristen, Martin, Renee, Todd, and Will, thank you for everything along the way in graduate school. The late night study groups, the social card nights, the understanding and encouragement that can only come from being a fellow graduate student, and all the talk in between. These memories will not be forgotten.

And last but certainly not least, I want to thank the my committee: Dr. Darrah, Dr. Eschen, Dr. Lai, Dr. Luo, and Dr. Milans. Thank you for your suggestions, your time serving on my committee, and the opportunity to get to know you. I especially want to thank Margie for her support over the years. You were the one who first showed me that a Ph. D. could be achieved and inspired me to set my goals at nothing less. It is only fitting that I ended up with a passion in discrete mathematics like you.

Contents

1	Preliminaries	1
1.1	Notation and Terminology	1
1.2	Extremal Study of k -max Digraphs	4
1.3	Minimax Properties Connectivity of Digraphs	5
1.4	Future Work	7
2	On k-maximal Strength Digraphs	9
2.1	Introduction	9
2.2	Properties of k -maximal Digraphs	13
2.3	Constructive Characterizations	19
2.4	The Extremal Function	24
3	Minimax Properties	30
3.1	Introduction	30
3.1.1	Previous Work on Subgraph Density	33

3.1.2	Previous Work on Graph Connectivity Algorithms	36
3.2	Minimax Theorems in Subdigraph Density Measures	45
3.2.1	Slicing and Proof of Theorem 3.1.2(i)	46
3.2.2	The Study of δ^+ -slicing, δ^- -slicing	48
3.2.3	Maximum Subdigraph Strong Connectivity	51
3.3	Applications to Algorithms	52
4	Future Work	55
4.1	Introduction	55
4.2	Previous Work on Spanning Trees	56
4.3	Future Work	60

Chapter 1

Preliminaries

1.1 Notation and Terminology

Graphs and digraphs have been considered models of various kinds of networks. Connectivity and other reliability measures have been the research topic for many, as seen in the bibliography of this dissertation. This dissertation focuses on the investigation of arc-strong connectivity of digraphs, including maximum subdigraph arc-strong connectivity, minimax dualities related to maximum subdigraph arc-strong connectivity, among other related topics.

Undefined terms and notations will follow [27] for graphs and [8] for digraphs. Let G be a graph. In this thesis, we will assume that all graphs are simple. That is, G is finite, undirected, and contains no loops or multiple edges. A **nontrivial** graph is any graph that contains at least two vertices. We use $V(G)$ to denote the set of vertices of G , and $|V(G)|$ to denote the size of $V(G)$. We use $E(G)$ to denote the set of edges of G , and $|E(G)|$ to denote the size of $E(G)$. For a simple graph G , G^c denotes the complement of G . For graphs H and G , we denote $H \subseteq G$ when H is a subgraph of G . We say H is a subgraph of G , if $V(H)$ is a subset of $V(G)$ and $E(H)$ is a subset of $E(G)$. If $X \subseteq E(G^c)$, then $G + X$ is the simple graph with vertex set $V(G)$ and edge set $E(G) \cup X$. We will use $G + e$ for $G + \{e\}$.

Let $e \in E(G)$. The **contraction** $G/\{e\}$ is the graph obtained from G by deleting e and identifying the two ends of e . For $X \subseteq E(G)$, the contraction G/X is obtained by contracting each edge in X . We often write G/e instead of $G/\{e\}$ and if $H \subseteq G$, we write G/H instead of $G/E(H)$. It is important to note that even though we are only considering simple graphs, the contraction of some edges in G could result in a graph that contains multiple edges and is thus no longer simple. We denote $G \setminus H$ to be the subgraph of G is obtained by deleting all the edges of H from G .

Let G and H be two graphs. We define the **join** of G and H , denoted $G \vee H$ to be

$$\begin{aligned} V(G \vee H) &= V(G) \cup V(H) \text{ and} \\ E(G \vee H) &= E(G) \cup E(H) \cup \{uv : u \in V(G) \text{ and } v \in V(H)\}. \end{aligned}$$

We say that two distinct vertices, u and v , are adjacent if $uv \in E(G)$. For $e \in E(G)$ with $e = xy$, then e is incident to x and y . G is a **complete graph** if for any two distinct vertices, they are adjacent in G . We denote a complete graph by K_n , where $n = |V(G)|$. If $W \subseteq V(G)$, then $G[W]$ denotes the subgraph of G induced by W . The induced subgraph $G[W]$ consists of $V(W)$ and $E(G) \cap \{uv : u, v \in W\}$. For $v \in V(G)$, we use $G - v$ for $G[V(G) - \{v\}]$.

For a vertex $v \in V(G)$, we call the **degree** of v in G , denoted by $d_G(v)$ or $d(v)$ when no confusion arises, the number of edges incident with v . We use $\delta(G)$ to denote the minimum degree of G ; $\delta(G) = \min\{d_G(v) : v \in V(G)\}$. Similarly, $\Delta(G)$ is used to denote the maximum degree of G . Given a graph G and a vertex $v \in V(G)$, we define **the neighborhood** $N_G(v)$ to be the set of all vertices adjacent to v in G .

We define a walk, W , to be an alternating sequence of vertices $(v_0, v_1, \dots, v_{n-1})$ and edges (e_1, e_2, \dots, e_m) such that $W = v_0, e_1, v_1, e_2, \dots, v_{k-1}, e_k, v_k$. For a walk in a simple graph G , we can express the walk by only stating the vertices. Thus W can be expressed by $W = v_0, v_1, \dots, v_{k-1}, v_k$. When the edges of a walk W is distinct, we say that W is a trail. If both the edges and vertices are distinct for a walk W , then W is a path. We define a walk $W = v_0, v_1, \dots, v_k$ to be **closed** if $k \geq 2$ and $v_0 = v_k$. A closed walk $W = v_0, v_1, \dots, v_k$ such that v_0, v_1, \dots, v_{k-1} is a path is defined to be a **cycle**. A graph that contains no cycles is **acyclic**.

We say two vertices $u, v \in V(G)$ are **connected** if there exists a path from u to v , which is denoted as a (u, v) -path. If for every $u, v \in V(G)$, there exists a (u, v) -path, then G is connected. Otherwise, G is **disconnected**. The **edge-connectivity** of G , denoted $\kappa'(G)$ is the minimum number of edges that must be removed in order to disconnect G . The **connectivity** of G , denoted $\kappa(G)$, is the minimum number of vertices that must be removed in order to disconnect G .

For a graph G , we say that G is a tree if for every distinct $u, v \in V(G)$ there is a unique (u, v) -path. Equivalently, a tree may be defined as a connected graph that is acyclic. For a graph G , T is a spanning tree of G if $T \subseteq G$, T is a tree, and $V(T) = V(G)$.

Let $X \subseteq V(G)$. X is a **vertex-cut** of G if $G - X$ is disconnected. Thus $\kappa(G)$ can equivalently be thought of as the minimum size of any vertex-cut in G . If no vertex-cut exists, as in the case of a complete graph, we say $\kappa(G) = |V(G)| - 1$. G is **k -connected** if $\kappa(G) \geq k$. Similarly, the edge-connectivity, $\kappa'(G)$ of G can be defined as the minimum size of any edge-cut of G . G is **k -edge-connected** if $\kappa'(G) \geq k$.

Let D be a directed graph, or digraph for short. In this thesis, D is assumed to be **strict**, which is analogous to a graph being simple. Formally, D is strict if D contains no loops or parallel arcs. We use $V(D)$ to denote the set of vertices of D and $|V(D)|$ to denote the size of $V(D)$. Similarly, $A(D)$ denotes the set of arcs of D , and $|A(D)|$ denotes the size of $A(D)$. Throughout this thesis, we use the notation (u, v) to denote an arc oriented from u to v in a digraph, and $[u, v]$ to denote an arc for which both (u, v) and (v, u) exists. A digraph D is **complete** if D is strict and for every pair u, v of distinct vertices of D , $[u, v] \in A(D)$. The complete digraph on n vertices will be denoted by K_n^* .

We define a directed path, P , to be an alternating sequence of vertices $(v_0, v_1, \dots, v_{n-1})$ and arcs (e_1, e_2, \dots, e_m) such that $P = v_0, e_1, v_1, e_2, \dots, v_{k-1}, e_k, v_k$ where all the vertices and arcs are distinct. We say two vertices $u, v \in V(D)$ are connected if there exists a dipath from u to v . A digraph D is **strongly connected** if for every distinct $u, v \in V(D)$ there exists both an (u, v) -dipath and an (v, u) -dipath. We often refer to D as **strong** when it is strongly connected. We define the **strong connectivity** of D , $\kappa(D)$, to be the minimum number of vertices that must be removed in order for remaining digraph to no longer be strong. Similarly, the **arc-strong connectivity** of a digraph D , denoted $\lambda(D)$, is the

minimum number of arcs that must be removed in order for the remaining digraph to no longer be strong. A **strong component** of a digraph D is a maximal strong subdigraph of D . A strong component H of D is **nontrivial** if $|A(H)| > 0$.

For any disjoint subsets $X, Y \subseteq V(G)$, define

$$\begin{aligned} (X, Y)_G &= \{xy \in E(G) : x \in X, y \in Y\}, \\ \partial_G(X) &= (X, V(G) - X)_G \text{ and} \\ \partial_G^-(X) &= \partial_G(V(G) - X). \end{aligned}$$

When $X = \{v\}$, we often use $\partial_G(v)$ for $\partial_G(\{v\})$. For notational convenience, we often drop the subscript G when no confusion arises for $(X, Y)_G$.

Likewise, for any disjoint subsets $X, Y \subseteq V(D)$, define

$$\begin{aligned} (X, Y)_D &= \{(x, y) \in A(D) : x \in X, y \in Y\}, \\ \partial_D^+(X) &= (X, V(D) - X)_D \text{ and} \\ \partial_D^-(X) &= \partial_D^+(V(D) - X). \end{aligned}$$

For each $v \in V(D)$, we use $\partial_D^+(v)$ for $\partial_D^+(\{v\})$ and $\partial_D^-(v)$ for $\partial_D^-(\{v\})$. The **out-degree** (**in-degree**, respectively) of v in D , is $d_D^+(v) = |\partial_D^+(v)|$ ($d_D^-(v) = |\partial_D^-(v)|$, respectively). We also define,

$$\begin{aligned} N_D^+(v) &= \{u \in V(D) : (v, u) \in A(D)\} \text{ and} \\ N_D^-(v) &= \{u \in V(D) : (u, v) \in A(D)\}. \end{aligned}$$

1.2 Extremal Study of k-max Digraphs

Graph clustering focuses on grouping vertices together based on graph theory properties that tries to minimize the edges between the groups while maximizing the edges within the group. Graph clustering is a useful tool, especially in networking. One graph theory density measure that is particularly useful is connectivity of a graph. In [87], Matula

defines a local connectivity measure, which he called the strength of a graph. Given a graph G , the strength of G is defined as $\bar{\kappa}'(G) = \max\{\kappa'(H) : H \subseteq G\}$. Using the quantity $\bar{\kappa}'(G)$, we have the following definition: For an integer $k > 0$, a simple graph G with $|V(G)| \geq k + 1$ is k -maximal if $\bar{\kappa}'(G) \leq k$ but for any edge $e \in E(G^c)$, $\bar{\kappa}'(G + e) > k$. In [85], Mader considered the extremal problem on k -maximal graphs, proving the following: For $n > k \geq 1$ vertices, if G is a k -maximal graph, then

$$|E(G)| \leq (n - k) + \binom{k}{2}.$$

Lai [79] further studied k -maximal graphs, giving a characterization of all k -maximal graphs.

In this thesis, we will consider this extremal problem for digraphs. Given a digraph D , let $\bar{\lambda}(D) = \max\{\lambda(H) : H \subseteq D\}$. We say that $\bar{\lambda}(D)$ is the strength of D . Let $k \geq 0$ be an integer. A strict digraph D with $|V(D)| \geq k + 1$ is k -maximal if $\bar{\lambda}(D) \leq k$ but for any arc $e \in A(D^c)$, $\bar{\lambda}(D + e) > k$. We prove: For $n \geq k + 1$ vertices, if D is k -maximal, then

$$|A(D)| \leq k(2n - k - 1) + \binom{n - k}{2}.$$

Furthermore, the result is best possible. We also characterize recursively all k -maximal digraphs.

1.3 Minimax Properties Connectivity of Digraphs

Many networking problems can be modeled using graphs, and Matula [87] indicates that many graph problems can be simplified by deconstructing the graph and looking independently on its components. To accomplish this, Matula used the idea of a local connectivity measure, the strength of a graph, $\bar{\kappa}'$.

For a proper non-empty vertex subset X of G , let $C = (X, V(G) - X)$ denote the set of edges in G with an endpoint in X and an endpoint in $V(G) - X$. Define a slicing as a sequence of disjoint edge subsets in the form $C_i = (X, V(G_i) - X)$ for sub-

graph $G_i = G - \bigcup_{j=1}^{i-1} E(C_j)$ for $2 \leq i \leq m$ where $G_1 = G$ and $E(G) - \bigcup_{j=1}^m E(C_j) = \emptyset$. A δ -slicing is defined to be a sequence of disjoint non-empty edge subsets of the form $C_i = (\{v_i\}, V(G_i) - \{v_i\})$ for a subgraph $G_i = G - \bigcup_{j=1}^{i-1} E(C_j)$ for $2 \leq i \leq m$ where $G_1 = G$ and $E(G) - \bigcup_{j=1}^m E(C_j) = \emptyset$. We say that the width of a slicing, denoted $w(Z)$, where $Z = (C_1, C_2, \dots, C_m)$, is the maximum size of any cut in the slicing Z . Matula [89], proved that for any graph G with $|E(G)| \geq 1$, each of the following holds:

- (i) $\bar{\kappa}'(G) = \max\{\kappa'(H) : H \subseteq G\} = \min\{w(Z) : Z \text{ is a slicing of } G\}$.
- (ii) $\bar{\delta}(G) = \max\{\delta(H) : H \subseteq G\} = \min\{w(Z) : Z \text{ is a } \delta\text{-slicing of } G\}$.

For this thesis, we consider these ideas for digraphs. We start with the development of the terminology. For a proper non-empty vertex subset X of D , let $\partial_D^+(X)$ denote the set of arcs in D directed from X to $V(D) - X$. Define a digraph slicing as a sequence of disjoint non-empty arc subsets in the form $\partial_{D_i}^+(X)$ for some subdigraph D_i of D where $D_1 = D$. Similarly, a δ^+ -slicing (δ^- -slicing, respectively) will be defined to be a sequence of disjoint non-empty arc subsets of the form $\partial_{D_i}^+(\{v\})$ ($\partial_{D_i}^-(\{v\})$, respectively) for some subdigraph D_i of D where $D_1 = D$. Let $\mathcal{S}(D)$ be the collection of all slicing of D and let $\mathcal{S}^+(D)$, $\mathcal{S}^-(D)$ be the collection of all δ^+ -slicing of D and all δ^- -slicing of D , respectively.

We prove in this thesis that for a digraph D with $A(D) \neq \emptyset$, each of the following holds:

Assume that $\bar{\lambda}(D) > 0$. Then

$$\begin{aligned} (i) \quad \bar{\kappa}'(D) &= \max\{\min\{|\partial_H^+(X)| : \emptyset \neq X \subset V(H)\} : H \subseteq D\} \\ &= \min\{\max\{|C_i| : 1 \leq i \leq m-1\} : S = (C_1, C_2, \dots, C_m) \in \mathcal{S}(D)\}. \end{aligned}$$

$$\begin{aligned} (ii) \quad \bar{\delta}^+(D) &= \max\{\min\{d_H^+(v) : v \in V(H)\} : H \subseteq D\} \\ &= \min\{\max\{|C_i| : 1 \leq i \leq m\} : S = (C_1, C_2, \dots, C_m) \in \mathcal{S}^+(D)\}. \end{aligned}$$

$$\begin{aligned}
(iii) \quad \bar{\delta}^-(D) &= \max\{\min\{d_H^-(v) : v \in V(H)\} : H \subseteq D\} \\
&= \min\{\max\{|C_i| : 1 \leq i \leq m\} : S = (C_1, C_2, \dots, C_m) \in \mathcal{S}^-(D)\}.
\end{aligned}$$

Further properties derived from these minimax results can be applied to obtained polynomial algorithms to determine the invariants $\bar{\delta}^+(D)$, $\bar{\delta}^-(D)$, $\bar{\lambda}(D)$ and $\bar{\kappa}(D)$.

1.4 Future Work

Complex networks can be modeled as graphs. Robustness of complex networks focuses on the network functioning after some type of failure. In terms of graph theory, robustness focuses on how well connected is a network. Much of the previous work focuses on what Klau and Weiskircher [76] terms as the worst-case. This focuses on the minimum number of edges (vertices) that when removed would disable the network. However, it is not guaranteed that removing an arbitrary minimum number of edges (vertices) will disable the network. A natural question that arises is “What is the probability of removing an arbitrary minimum number of edges (vertices) does disable the network?” Ariannejad and Tusserkani [6], ask this same question but only with a random spanning tree.

Two spanning trees of G , T_1 and T_2 , are edge disjoint spanning trees if $E(T_1) \cap E(T_2) = \emptyset$. The spanning tree packing number, $STP(G)$, is defined as the maximum number of edge disjoint spanning trees contained in G . Let G be a graph with $STP(G) = t$ for some positive integer t . Ariannejad and Tusserkani [6] classify the types of spanning trees by the following definition: A spanning tree T of G is defined to be type ST_i if $STP(G \setminus T) = i - 1$ for $1 \leq i \leq t$. They define $ST_i(G)$ to be the set of all spanning trees of type ST_i and define the cardinality of $ST_i(G)$ to be denoted $\tau_i(G)$. Let $\mathcal{ST}(G)$ be the set of all spanning trees of a graph G and let $\tau(G)$ be the cardinality of the set. The st-robustness of a graph G , denoted $STR(G)$, is the expected value of $STP(G \setminus T)$ summed over every $T \in \mathcal{ST}(G)$. Formally,

$$STR(G) = \sum_i \frac{\tau_i(G)}{\tau(G)} (i - 1) \text{ for } 1 \leq i \leq t.$$

Their main results focus on relationships between classical connectivity measures and defined spanning tree types.

For this thesis, we present terminology needed for digraphs and begin an investigation on which terminology is best suited for future work.

We say that T is an arborescence of D if T is acyclic (i.e. no directed cycles) and there exists a vertex u such that for every $v \in V(D) - \{u\}$, there is a unique (u, v) -dipath. We say that u is the root of T or that T is rooted at u . We say that an arborescence is spanning if $V(T) = V(D)$.

The spanning tree packing number of a graph G , $STP(G)$, can be extended to spanning arborescences. We will define the spanning arborescence packing number of a digraph D , denoted, $SAP(D)$, to be the maximum number of arc disjoint spanning arborescences contained in D . Classifying the types of spanning arborescences is not as easy as classifying the types of spanning trees. One definition for types of spanning arborescences focuses on the idea of free roots. That is, we consider all spanning arborescences. In this case, we say that a spanning arborescence T of D is of type SA_i ($1 \leq i \leq SAP(D)$) if the maximum number of arc disjoint spanning arborescences contained in $D \setminus A$ is $i - 1$ (i.e. $SAP(D \setminus A) = i - 1$). There are other definitions that we will also explore.

And we can similarly define st-robustness of a directed graph D , $STR(D)$, is defined as the expected value of $SAP(D \setminus A)$, when A ranges over all spanning arborescences of D . Formally,

$$STR(D) = \sum_i \frac{\tau_i(D)}{\tau(D)}(i - 1) \text{ for } 1 \leq i \leq SAP(D).$$

where $\tau(D)$ is the number of spanning arborescences contained in D , and $\tau_i(D)$ is the number of spanning arborescences of type SA_i .

Chapter 2

On k -maximal Strength Digraphs

2.1 Introduction

Schaeffer [99] defines graph clustering to be “the task of grouping the vertices of the graph into clusters taking into consideration the edge structure of the graph in such a way that there should be many edges within each cluster and relatively few between the clusters.” Graph clustering relates to many different applications in chemistry, biology, machine learning, and networks to name a few. While the definition given by Schaeffer is not a universal definition, the definition does lead one to ask, how do you perform a clustering operation? An idea from graph theory is to look at the connectivity of the graph and develop a local connectivity measure.

Given a graph G , Matula [89] first studied the quantity

$$\bar{\kappa}'(G) = \max\{\kappa'(H) : H \subseteq G\}.$$

He called $\bar{\kappa}'(G)$ the **strength** of G . The strength of G is a local connectivity property, finding the maximum edge-connectivity of a graph over any of its subgraphs. While the strength of G has been studied by multiple individuals, Mader [85] considered an extremal problem related to $\bar{\kappa}'(G)$. For an integer $k > 0$, a simple graph G with $|V(G)| \geq k + 1$ is k -**maximal** if $\bar{\kappa}'(G) \leq k$ but for any edge $e \in E(G^c)$, $\bar{\kappa}'(G + e) > k$. In [85], Mader

proved the following:

Theorem 2.1.1 (*Mader [85]*) *If G is a k -maximal graph on $n > k \geq 1$ vertices, then*

$$|E(G)| \leq (n - k)k + \binom{k}{2}.$$

Furthermore, this bound is best possible.

In [79], Lai gave a characterization of all k -maximal graphs. Let n, k be integers such that $n > k \geq 1$. Lai defined

$$f(n, k) = \min\{|E(G)| : G \text{ is simple, } |V(G)| = n, \text{ and } k\text{-maximal}\}$$

and

$$F(n, k) = \max\{|E(G)| : G \text{ is simple, } |V(G)| = n, \text{ and } k\text{-maximal}\}$$

Lai also defined $\mathcal{D}(f; n, k)$ to be the set of simple k -maximal graphs such that $|V(G)| = n$, and $|E(G)| = f(n, k)$.

Using this terminology, we see that Mader's Theorem can be rewritten as:

Let $n > k \geq 1$ be integers.

$$F(n, k) = (n - k)k + \binom{k}{2},$$

which is the upperbound for the number of edges a k -maximal graph can have. Lai [79] proves

$$f(n, k) = (n - 1)k - \binom{k}{2} \left\lfloor \frac{n}{k + 2} \right\rfloor$$

which is the lowerbound. We denote $\lfloor x \rfloor$, the **floor** of x , to be the greatest integer that is not larger than x .

Theorem 2.1.2 (*Lai [79]*) *If $n = |V(G)| > k + 1$ and G is a k -maximal graph, then $\overline{\kappa}'(G) = \kappa(G) = k$.*

Let k be an integer. Let H_1 and H_2 be graphs such that their vertex sets are disjoint and at least $|V(H_1)| \geq k$ or $|V(H_2)| \geq k$. We define a **k -edge-join** of H_1 and H_2 , denoted $H_1 \vee_k H_2$ to be:

- (i) $V(H_1 \vee_k H_2) = V(H_1) \cup V(H_2)$
- (ii) $E(H_1 \vee_k H_2) = E(H_1) \cup E(H_2)$
- (iii) We add k new edges e_1, e_2, \dots, e_k such that e_i is incident with a vertex of H_1 and a vertex of H_2 .

We denote the set of all k -edge joins of H_1 and H_2 by $[H_1, H_2]_k$.

Lemma 2.1.3 (*Lai [79]*) *Let k be an integer. Let H_1 be a k -maximal graph and let H_2 be either a K_1 or a k -maximal graph. Then all graphs in $[H_1, H_2]_k$ are k -maximal.*

Let n, k be integers such that $n > k$. Let $\mathcal{M}(k)$ denote the family of graphs such that K_{k+1} is the only graph of order $n+1$ contained in the graph. For a graph G with $|V(G)| = n \geq k+2$, $G \in \mathcal{M}(k)$ if and only if there exists graphs H_1 and H_2 such that $H_i \in \mathcal{M}(k)$ or $H_i = K_1$ ($i = 1, 2$), H_1 and H_2 not both K_1 , and $G \in [H_1, H_2]_k$.

Corollary 2.1.4 (*Lai [79]*) *Let n, k be integers with $n > k$. A graph G with $|V(G)| = n$ is k -maximal if and only if $G \in \mathcal{M}(k)$.*

Let n, k be integers with $n > k+1$. Let $\mathcal{F}(k)$ denote the graph family such that $H(k, 2)$ is the only graph of order $k+2$ contained in the graph. A graph G with $|V(G)| = n > k+2$, $G \in \mathcal{F}(k)$ if and only if there exists H_1 and H_2 such that $|V(H_1)| = n_1$ and $|V(H_2)| = n_2$, $H_i \in \mathcal{F}(k)$ or $H_i = K_1$ ($i = 1, 2$), H_1 and H_2 not both K_1 , with $G \in [H_1, H_2]_k$ and such that

$$\left\lfloor \frac{n}{k+2} \right\rfloor = \left\lfloor \frac{n_1}{k+2} \right\rfloor + \left\lfloor \frac{n_2}{k+2} \right\rfloor.$$

Notice that $\mathcal{F}(k)$ is a subfamily of $\mathcal{M}(k)$.

Theorem 2.1.5 (*Lai [79]*) *For n, k integers with $n > k+1 \geq 2$, we have*

- (i) $f(n, k) = (n-1)k - \binom{k}{2} \left\lfloor \frac{n}{k+2} \right\rfloor$
- (ii) $G \in \mathcal{D}(f; n, k)$ if and only if $G \in \mathcal{F}(k)$ and $|V(G)| = n$.

The purpose of this chapter is to investigate whether the same theorems are true for strict digraphs. Naturally, for a digraph D , we define the strength of a digraph D to be the maximum arc-strong connectivity of any subdigraph of D . Formally,

$$\bar{\lambda}(D) = \max\{\lambda(H) : H \subseteq D\}.$$

Let $k \geq 0$ be an integer. A strict digraph D with $|V(D)| \geq k + 1$ is **k -maximal** if $\bar{\lambda}(D) \leq k$ but for any arc $e \in A(D^c)$, $\bar{\lambda}(D + e) \geq k + 1$.

For positive integer n and k with $n \geq k + 1$, define

$$\mathcal{D}(n, k) = \{D : D \text{ is a strict digraph with } |V(D)| = n \text{ and } D \text{ is } k\text{-maximal}\}.$$

Our goal is to determine $\max\{|A(D)| : D \in \mathcal{D}(n, k)\}$. If $h < k$, we define $\binom{h}{k} = 0$. Our main result is the following:

Theorem 2.1.6 (*Anderson, Lai, Lin, & Xu*) *Let $n \geq k + 1$ be integers. If $D \in \mathcal{D}(n, k)$, then*

$$|A(D)| \leq k(2n - k - 1) + \binom{n - k}{2}.$$

Furthermore, the bound is best possible.

The corollary below is immediate.

Corollary 2.1.7 (*Anderson et al.*) *Let $k > 0$ be an integer and let D be a simple digraph on $n > k$ vertices. If*

$$|A(D)| > k(2n - k - 1) + \binom{n - k}{2},$$

then D must have a subdigraph H such that $\lambda(H) \geq k + 1$.

In the next section, we investigate properties of k -maximal digraphs. In Section 3, we present a constructive characterization of a family of k -maximal subdigraphs $\mathcal{M}(k)$. In the last section, we will prove Theorem 2.1.6 and show that the members in the family $\mathcal{M}(k)$ are precisely the digraphs attaining the upper bound in Theorem 2.1.6.

2.2 Properties of k -maximal Digraphs

In this section, we will present some of our results on the properties of k -maximal digraphs. Throughout this section, let $k \geq 0$ be an integer. Define

$$\mathcal{D}(k) = \cup_{n \geq k+1} \mathcal{D}(n, k).$$

Thus $\mathcal{D}(k)$ is the family of all k -maximal digraphs. Recall that a tournament on n vertices is an orientation of the complete graph K_n on n vertices. The following lemma indicates that the case when $k = 0$ has a clear structure.

Lemma 2.2.1 (*Anderson et al.*) *A digraph $D \in \mathcal{D}(0)$ if and only if D is an acyclic tournament.*

Proof. Suppose first that D is an acyclic tournament. Then $\bar{\lambda}(D) = 0$. For any $e = (u, v) \in A(D^c)$, since D is a tournament, the arc $(v, u) \in A(D)$ and so $D + e$ has a directed cycle of length 2 with vertices $\{u, v\}$. Hence $\lambda(D + e) \geq 1$.

Conversely, we assume that D is a digraph in $\mathcal{D}(0)$. Then since $\bar{\lambda}(D) = 0$, D must be acyclic, and so for any pair of distinct vertices $u, v \in V(D)$, at most one arc in $\{(u, v), (v, u)\}$ is in $A(D)$. It remains to show that for any pair of distinct vertices $u, v \in V(D)$, either (u, v) or (v, u) is in $A(D)$, which proves D is an acyclic tournament.

By contradiction, we assume that for some pair of distinct vertices u and v in $V(D)$, $\{(u, v), (v, u)\} \cap A(D) = \emptyset$. Since $D \in \mathcal{D}(0)$, $D + (u, v)$ has a directed cycle C_1 . Since D is acyclic, C_1 must use the arc (u, v) . Then $P_1 = C_1 - (u, v)$ is a dipath in D from v to u . Similarly, $D + (v, u)$ has a directed cycle C_2 , and so $P_2 = C_2 - (v, u)$ is a dipath of D from u to v . Since $u \neq v$, we note that $P_1 \neq P_2$ with $\{u, v\} \in V(P_1) \cap V(P_2)$. Denote the dipath $P_1 = w_0 w_1 \cdots w_s$ with $v = w_0$ and $u = w_s$, and $P_2 = z_0 z_1 \cdots z_t$ with $u = z_0$ and $v = z_t$, where $s, t > 0$ are integers. If $V(P_1) \cap V(P_2) = \{u, v\}$, then $D[A(P_1) \cup A(P_2)]$ is a directed cycle of D , contrary to the fact that D is acyclic. Hence we assume that there exists a vertex $x \notin \{u, v\}$ such that $x \in V(P_1) \cap V(P_2)$. Let h be the smallest integer with $0 < h < t$ such that $x = z_h \in V(P_1)$. This implies that P_1 has a vertex w_k with $0 < k < s$ such that

$z_h = w_k$. By the choice of h , we have $\{z_0, z_1, z_2, \dots, z_h\} \cap \{w_k, w_{k+1}, \dots, w_s\} = \{u, z_h\}$. Hence $D[\{z_0, z_1, z_2, \dots, z_h\} \cup \{w_k, w_{k+1}, \dots, w_s\}]$ contains a directed cycle of D , contrary to the fact that D is acyclic. This proves the lemma. \square

For any integer $n \geq 0$, let K_n^* denote the complete digraph on n vertices. Thus K_n^* is a strict digraph such that for any pair of distinct vertices $u, v \in V(K_n^*)$, both (u, v) and (v, u) are in $A(K_n^*)$. By definition, we observe the following:

$$K_{k+1}^* \in \mathcal{D}(k) \text{ and if } H \in \mathcal{D}(k) \text{ and } |V(H)| = k + 1, \text{ then } H \cong K_{k+1}^*. \quad (2.1)$$

Recall for a digraph D and disjoint subsets $X, Y \subseteq V(D)$,

$$(X, Y)_D = \{(x, y) \in A(D) : x \in X, y \in Y\}.$$

Further recall that for $X \subseteq V(D)$,

$$\partial_D^+(X) = (X, V(D) - X)_D \text{ and } \partial_D^-(X) = (V(D) - X, X)_D.$$

For each $v \in V(D)$, we define,

$$N_D^+(v) = \{u \in V(D) : (v, u) \in A(D)\} \text{ and } N_D^-(v) = \{u \in V(D) : (u, v) \in A(D)\}.$$

When the digraph D is understood from the context, we sometimes omit the subscript D in the notations above. By the definition of arc-strong connectivity in [8], a digraph D satisfies $\lambda(D) \geq k$ if and only if for any non-empty proper subset $X \subset V(D)$, $|\partial_D^+(X)| \geq k$.

Suppose that $D \in \mathcal{D}(k)$. By definition, $\lambda(D) \leq \bar{\lambda}(D) \leq k$. Therefore, there must be a proper non-empty subset $X \subset V(D)$ such that $|\partial^+(X)| \leq k$. The following lemmas gives more information on D .

Lemma 2.2.2 (*Anderson et al.*) *If $D \in \mathcal{D}(k)$ and if D is not a complete digraph, then for any proper non-empty subset $X \subset V(D)$ such that $|\partial_D^+(X)| \leq k$, each of the following holds:*

- (i) $(X, V(D) - X)_{D^c} \neq \emptyset$.
- (ii) $|\partial_D^+(X)| = k$.
- (iii) $(V(D) - X, X)_D = \{(y, x) : \text{for any } y \in V(D) - X \text{ and for any } x \in X\}$.

Proof. Let $Y = V(D) - X$. Let $|V(D)| = n$. By contradiction, suppose that $(X, Y)_{D^c} = \emptyset$. Then the arcs in $\partial_D^+(X)$ induce an underlying complete bipartite graph with a vertex bipartition $\{X, Y\}$. It follows from $|\partial_D^+(X)| \leq k$ that we must have

$$|X|(n - |X|) = |X| \cdot |Y| \leq k, \text{ and } |X| + |Y| = n \geq k + 1.$$

As the minimum of $|X|(n - |X|)$ must be attained at the boundary point of the domain $1 \leq |X| \leq n - 1$, we observe that

$$k \leq n - 1 \leq |X|(n - |X|) \leq k,$$

it follows that we must have $n = k + 1$, and so by (2.1) that D must be a K_{k+1}^* , contrary to the assumption that D is not a complete digraph. This proves (i).

Therefore there must be an arc $(x, y) \in (X, Y)_{D^c}$. Since $D \in \mathcal{D}(k)$, we have $\bar{\lambda}(D + (x, y)) \geq k + 1$. It follows that $D + (x, y)$ has a subdigraph H such that $\lambda(H) \geq k + 1$. Since $\bar{\lambda}(D) \leq k$, $(x, y) \in A(H)$, then both $X \cap V(H) \neq \emptyset$ and $Y \cap V(H) \neq \emptyset$. As $\partial_H^+(X \cap V(H)) \subseteq \partial_D^+(X) \cup \{(x, y)\}$, we have $k + 1 \leq |\partial_H^+(X \cap V(H))| \leq |\partial_D^+(X) \cup \{(x, y)\}| \leq k + 1$, which implies that $|\partial_D^+(X)| = k$. This proves (ii).

To prove (iii), we again argue by contradiction and assume that for some $x \in X$ and $y \in Y$, $(y, x) \notin A(D)$. Then since $D \in \mathcal{D}(k)$, $\bar{\lambda}(D + (y, x)) \geq k + 1$. It follows that $D + (y, x)$ has a subdigraph H' with $\lambda(H') \geq k + 1$ and with $(y, x) \in A(H')$. Hence both $X \cap V(H') \neq \emptyset$ and $Y \cap V(H') \neq \emptyset$. As $\partial_{H'}^+(X \cap V(H')) \subseteq \partial_D^+(X)$, we have $k + 1 \leq |\partial_{H'}^+(X \cap V(H'))| \leq |\partial_D^+(X)| \leq k$, a contradiction. This proves (iii). \square

Lemma 2.2.3 (*Anderson et al.*) Suppose that $D \in \mathcal{D}(k) - \{K_{k+1}^*\}$. For any proper non-empty subset $X \subset V(D)$ such that $|\partial_D^+(X)| = k$ and $Y = V(D) - X$, one of the following must hold:

- (i) $|X| = 1$ and $D[Y] \in \mathcal{D}(k)$. In particular, $|Y| \geq k + 1$.
- (ii) $|X| \geq k + 1$ and $D[X] \in \mathcal{D}(k)$. In particular, $|Y| = 1$.
- (iii) Both $D[X] \in \mathcal{D}(k)$ and $D[Y] \in \mathcal{D}(k)$. In particular, both $|X| \geq k + 1$ and $|Y| \geq k + 1$.

Proof. We make the following claims:

Claim 1. If $D[X]$ (or $D[Y]$, respectively) is a complete digraph, then $|X| \in \{1, k+1\}$ (or $|Y| \in \{1, k+1\}$, respectively).

By symmetry, it suffices to prove the case when $D[X]$ is a complete digraph. Let $m = |X|$. Then $D[X] = K_m^*$. Since $m-1 = \lambda(K_m^*) \leq \bar{\lambda}(D) \leq k$, we have $m \leq k+1$. If $m \in \{1, k+1\}$, then the claim holds. By contradiction, we assume that $1 < m \leq k$.

By Lemma 2.2.2(i), there exists an arc $(x, y) \in (X, Y)_{D^c}$. As $D \in \mathcal{D}(k)$, $D + (x, y)$ has a subdigraph H with $\lambda(H) \geq k+1$ and $(x, y) \in A(H)$. Hence $X \cap V(H) \neq \emptyset$. Let $X' = X \cap V(H)$ and $m' = |X'|$. Denote $X' = \{x_1, x_2, \dots, x_{m'}\}$. For each x_i , let k_i denote the number of arcs in $(X', V(H) - X')_H$ incident with x_i . Then $\sum_{i=1}^{m'} k_i = |\partial_H^+(X')| \geq \lambda(H) \geq k+1$. Since $\partial_H^+(X') \subseteq \partial_X^+(X) \cup (x, y)$, then $|\partial_H^+(X')| \leq |\partial_X^+(X) \cup \{(x, y)\}| = |\partial_X^+(X)| + 1 = k+1$. Hence $|\partial_H^+(X')| = k+1$.

Since $\lambda(H) \geq k+1$, then $d_H^+(x_i) \geq k+1$. Hence $\sum_{i=1}^{m'} d_H^+(x_i) \geq m'(k+1)$. Since $X' = X \cap V(H)$, then $|X'| \leq |X|$. If $X' = X$, then $D[X'] = K_{m'}^*$ and each x_i has $d_{D[X']}^+(x_i) = m'-1$. Then $\sum_{i=1}^{m'} d_H^+(x_i) \leq \sum_{i=1}^{m'} (k_i + m' - 1)$. Using the fact $\sum_{i=1}^{m'} k_i = k+1$, we see that

$$\sum_{i=1}^{m'} (k_i + m' - 1) = \sum_{i=1}^{m'} k_i + \sum_{i=1}^{m'} (m' - 1) = (k+1) + m'(m' - 1).$$

It follows that

$$m'(k+1) \leq \sum_{i=1}^{m'} d_H^+(x_i) \leq \sum_{i=1}^{m'} (k_i + m' - 1) = (k+1) + m'(m' - 1).$$

Algebraic manipulation leads to $(m' - 1)(k+1) \leq m'(m' - 1)$. If $m' > 1$, then $k+1 \leq m' \leq k$, a contradiction. Therefore, we must have $m' = 1$ and so we may assume that $X' = \{x_1\}$. Since $|\partial_H^+(X')| \geq k+1$, $(x, y) \in \partial_H^+(X')$, and $X' \subseteq X$, it follows that all arcs in $(X, Y)_D$ must be incident with x_1 in X .

Since $m \geq 2$, x_2 is not incident with any arcs in $(X, Y)_D$, and so for any $y \in Y$, $(x_2, y) \in A(D^c)$. As $D \in \mathcal{D}(k)$, $D + (x_2, y)$ must have a subdigraph H'' such that $\lambda(H'') \geq k+1$ and $(x_2, y) \in A(H'')$. Since $m \leq k$, $d_{H''}^+(x_2) \leq |\partial_{D[X]}^+(x_2) \cup \{(x_2, y)\}| =$

$(m-1)+1 \leq k$, contrary to the assumption that $\lambda(H'') \geq k+1$. Therefore if $D[X]$ is a complete digraph, then we must have $|X| \in \{1, k+1\}$. This proves Claim 1.

Claim 2. If $D[X]$ (or $D[Y]$, respectively) is not complete digraph, then $D[X] \in \mathcal{D}(k)$, (or $D[Y] \in \mathcal{D}(k)$, respectively).

Again by symmetry, it suffices to show that $D[X] \in \mathcal{D}(k)$. The case for showing $D[Y] \in \mathcal{D}(k)$ is similar and will be omitted.

Since $D[X]$ is not complete, $A(D[X]^c) \neq \emptyset$. For any $e \in A(D[X]^c)$, since $A(D[X]^c) \subseteq A(D^c)$ and $D \in \mathcal{D}(k)$, it follows by definition of $\mathcal{D}(k)$ that $D+e$ has a subdigraph H_e with $e \in A(H_e)$ and $\lambda(H_e) \geq k+1$. If for any $e \in A(D[X]^c)$, $V(H_e) \subseteq X$, then by definition of $\mathcal{D}(k)$, we have $D[X] \in \mathcal{D}(k)$ and so $|X| \geq k+1$.

Assume by contradiction that there exists an arc $(u, v) \in A(D[X]^c)$ such that $D+(u, v)$ has a subdigraph H with $\lambda(H) \geq k+1$, $(u, v) \in A(H)$, and $V(H) \cap Y \neq \emptyset$. Then as $u, v \in X$ and $V(H) \cap Y \neq \emptyset$, we have $\partial_H^+(X \cap V(H)) = (X \cap V(H), Y \cap V(H))_H \subseteq (X, Y)_D$. It follows that $k+1 \leq \lambda(H) \leq |\partial_H^+(X \cap V(H))| \leq |(X, Y)_D| = |\partial_D^+(X)| = k$, a contradiction. This proves Claim 2.

Claim 3. If $|X| = 1$, then $|Y| \geq k+1$; or if $|Y| = 1$, then $|X| \geq k+1$.

By symmetry, we shall assume that $|X| = 1$ to prove $|Y| \geq k+1$. The other case can be done with symmetric arguments.

By contradiction, we assume that $X = \{x\}$ and $Y = \{y_1, y_2, \dots, y_h\}$ with $h \leq k$. By Lemma 2.2.2(iii), for each i with $1 \leq i \leq h$, we have $(y_i, x) \in A(D)$. Since D is strict and $|(X, Y)_D| = |\partial_D^+(X)| = k$, then $h = k$ and $(x, y_i) \in A(D)$ for every i with $1 \leq i \leq k$. It follows that $D = K_{k+1}^*$, contrary to the assumption that $D \neq K_{k+1}^*$. This proves Claim 3.

With these claims, we now prove the lemma. Assume that both $|X| > 1$ and $|Y| > 1$. If $D[X]$ is a complete digraph, then by Claim 1, $D[X] \cong K_{k+1}^* \in \mathcal{D}(k)$. If $D[X]$ is not a complete digraph, then by Claim 2, $D[X] \in \mathcal{D}(k)$. Similarly we see $D[Y] \in \mathcal{D}(k)$. Since $D[X], D[Y] \in \mathcal{D}(k)$, then $|X| \geq k+1$ and $|Y| \geq k+1$.

If $|X| = 1$, then by Claim 3, $|Y| \geq k+1$. If $D[Y]$ is complete, then by Claim 1,

$D[Y] \cong K_{k+1}^* \in \mathcal{D}(k)$. If $D[Y]$ is not a complete digraph, then by Claim 2, $D[Y] \in \mathcal{D}(k)$.

Similarly, if $|Y| = 1$, using symmetric arguments we have that $|X| \geq k + 1$ and $D[X] \in \mathcal{D}(k)$. This proves the lemma. \square

Definition 2.2.4 Let $H \in \mathcal{D}(k)$ and let $\{v_1, v_2, \dots, v_k\} \subset V(H)$ be a subset of k distinct vertices. Let u be a vertex not in $V(H)$. Define a digraph $[H, K_1]_k$ ($[K_1, H]_k$, respectively) as follows:

- (i) $V([H, K_1]_k) = V([K_1, H]_k) = V(H) \cup \{u\}$.
- (ii) $A([H, K_1]_k) = A(H) \cup \{(v_1, u), (v_2, u), \dots, (v_k, u)\} \cup \left(\bigcup_{v \in V(H)} \{(u, v)\} \right)$.
 $A([K_1, H]_k) = A(H) \cup \{(u, v_1), (u, v_2), \dots, (u, v_k)\} \cup \left(\bigcup_{v \in V(H)} \{(v, u)\} \right)$, respectively).

Note that each of $[H, K_1]_k$ and $[K_1, H]_k$ represents a family of graphs as the set $\{v_1, v_2, \dots, v_k\} \subset V(H)$ may vary. For notational convenience, we often use $[H, K_1]_k$ ($[K_1, H]_k$, respectively) to denote any member in the family $[H, K_1]_k$ ($[K_1, H]_k$, respectively).

Definition 2.2.5 Let $H_1, H_2 \in \mathcal{D}(k)$, and let $\{u_1, u_2, \dots, u_k\} \subset V(H_1)$ be a multiset of $V(H_1)$ and $\{v_1, v_2, \dots, v_k\} \subset V(H_2)$ be a multiset of $V(H_2)$ such that all the arcs $(u_1, v_1), (u_2, v_2), \dots, (u_k, v_k)$ are distinct. Define a digraph $[H_1, H_2]_k$ as follows:

- (i) $V([H_1, H_2]_k) = V(H_1) \cup V(H_2)$.
- (ii) $A([H_1, H_2]_k) = A(H_1) \cup A(H_2) \cup \{(u_1, v_1), (u_2, v_2), \dots, (u_k, v_k)\} \cup \left(\bigcup_{u \in V(H_1), v \in V(H_2)} \{(v, u)\} \right)$.

Note that $[H_1, H_2]_k$ represents a family of digraphs. For notational convenience, we often use $[H_1, H_2]_k$ to denote any member in the family $[H_1, H_2]_k$.

Corollary 2.2.6 (Anderson et al.) Let $D \in \mathcal{D}(k) - \{K_{k+1}^*\}$ be a digraph. Then there exists a non-empty proper subset $X \subseteq V(D)$ such that one of the following holds:

- (i) $|X| = 1$, and for some $H \in \mathcal{D}(k)$, $D = [H, K_1]_k$.
- (ii) $|V(D) - X| = 1$ and for some $H \in \mathcal{D}(k)$, $D = [K_1, H]_k$.
- (iii) For some $H_1, H_2 \in \mathcal{D}(k)$, $D = [H_1, H_2]_k$.

Proof. Since $D \in \mathcal{D}(k) - \{K_{k+1}^*\}$, by Lemma 2.2.2, there exists a proper non-empty subset $X \subset V(D)$ such that $|\partial_D^+(X)| = k$. Let $Y = V(D) - X$. By Lemma 2.2.3, one of the following must hold:

Case 1. Lemma 2.2.3(i) holds.

Then $|X| = 1$ and $D[Y] \in \mathcal{D}(k)$. Let $X = \{x\}$ and $H = D[Y]$. As $|(\{x\}, Y)_D| = k$ and as, by Lemma 2.2.2(iii), for each $y \in Y$, $(y, x) \in A(D)$, it follows by Definition 2.2.4 that $D = [K_1, H]_k$, and so Corollary 2.2.6 (ii) must hold in Case 1.

Case 2. Lemma 2.2.3(ii) holds.

Then $|Y| = 1$ and $D[X] \in \mathcal{D}(k)$. Let $H = D[X]$. With a symmetric argument of Case 1, we conclude that Corollary 2.2.6 (i) must hold in Case 2.

Case 3. Lemma 2.2.3(iii) holds.

Let $H_1 = D[X]$ and $H_2 = D[Y]$. Since $D \in \mathcal{D}(k)$, $|(X, Y)_D| = k$. Since D is not complete, then by Lemma 2.2.2(iii), for each $y \in Y$ and $x \in X$, $(y, x) \in A(D)$. Using this, we conclude that $D = [H_1, H_2]_k$. \square

2.3 Constructive Characterizations

In this section, we present a constructive characterization for a family of strict digraphs in $\mathcal{D}(k)$. This family will be used in next section.

Definition 2.3.1 *Let $\mathcal{M}(k)$ be the family of digraphs such that*

- (i) $K_{k+1}^* \in \mathcal{M}(k)$ and
- (ii) *a digraph $D \neq K_{k+1}^*$ is in $\mathcal{M}(k)$ if and only if for some $H \in \mathcal{M}(k)$, $D = [H, K_1]_k$ or $D = [K_1, H]_k$.*

Suppose that $H_1 \in \mathcal{M}(k)$ with $|V(H_1)| = k + 2$. Then by Definition 2.3.1, there must be a digraph $H_0 \in \mathcal{M}(k)$ and a vertex $v_1 \in V(H_1) - V(H_0)$ such that $H_1 \in$

$\{[\{v_1\}, H_0]_k, [H_0, \{v_1\}]_k\}$. Since $|V(H_0)| = k + 1$, it follows by Definition 2.3.1 that $H_0 \cong K_{k+1}^*$, and there exists a unique vertex $v_0 \in V(H_0)$ such that

$$d_{H_1}^-(v_0) = k \text{ if } H_1 = [\{v_1\}, H_0]_k; \text{ or } d_{H_1}^+(v_0) = k \text{ if } H_1 = [H_0, \{v_1\}]_k. \quad (2.2)$$

Throughout the rest of this section, we will use v_0 to denote the only vertex in H_0 satisfying (2.2).

Lemma 2.3.2 (*Anderson et al.*) *Let $D \in \mathcal{M}(k)$ with $n = |V(D)|$ vertices. Each of the following holds:*

- (i) D has a subdigraph $H_0 \cong K_{k+1}^*$.
- (ii) Suppose that $n > k + 1$. Let v_0 denote the unique vertex in H_0 satisfying (2.2), $V(D) - V(H_0) = \{v_1, v_2, \dots, v_h\}$ with $h = n - (k + 1)$. Then D has a sequence of subdigraphs H_1, H_2, \dots, H_h such that for each i with $1 \leq i \leq h$,

$$H_i \in \{[\{v_i\}, H_{i-1}]_k, [H_{i-1}, \{v_i\}]_k\}, \text{ and } D = H_h. \quad (2.3)$$

- (iii) Suppose that $D \neq K_{k+1}^*$. Then for each i with $1 \leq i \leq h$, H_i has a unique vertex u_1 and another vertex u_2 with the following properties:

$$\left\{ \begin{array}{ll} d_{H_i}^+(u_1) = k \text{ and } d_{H_i}^-(u_1) \geq k + 1 & \text{if } H_i = [\{v_i\}, H_{i-1}] \\ d_{H_i}^+(u_2) \geq k + 1 \text{ and } d_{H_i}^-(u_2) \geq k & \text{if } H_i = [\{v_i\}, H_{i-1}] \\ d_{H_i}^-(u_1) = k \text{ and } d_{H_i}^+(u_1) \geq k + 1 & \text{if } H_i = [H_{i-1}, \{v_i\}] \\ d_{H_i}^-(u_2) \geq k + 1 \text{ and } d_{H_i}^+(u_2) \geq k & \text{if } H_i = [H_{i-1}, \{v_i\}] \\ \min\{d_{H_i}^+(v), d_{H_i}^-(v)\} \geq k + 1 & \text{if } v \in V(H_i) - \{u_1, u_2\} \end{array} \right. \quad (2.4)$$

- (iv) $u_1 = v_i$ and we can always choose $u_2 \in \{v_0, v_1, \dots, v_{i-1}\}$. Furthermore, if for some $0 \leq j \leq i - 1$, $\min\{d_{H_i}^+(v_j), d_{H_i}^-(v_j)\} = k$, then $u_2 = v_j$.

Proof. (i) follows from Definition 2.3.1(i).

For the rest of the proof in this lemma, we assume that

$$V(D) - V(H_0) = \{v_1, v_2, \dots, v_h\} \text{ with } h = n - (k + 1).$$

(ii) We argue by induction on $h = |V(D)| - |V(H_0)|$. Since $n > k + 1$, then $h \geq 1$. Assume $h = 1$, then $|V(D)| = k + 2$. By Definition 2.3.1, there exists a digraph $H_1 \in \mathcal{M}(k)$ and a vertex $v_1 \in V(D) - V(H_0)$ such that $D \in \{[H_0, \{v_1\}]_k, [\{v_1\}, H_0]_k\}$. Then D has a sequence of subdigraphs $H_1 = H_h$ such that $H_1 \in \{[H_0, \{v_1\}]_k, [\{v_1\}, H_0]_k\}$, and $D = H_h = H_1$.

Assume $h > 1$. By Definition 2.3.1(ii), for some $H \in \mathcal{M}(k)$ and vertex $v_h \in V(D)$, $D = [H, \{v_h\}]_k$ or $D = [\{v_h\}, H]_k$. Since $v_h \notin V(H_0)$, we observe that $H_0 \subseteq H$, and $|V(H)| - |V(H_0)| = |V(D)| - |V(H_0)| - 1 = h - 1$. By the inductive hypothesis, H has a sequence of subdigraphs H_1, H_2, \dots, H_{h-1} such that for each i with $1 \leq i \leq h - 1$, (2.3) holds with $H_{h-1} = H$. Since $D = [H, \{v_h\}]_k$ or $D = [\{v_h\}, H]_k$, then $D = [H_{h-1}, \{v_h\}]_k = H_h$ or $D = [\{v_h\}, H_{h-1}]_k = H_h$. Thus D has a sequence of subdigraphs H_1, H_2, \dots, H_h such that for each i with $1 \leq i \leq h$, (2.3) holds with $H_h = D$. Hence (ii) is proven by induction.

(iii) and (iv) We again by argue by induction on i with $h \geq i \geq 1$. Assume $h = 1$. Let $H_1 = D$. Then $V(H_1) - V(H_0) = \{v_1\}$ and $1 = n - (k + 1)$. Since $H_0 \cong K_{k+1}^*$, $d_{H_0}^+(u) = d_{H_0}^-(u) = k$ for every $u \in V(H_0)$. Assume $V(H_0) = \{v_0, x_1, x_2, \dots, x_k\}$. By Definition 2.3.1(ii), $H_1 \in \{[\{v_1\}, H_0]_k, [H_0, \{v_1\}]_k\}$.

Assume that $H_1 \in [\{v_1\}, H_0]_k$. By definition, $d_{H_1}^-(v_0) = k$. Since $d_{H_1}^-(v_0) = d_{H_0}^-(v_0)$, then $(v_1, v_0) \notin A(H_1)$. Thus $N_{H_1}^+(v_1) = \{x_1, x_2, \dots, x_k\}$. Thus $d_{H_1}^-(x_i) = k + 1$ for every $i \in \{1, 2, \dots, k\}$. From the definition of $[\{v_1\}, H_0]_k$, $d_{H_1}^-(v_1) = k + 1$, and $d_{H_1}^+(u) = d_{H_0}^+(u) + 1$ for every $u \in V(H_0)$. Thus $d_{H_1}^+(u) = k + 1$ for every $u \in V(H_0)$.

Let $u_1 = v_1$ and $u_2 = v_0$. Then $\min\{d_{H_1}^+(v), d_{H_1}^-(v)\} \geq k + 1$ for every $v \in V(H_1) - \{u_1, u_2\}$. Hence $u_1 = v_1$ and $u_2 = v_0$ satisfies (2.4) with $i = 1$.

If $H_1 \in [H_0, \{v_1\}]_k$, using a symmetric argument, we find that $u_1 = v_1$ and $u_2 = v_0$ satisfies (2.4) with $i = 1$. Thus (iii) and (iv) hold for $i = 1$.

Next we assume that $h \geq i > 2$, and both (iii) and (iv) hold for smaller values of i . By (2.3) and by symmetry, we may assume that $H_i = [\{v_i\}, H_{i-1}]_k$. By induction, H_{i-1} satisfies the conclusions of Lemma 2.3.2 (iii) and (iv). Therefore, H has two vertices u'_1, u'_2 satisfying (2.4), $u'_1 = v_{i-1}$, and $u'_2 \in \{v_0, v_1, \dots, v_{i-2}\}$.

Since $H_i = [\{v_i\}, H_{i-1}]_k$, by Definition 2.2.4, for any $y \in V(H_{i-1})$, $(y, v_i) \in A(H_i)$. It follows that $d_{H_i}^+(u'_1) = k + 1$. Thus setting $u_1 = v_i$ and $u_2 = u'_2$, we conclude that u_1 and u_2 satisfy (2.4). This proves both (iii) and (iv). \square

Let $D \in \mathcal{M}(k) - \{K_{k+1}^*\}$. By Lemma 2.3.2, D has a sequence of subdigraphs H_0, H_1, \dots, H_h satisfying (2.3). For $h \geq i \geq 1$, let v_i denote the only vertex in $V(H_i) - V(H_{i-1})$. Thus each $D \in \mathcal{M}(k) - \{K_{k+1}^*\}$ is associated with a sequence

$$(H_0, v_0), (H_1, v_1), \dots, (H_h, v_h). \quad (2.5)$$

We call (2.5) a **construction sequence** of D . For each $i > 0$, define

$$S_i = \begin{cases} N_{H_i}^+(v_i) & \text{if } H_i = [\{v_i\}, H_{i-1}]_k \\ N_{H_i}^-(v_i) & \text{if } H_i = [H_{i-1}, \{v_i\}]_k \end{cases}.$$

Then $S_i \subseteq V(H_{i-1})$. The next lemma gives more information about D .

Lemma 2.3.3 (*Anderson et al.*) *Let $D \in \mathcal{M}(k) - \{K_{k+1}^*\}$ and let (2.5) be a construction sequence of D . If there exists an i such that for all $h \geq j > i$, $v_i \notin \cup_{j>i}^h S_j$, then $D - v_i \in \mathcal{M}(k)$.*

Proof. We argue by induction on $h \geq 1$. Assume that $h = 1$. Then $i \in \{0, 1\}$. If $i = 1$, then $D - v_1 = H_0 \cong K_{k+1}^* \in \mathcal{M}(k)$; if $i = 0$, then by Definition 2.2.4, $D - v_0 \cong K_{k+1}^* \in \mathcal{M}(k)$. Hence the lemma holds for $h = 1$.

Assume that $h > 1$ and Lemma 2.3.3 holds for smaller values of h . Let (2.5) denote a construction sequence of D . Then $H_{h-1} \in \mathcal{M}(k)$. If $i = h - 1$, then by (2.5), $H_{h-1} - \{v_{h-1}\} = H_{h-2} \in \mathcal{M}(k)$. Since $v_{h-1} \notin S_h$, it follows that $D - \{v_{h-1}\} \in \{[\{v_h\}, H_{h-2}]_k, [H_{h-2}, \{v_h\}]_k\} \in \mathcal{M}(k)$, and the lemma is proved by induction in this case. Now assume that $h - 1 > i$. Then by induction, $H = H_{h-1} - \{v_i\} \in \mathcal{M}(k)$, and so by Definition 2.2.4, $D - v_i \in \{[\{v_h\}, H]_k, [H, \{v_h\}]_k\}$. Thus $D - v_i \in \mathcal{M}(k)$. and the lemma is proved by induction in this case as well. \square

Lemma 2.3.4 (*Anderson et al.*) *Let $k \geq 1$ be an integer. Then $\mathcal{M}(k) \subseteq \mathcal{D}(k)$.*

Proof. Let $D \in \mathcal{M}(k)$ be a digraph with $n = |V(D)| \geq k + 1$. We now argue by induction on $n \geq k + 1$ to show that $\mathcal{M}(k) \subseteq \mathcal{D}(k)$.

If $n = k + 1$, then $D \cong K_{k+1}^*$. Since $K_{k+1}^* \in \mathcal{D}(k)$, we are finished.

Assume that $n > k + 1$ and for any digraph $D' \in \mathcal{M}(k)$ with $|V(D')| \leq n - 1$, $D' \in \mathcal{D}(k)$. Since $n > k + 1$, D is not a complete digraph. Then $h = n - (k + 1) > 0$. Let (2.5) denote a construction sequence of D , $u_1 = u_1^h$ and $u_2 = u_2^h$ be the two vertices satisfying Lemma 2.3.2(iii). Let $a = (u, v) \in A(D^c)$. We shall show that $\bar{\lambda}(D + a) \geq k + 1$.

If $v_h \notin \{u, v\}$, then $a = (u, v) \in A(H_{h-1}^c)$. By induction, $H_{h-1} \in \mathcal{D}(k)$, and so $\bar{\lambda}(H_{h-1} + a) \geq k + 1$. Since $H_{h-1} \subset H_h = D$, then $\bar{\lambda}(D + a) \geq \bar{\lambda}(H_{h-1} + a) \geq k + 1$.

Assume that $v_h \in \{u, v\}$. Without loss of generality, we assume that $v_h = v$. By Lemma 2.3.2, $u_1 = v$, and $u_2 \in \{v_0, v_1, \dots, v_{h-1}\}$.

Case 1. $\min\{d_D^+(u_2), d_D^-(u_2)\} = k$.

If $u \neq u_2$, then we may assume that $u_2 = v_j$ for some $0 \leq j \leq h - 1$. Since $\min\{d_D^+(u_2), d_D^-(u_2)\} = k$, by Definition 2.2.4, $v_j \notin \cup_{i>j}^h S_i$, and so by Lemma 2.3.3, $D - v_j \in \mathcal{M}(k)$. By induction, $D - v_j \in \mathcal{D}(k)$. As $v_j \notin \{u, v\}$, $a \in A(D - v_j)^c$ and since $D - v_j \in \mathcal{M}(k)$, we have $\bar{\lambda}(D + a) \geq \bar{\lambda}((D - v_j) + a) \geq k + 1$.

Case 2. $\min\{d_D^+(u_2), d_D^-(u_2)\} > k$.

By Lemma 2.3.2, for any vertex $z \in V(D) - \{v\}$, $\min\{d_D^+(z), d_D^-(z)\} > k$. Since $(u, v) \in A(D^c)$ and since $u_1 = v_h = v$, by Definition 2.2.4, we must have $D = [H_{h-1}, \{v\}]_k$ and $d_D^-(v) = k$. Thus

$$\min\{d_{D+a}^+(z), d_{D+a}^-(z)\} \geq k + 1, \text{ for any } z \in V(D + a) = V(D). \quad (2.6)$$

We shall show that $\lambda(D + a) \geq k + 1$. Let $\emptyset \neq X \subset V(D) = V(D + a)$ be a non-empty proper subset. We shall show that $|\partial_D^+(X)| \geq k + 1$. Let $Y = V(D) - X$. By (2.6), we may assume that $|X| \geq 2$ and $|Y| \geq 2$.

Subcase 2.1. $V(H_0) \subseteq X$ or $V(H_0) \subseteq Y$.

By symmetry, we assume that $V(H_0) \subseteq X$. The proof for $V(H_0) \subseteq Y$ is similar and

will be omitted. As $|Y| \geq 2$, $Y \cap \{v_1, v_2, \dots, v_h\} \neq \emptyset$. Pick $v_i \in Y$ such that i is the smallest with $i \geq 1$. Then $V(H_{i-1}) \subseteq X$. Assume first that $H_i = [\{v_i\}, H_{i-1}]_k$. Then by the choice of i , we have $N_{H_i}^-(v_i) \subseteq X$, and so by Definition 2.2.4, $|\partial_D^+(X)| \geq |\partial_{H_i}^-(v_i)| = |N_{H_i}^-(v_i)| \geq |V(H_0)| = k + 1$.

Assume then that $H_i = [H_{i-1}, \{v_i\}]_k$. Then by the choice of i , we have $N_{H_i}^+(v_i) \subseteq X$. Since $|Y| \geq 2$, we can pick $v_j \in Y - \{v_i\}$ such that $j > i$ is the smallest. By the choice of j and by Definition 2.2.4, there exists a vertex $z \in X$ such that $(v_j, z) \in A(H_j)$. It follows that $\partial_{H_i}^+(v_i) \cup \{(v_j, z)\} \subseteq \partial_D^+(X)$, and so $|\partial_D^+(X)| \geq |\partial_{H_i}^+(v_i)| + 1 \geq k + 1$. This proves this subcase.

Subcase 2.2. $V(H_0) \cap X \neq \emptyset$ and $V(H_0) \cap Y \neq \emptyset$.

Since $h > 0$, we may assume that $v_1 \in X$. (The case when $v_1 \in Y$ can be proved similarly). As $|Y| \geq 2$, we either have $|V(H_0) \cap Y| \geq 2$ or $Y - V(H_0) \neq \emptyset$. If $H_1 = [H_0, \{v_1\}]_k$, then by Definition 2.2.4, $|(V(H_0) \cap Y, \{v_1\})_D| \geq 1$. It follows that $|\partial_D^+(X)| \geq |(V(H_0) \cap X, V(H_0) \cap Y)_D| + |(V(H_0) \cap Y, \{v_1\})_D| \geq k + 1$.

Hence we may assume that $H_1 = [\{v_1\}, H_0]_k$. If $V(H_0) \cap Y \neq \{v_0\}$, then $|(\{v_1\}, V(H_0) \cap Y)_D| \geq 1$, and so $|\partial_D^+(X)| \geq |(V(H_0) \cap X, V(H_0) \cap Y)_D| + |(\{v_1\}, V(H_0) \cap Y)_D| \geq k + 1$.

Hence we may assume that $V(H_0) \cap Y = \{v_0\}$. Since $|Y| \geq 2$, there must be a smallest $i > 1$ such that $v_i \in Y$. By the minimality of i , $V(H)_{i-1} - \{v_0\} \subseteq X$. By Definition 2.2.4, $|(V(H)_{i-1} - \{v_0\}, \{v_i\})_D| \geq 1$, and so $|\partial_D^+(X)| \geq |(V(H_0) \cap X, V(H_0) \cap Y)_D| + |(V(H_{i-1}) - \{v_0\}, \{v_i\})_D| \geq k + 1$. This proves Subcase 2.2, and completes the proof of the lemma. \square

2.4 The Extremal Function

In this section, we shall determine the extremal function as shown in Theorem 2.4.1 below. This clearly implies Theorem 2.1.6.

Theorem 2.4.1 (*Anderson et al.*) *Let $D \in \mathcal{D}(n, k)$. Then*

$$|A(D)| \leq k(2n - k - 1) + \binom{n - k}{2}. \quad (2.7)$$

Furthermore, each of the following holds:

- (i) *If $k = 1$, then every digraph $D \in \mathcal{D}(1)$ satisfies equality in (2.7).*
- (ii) *If $k \geq 2$, then a digraph $D \in \mathcal{D}(k)$ satisfies equality in (2.7) if and only if $D \in \mathcal{M}(k)$.*

Proof. We first prove (2.7) by induction on $n = |V(D)|$.

If $n = k + 1$, then $D = K_{k+1}^*$. Since $|A(K_{k+1}^*)| = k(k + 1)$, we observe that (2.7) holds when $n = k + 1$.

Assume that $n > k + 1$ and (2.7) holds for smaller values of n . Since $n > k + 1$, D cannot be a complete digraph. By Corollary 2.2.6, one of the three conclusions of Corollary 2.2.6 must hold:

Claim 1. If Corollary 2.2.6 (i) or (ii) holds, then (2.7) holds.

Without loss of generality, we assume that $D = [H, K_1]_k$ for some $H \in \mathcal{D}(k)$ with $V(K_1) = \{v\}$. $|V(H)| = n - 1$. By Definition 2.2.4, $|\partial_D^+(v)| = k$ and $|\partial_D^-(v)| = n - 1$. Then $|A(D)| = |A(H)| + k + (n - 1)$. Since $|V(H)| < n$, we can use the inductive hypothesis and obtain:

$$\begin{aligned} |A(D)| &\leq k(2(n - 1) - k - 1) + \binom{(n - 1) - k}{2} + k + (n - 1) \\ &\leq k(2n - k - 3) + \left(\sum_{i=1}^{n-k-2} i \right) + k + (n - 1) \\ &= k(2n - k - 3) + \left(\sum_{i=1}^{n-k-2} i \right) + k + (n - 1) + 2k - 2k \\ &= k(2n - k - 3) + 2k + \left(\sum_{i=1}^{n-k-2} i \right) + (n - k - 1) \end{aligned}$$

$$\begin{aligned}
&= k(2n - k - 3) + 2k + \left(\sum_{i=1}^{n-k-2} i \right) + (n - k - 1) \\
&= k(2n - k - 1) + \left(\sum_{i=1}^{n-k-1} i \right) \\
&= k(2n - k - 1) + \binom{n-k}{2}
\end{aligned}$$

Claim 2. If Corollary 2.2.6 (iii) holds, then (2.7) holds. Furthermore, if $k \geq 2$, then (2.7) holds with strict inequality.

Without loss of generality, assume that $D = [H_1, H_2]_k$ for some $H_1, H_2 \in \mathcal{D}(k)$. Let $n_1 = |V(H_1)|$ and $n_2 = |V(H_2)|$. By Definition 2.2.4, $|\partial_D^+(H_1)| = k$ and $|\partial_D^-(H_1)| = n_1 n_2$. Then $|A(D)| = |A(H_1)| + k + n_1 n_2 + |A(H_2)|$. Since $|V(H_1)|, |V(H_2)| < n$, we can use the inductive hypothesis and obtain:

$$\begin{aligned}
|A(D)| &\leq k(2n_1 - k - 1) + \binom{n_1 - k}{2} + k + n_1 n_2 + k(2n_2 - k - 1) + \binom{n_2 - k}{2} \\
&= k(2n_1) - k^2 - k + k(2n_2) - k^2 - k + k + n_1 n_2 + \binom{n_1 - k}{2} + \binom{n_2 - k}{2} \\
&= (k(2n_1) + k(2n_2) - k^2 - k) - k^2 + n_1 n_2 + \binom{n_1 - k}{2} + \binom{n_2 - k}{2} \\
&= k(2n_1 + 2n_2 - k - 1) - k^2 + n_1 n_2 + \binom{n_1 - k}{2} + \binom{n_2 - k}{2} \\
&= k(2n - k - 1) - k^2 + n_1 n_2 + \binom{n_1 - k}{2} + \binom{n_2 - k}{2} \\
&\quad + \binom{n - k}{2} - \binom{n - k}{2}
\end{aligned}$$

$$\begin{aligned}
&= k(2n - k - 1) + \binom{n - k}{2} \\
&\quad - k^2 + n_1 n_2 + \binom{n_1 - k}{2} + \binom{n_2 - k}{2} - \binom{n - k}{2}
\end{aligned}$$

We further observe the following:

$$\begin{aligned}
&2 \left[n_1 n_2 + \binom{n_1 - k}{2} + \binom{n_2 - k}{2} - \binom{n - k}{2} \right] \\
&= 2n_1 n_2 + (n_1 - k)(n_1 - k - 1) + (n_2 - k)(n_2 - k - 1) - (n - k)(n - k - 1) \\
&= 2n_1 n_2 + \left[\sum_{i=1}^2 (n_i^2 - 2n_i k + k^2 - n_i + k) \right] - (n^2 - 2nk + k^2 - n + k) \\
&= (n_1 + n_2)^2 - 2nk + 2k^2 - (n_1 + n_2) + 2k - n^2 + 2nk - k^2 + n - k = k(k + 1).
\end{aligned}$$

Now, we have $-k^2 + \frac{k(k+1)}{2} = -\frac{k^2-k}{2} \leq 0$, and so

$$|A(D)| \leq k(2n - k - 1) + \binom{n - k}{2} - \frac{k^2 - k}{2}.$$

This implies Claim 2.

Claim 3. If $k \geq 2$, and if $D \in \mathcal{D}(k)$ satisfies equality in (2.7), then $D \in \mathcal{M}(k)$.

Let $D \in \mathcal{D}(k)$ be a digraph satisfying equality in (2.7). We argue by induction on $n = |V(D)| \geq k + 1$ to prove Claim 3. If $n = k + 1$, then $D = K_{k+1}^* \in \mathcal{M}(k)$. Hence we assume that $n > k + 1$ and that Claim 3 holds for smaller values of n .

If D satisfies Corollary 2.2.6(iii), then by Claim 2, D cannot satisfy equality in (2.7), contrary to the assumption of Claim 3. Hence D must satisfy Corollary 2.2.6(i) or (ii). By symmetry, we assume that for some digraph $H \in \mathcal{D}(k)$ and vertex $v \in V(D)$, we have $D = [\{v\}, H]_k$. By Definition 2.2.4, we have $|A(D)| = |A(H)| + k + n - 1$. As D satisfies

equality in (2.7), this implies that

$$\begin{aligned}
|A(H)| &= k(2n - k - 1) + \binom{n - k}{2} - (n + k - 1) \\
&= 2nk - k^2 - k + \binom{n - k}{2} - (n + k - 1) - 2k + 2k \\
&= 2nk - 2k - k^2 - k + \left(\sum_{i=1}^{n-k-1} i \right) - (n + k - 1) + 2k \\
&= k(2n - 2) - k - 1 + \left(\sum_{i=1}^{n-k-1} i \right) - (n - k - 1) \\
&= k(2(n - 1) - k - 1) + \left(\sum_{i=1}^{n-k-2} i \right) \\
&\quad - k(2(n - 1) - k - 1) + \binom{(n - 1) - k}{2}.
\end{aligned}$$

It follows by induction that $H \in \mathcal{M}(k)$. By Definition 2.3.1, $D \in \mathcal{M}(k)$, and so Claim 3 is proved by induction.

Claim 4. If $k = 1$, then for any $D \in \mathcal{D}(k)$, we have

$$|A(D)| = k(2n - k - 1) + \binom{n - k}{2} = 2(n - 1) + \frac{(n - 1)(n - 2)}{2}. \quad (2.8)$$

We again argue by induction on n . Since $|A(K_2^*)| = 2$, (2.8) holds when $n = k + 1 = 2$. Assume that $n > 2$ and (2.8) holds for smaller values of n . Hence D is not a complete digraph. By Corollary 2.2.6, we have the following observations:

Suppose that Corollary 2.2.6(i) or (ii) holds. Then $D \in \{[K_1, H]_k, [H, K_1]_k\}$ for some $H \in \mathcal{D}(1)$. By Definition 2.2.4, $|A(D)| = |A(H)| + 1 + (n - 1)$. By the inductive hypothesis,

$$\begin{aligned}
|A(D)| &= 2((n - 1) - 1) + \frac{((n - 1) - 1)((n - 1) - 2)}{2} + 1 + (n - 1) \\
&= 2((n - 1) - 1) + n + \frac{(n - 2)(n - 3)}{2}
\end{aligned}$$

$$\begin{aligned}
&= 2(n-1) + \binom{n-3}{\sum_{i=1}^{n-3} i} + (n-2) \\
&= 2(n-1) + \binom{n-2}{\sum_{i=1}^{n-2} i} \\
&= 2(n-1) + \frac{(n-1)(n-2)}{2}.
\end{aligned}$$

Thus (2.8) holds for all values of n and this case is proven by induction.

Now suppose that for some $H_1, H_2 \in \mathcal{D}(1)$, we have $D = [H_1, H_2]_k$. Let $n_1 = |V(H_1)|$ and $n_2 = |V(H_2)|$. By Definition 2.2.4, we have $|A(D)| = |A(H_1)| + |A(H_2)| + 1 + n_1 n_2$. By the inductive hypothesis,

$$\begin{aligned}
|A(D)| &= \sum_{i=1}^2 \left[2(n_i - 1) + \frac{(n_i - 1)(n_i - 1)}{2} \right] + 1 + n_1 n_2 \\
&= 2(n_1 + n_2 - 1) - 1 + \frac{(n-1)(n-2)}{2} + \\
&\quad \frac{2n_1 n_2 + (n_1 - 1)(n_1 - 2) + (n_2 - 1)(n_2 - 2) - (n-1)(n-2)}{2} \\
&= 2(n_1 + n_2 - 1) - 1 + \frac{(n-1)(n-2)}{2} + \frac{1+1}{2} \\
&= 2(n_1 + n_2 - 1) + \frac{(n-1)(n-2)}{2}.
\end{aligned}$$

Therefore, (2.8) also holds for all values of n and Claim 4 is proved by induction.

By Lemma 2.3.4, $\mathcal{M}(k) \subseteq \mathcal{D}(k)$. By Definition 2.3.1, we have, for any $D \in \mathcal{M}(k)$,

$$|A(D)| = k(2n - k - 1) + \binom{n-k}{2}.$$

This, together with Claims 1, 2, 3 and 4, imply Theorem 2.4.1. \square

Chapter 3

Minimax Properties

3.1 Introduction

It is well known that graph theory plays a central role as a model for reliability, invulnerability, and survivability studies of large-scale networks [74]. By modeling networks as graphs, we are able to apply techniques and tools from graph theory to solve related problems. Oftentimes this will allow the problem to be solved more easily. Matula [87] indicated that many problems in graph theory can be further simplified by working on the components and blocks (defined as a maximal connected subgraph with no vertex cut) individually. Using this idea, a local connectivity measure is needed.

For a graph G , let $f(G)$ denote the edge-connectivity $\kappa'(G)$ or the minimum degree $\delta(G)$ of G , and define $\bar{f}(G) = \max\{f(H) : H \text{ is a subgraph of } G\}$. As indicated in [74], networks modeled as a graph G with $f(G) = \bar{f}(G)$, generally known as f -uniformly dense graphs, are of particular interest to investigate. Matula first studied the quantities

$$\bar{\kappa}'(G) = \max\{\kappa'(H) : H \subseteq G\} \text{ and } \bar{\delta}(G) = \max\{\delta(H) : H \subseteq G\}.$$

In order to compute $\bar{\kappa}'(G)$ and $\bar{\delta}(G)$, Matula defined slicings.

For a proper non-empty vertex subset X of G , let $C = (X, V(G) - X)$ denote the set

of edges in G leaving from X . Recall that we defined a slicing as a sequence of disjoint edge subsets in the form $C_i = (X, V(G_i) - X)$ for subgraph $G_i = G - \bigcup_{j=1}^{i-1} E(C_j)$ for $2 \leq i \leq m$ where $E(G) - \bigcup_{j=1}^m E(C_j) = \emptyset$. Further recall that a δ -slicing is defined to be a sequence of disjoint non-empty edge subsets of the form $C_i = (\{v_i\}, V(G_i) - \{v_i\})$ for a subgraph $G_i = G - \bigcup_{j=1}^{i-1} E(C_j)$ for $2 \leq i \leq m$ where $E(G) - \bigcup_{j=1}^m E(C_j) = \emptyset$. We say that the width of a slicing, denoted $w(S)$, where $Z = (C_1, C_2, \dots, C_m)$, is the maximum size of any cut in the slicing Z . The concept of slicing will be formally defined in the next section.

In [89], Matula discovered some minimax results involving $\bar{\kappa}'(G)$ and $\bar{\delta}(G)$.

Theorem 3.1.1 (Matula) [89]) *For any graph G with $|E(G)| \geq 1$, each of the following holds:*

- (i) $\bar{\kappa}'(G) = \max\{\kappa'(H) : H \subseteq G\} = \min\{w(Z) : Z \text{ is a slicing of } G\}$.
- (ii) $\bar{\delta}(G) = \max\{\delta(H) : H \subseteq G\} = \min\{w(Z) : Z \text{ is a } \delta\text{-slicing of } G\}$.

The purpose of this chapter is to investigate whether digraphs will have similar behavior. As in previously defined, the **minimum out-degree** and the **minimum in-degree** of a digraph D are

$$\begin{aligned} \delta^+(D) &= \min\{d_D^+(v) : v \in V(D)\} \text{ and} \\ \delta^-(D) &= \min\{d_D^-(v) : v \in V(D)\}, \text{ respectively.} \end{aligned}$$

Naturally, for a digraph D , we define

$$\begin{aligned} \bar{\lambda}(D) &= \max\{\lambda(H) : H \subseteq D\}, \\ \bar{\delta}^+(D) &= \max\{\delta^+(H) : H \subseteq D\} \text{ and} \\ \bar{\delta}^-(D) &= \max\{\delta^-(H) : H \subseteq D\}. \end{aligned}$$

Some of the recent studies on $\lambda(D)$ and $\bar{\lambda}(D)$ focused on extremal properties and the relationship with arc disjoint spanning arborescences, as seen in [5, 81, 83], among others.

By the definition of $\bar{\lambda}(D)$, we observe that $\bar{\lambda}(D) = 0$ if and only if D does not contain a directed cycle. That is, D is acyclic. Therefore, throughout this chapter, when discussing $\bar{\lambda}(D)$, we always assume that $\lambda(D) > 0$. A natural model for digraph slicing will be a sequence of disjoint non-empty arc subsets in the form $\partial_{D_i}^+(X)$ for some subdigraph D_i of D . Similarly, δ^+ -slicings (δ^- -slicings, respectively) will be sequence of disjoint non-empty arc subsets in the form $\partial_{D_i}^+(v)$ ($\partial_{D_i}^-(v)$, respectively) for some subdigraph D_i of D . The concept of slicing will be formally defined later in the chapter. Our main result is stated below.

Theorem 3.1.2 (*Anderson, Lai, & Xu*) *Let D be a digraph with $A(D) \neq \emptyset$. Let $\mathcal{S}(D)$ be the collection of all slicings of D and let $\mathcal{S}^+(D)$, $\mathcal{S}^-(D)$ be the collection of all δ^+ -slicings of D and all δ^- -slicings of D , respectively. Each of the following holds:*

(i) *Assume that $\bar{\lambda}(D) > 0$. Then*

$$\max\{\min|\partial_H^+(X)| : \emptyset \neq X \subset V(H)\} : H \subseteq D\} = \min\{\max\{|C_i| : 1 \leq i \leq m-1\} : S = (C_1, C_2, \dots, C_m) \in \mathcal{S}(D)\}.$$

$$(ii) \max\{\min\{d_H^+(v) : v \in V(H)\} : H \subseteq D\} = \min\{\max\{|C_i| : 1 \leq i \leq m\} : S = (C_1, C_2, \dots, C_m) \in \mathcal{S}^+(D)\}.$$

$$(iii) \max\{\min\{d_H^-(v) : v \in V(H)\} : H \subseteq D\} = \min\{\max\{|C_i| : 1 \leq i \leq m\} : S = (C_1, C_2, \dots, C_m) \in \mathcal{S}^-(D)\}.$$

Likewise, we define $\bar{\kappa}(D) = \max\{\kappa(H) : H \subseteq D\}$. Related properties on $\bar{\kappa}(D)$ are also discussed. In the following subsections we will present previous work on subgraph density measures, and graph connectivity algorithms. In the next section, we present the proofs for these minimax relations stated in Theorem 3.1.2, as well as discussions of other related properties. In the last section, we explain that these quantities $\bar{\delta}^+(D)$, $\bar{\delta}^-(D)$, $\bar{\lambda}(D)$ and $\bar{\kappa}(D)$ can be computationally determined in polynomial time.

3.1.1 Previous Work on Subgraph Density

Matula was the first to study the strength of a graph G . To study this invariant, he built up terminology based on cuts. An edge cut set, denoted $C = (X, \overline{X})$, of G is an edge set of G such that an edge e is in C if and only if $e = xy$ for some $x \in X$ and $y \in \overline{X}$. When no confusion arises, we shorten edge cut set to simply a cut. Using this terminology, for $k \geq 1$, we can define a graph G to be k -edge connected, if G is connected, $|V(G)| \geq 2$, and every cut has at least k -edges. Matula [87] defined a k -component (for any $k \geq 1$) of G to be a maximal k -edge connected subgraph of G .

The edge-connectivity of a graph can also be expressed in terms of cuts. For any graph G such that $|V(G)| \geq 2$, then the edge-connectivity of G is equal to the minimum number of edges in any cut of G . Formally, $\kappa'(G) = \min\{|C| : C \text{ is a cut of } G\}$. If G is not connected (i.e. does not contain any cut), then $\kappa'(G) = 0$. We say any cut that has $\kappa'(G)$ edges is a minimum cut. Notice, however, that a minimum cut of G does not have to be unique.

Lemma 3.1.3 (Matula [87]) *Let G_1, G_2, \dots, G_n be subgraphs of a graph G such that $\bigcup_i G_i$ is connected. Then $\kappa'(\bigcup_i G_i) \geq \min_i \kappa'(G_i)$.*

Corollary 3.1.4 *For any $k \geq 1$, the k -components of a graph G are vertex disjoint.*

Definition 3.1.5 *Let G be a graph with $E(G) \neq \emptyset$. A sequence of disjoint non-empty edge subsets $Z = (C_1, C_2, \dots, C_m)$ is a slicing of G if*

- (i) C_1 is a cut of G
- (ii) For each i with $2 \leq i \leq m$, C_i is a cut of $G - \bigcup_{j=1}^{i-1} C_j$

If $Z = (C_1, C_2, \dots, C_m)$ is a slicing of G , then the **width** of Z , is

$$w(Z) = \max\{|C_i| : 1 \leq i \leq m\}.$$

We say that Z is a **narrow slicing** if C_1 is a minimum cut of G and each C_i for $2 \leq i \leq m$ is a minimum cut of $G - \bigcup_{j=1}^{i-1} C_j$.

Theorem 3.1.6 (Matula [89]) *For any graph G with $|E(G)| \geq 1$, $\max\{w(Z) : Z \text{ is a slicing of } G\} = \max\{|C| : C \text{ is a cut of } G\}$.*

Theorem 3.1.7 (Matula [87]) *For any graph G containing at least one edge, the minimum width of any slicing of G is equal to the strength of the graph. Formally, $\bar{\kappa}(G) = \max\{\kappa'(H) : H \subseteq G\} = \min\{w(Z) : Z \text{ is a slicing of } G\}$*

Definition 3.1.8 *Let G be a graph with $E(G) \neq \emptyset$. A sequence of disjoint non-empty edge subset $Z = (C_1, C_2, \dots, C_m)$ is a **δ -slicing** of G if $C_1 = (\{x\}, V(G) - \{x\})$ for some $x \in V(G)$ is an edge cut of G , and for each i with $2 \leq i \leq m$, $C_i = (\{x\}, V(G) - \{x\})$ for some $x \in V(G)$ is an edge cut of $G - \bigcup_{j=1}^{i-1} E(C_j)$.*

Theorem 3.1.9 (Matula [89]) *For any graph G with $|E(G)| \geq 1$, the maximum minimum degree of any subgraph of G (called the δ -strength of G) is equal to the minimum width of any δ -slicing. Formally,*

$$\bar{\delta}(G) = \max\{\delta(H) : H \subseteq G\} = \min\{w(Z) : Z \text{ is a } \delta\text{-slicing of } G\}.$$

In 1932, Whitney published a classical result that relates the connectivity, edge-connectivity, and the minimum degree of a graph.

Theorem 3.1.10 (Whitney [113]) *For any graph G ,*

$$\kappa(G) \leq \kappa'(G) \leq \delta(G).$$

Let $\bar{\kappa}(G) = \max\{\kappa(H) : H \subseteq G\}$, which we call the vertex-strength. Matula was able to show Whitney's classical result was analogous when looking at the strengths of graphs.

Theorem 3.1.11 (Matula [89]) *For a graph G ,*

$$\bar{\kappa}(G) \leq \bar{\kappa}'(G) \leq \bar{\delta}(G)$$

Finding all subgraphs of a graph can be very difficult and time consuming, especially as the size of the graph increases. Matula was able to eliminate most of the work needed to solve $\bar{\kappa}'(G)$ and $\bar{\delta}(G)$. His theorem focuses on only having to find one cut for the graph, instead of having to search for every subgraph to ensure you have found the maximum.

Theorem 3.1.12 (Matula [89]) *Let G be a graph with $E(G) \neq \emptyset$. Then*

(i) *For any δ -slicing $Z = (C_1, C_2, \dots, C_m)$ of G ,*

$$\bar{\delta}(G) = \max_{1 \leq i \leq m} \{|C_i|\}$$

(ii) *For any narrow slicing $Z' = (C'_1, C'_2, \dots, C'_m)$ of G ,*

$$\bar{\kappa}'(G) = \max_{1 \leq i \leq m} \{|C'_i|\}$$

Matula developed a few lemmas, which he called cut lemmas that are very useful in determining $\bar{\kappa}'(G)$ and $\bar{\delta}(G)$, which we will discuss further in the next subsection on connectivity algorithms.

Lemma 3.1.13 (Matula [89]) *For a graph G with $E(G) \neq \emptyset$. If v is a minimum degree vertex in G , then*

$$\bar{\delta}(G) = \max\{\delta(G), \bar{\delta}(G - \{v\})\}$$

Lemma 3.1.14 (Matula [89]) *For a graph G with $E(G) \neq \emptyset$. If $C \subseteq E(G)$ is a minimum cut for some component of G , then*

$$\bar{\kappa}'(G) = \max\{|C|, \bar{\kappa}'(G - C)\}$$

3.1.2 Previous Work on Graph Connectivity Algorithms

Abdol-Hossein Esfahani [40] previously wrote about the history of graph connectivity algorithms and we shall follow a similar approach.

T. E. Harris and F. S. Ross first introduced the maximum flow problem in 1955. They were working for the U.S. government to develop a method of evaluating railroad capacities. In [72], the problem presented states that given a railroad network with known (or approximated) capacities for each line and specific locations designated as the origins (source) and others listed as the destinations (sink), find a route in the network that enables the maximum amount to be moved from the origin to the destination periodically and also find the numerical value for the maximum. Ford and Fulkerson [45] created the first algorithm to compute the maximum flow based on a greedy algorithm (at each step a greedy algorithm will take the best local choice) in 1956.

Since then, there have been many algorithms created for maximum flow and these algorithms are helpful in solving edge-connectivity and vertex-connectivity of graphs. Algorithms are helpful in solving edge-connectivity and vertex-connectivity of graphs do so by calling maximum flow algorithms for some steps, and these calls carry the most computational intensities. Naturally, minimizing the number of calls to these maximum flow algorithm will optimize computing speeds in connectivity algorithms.

Let x and y be distinct vertices in a graph G . Let us define $\kappa'(x, y)$ to be the least amount of edges that must be removed in order that no (x, y) -paths exist. Recall that we have previously defined $\kappa'(G) = \min\{|C| : C \text{ is a cut of } G\}$. If $G \neq K_1$, then we define $\kappa'(G) = \min\{\kappa'(x, y) : x, y \in V(G) \text{ and } x \neq y\}$. If one was able to solve $\kappa'(x, y)$ for arbitrary distinct vertices, then $\kappa'(G)$ could be solved using our definition.

S. Even [43] developed an algorithm to solve $\kappa'(x, y)$ based on solving a maximum flow problem. The algorithm that follows is from a proof that Even's presented in [43] as a variation of Menger's Theorem. The approach for the proof was first suggested by Dantzig and Fulkerson [37].

Algorithm I (Even [43])

Input: A graph G and distinct vertices x and y ($|V(G)| = n > 0$ and $|E(G)| = m > 0$).

Output: $\kappa'(x, y)$

(Step 1) Replace every edge uv with arcs (u, v) and (v, u)

(Step 2) Assign x as the source vertex and y as the sink vertex.

(Step 3) Assign each arc to have capacity 1, and let H be the resulting network.

(Step 4) Find a maximum flow function f in H .

(Step 5) Set $\kappa'(x, y)$ equal to the total flow of f . **Stop**.

In [43], Even found the time complexity of this algorithm to be $O(mn)$. One could develop a program to run **Algorithm I** for every possible distinct pair of vertices and then take the minimum quantity of the results to compute $\kappa'(G)$. Since the ordering of edges does not matter in graphs, for a graph G such that $|V(G)| = n$, there are $\frac{n(n-1)}{2}$ distinct vertex pairs that must compute $\kappa'(x, y)$ to determine $\kappa'(G)$.

Let x and y be distinct vertices in a digraph D . Let us define $\lambda(x, y)$ to be the least amount of arcs that must be removed in order that no (x, y) -dipaths exist. It is important to note that $\lambda(x, y)$ does not always equal $\lambda(y, x)$. Assume that $D \neq K_1^*$ and $|A(D)| \geq 1$, we can define $\lambda(D) = \min\{\lambda(x, y) : x, y \in V(D) \text{ and } x \neq y\}$.

Even's **Algorithm I** computes $\lambda(x, y)$ for arbitrary distinct pairs by simply updating the input and output, and removing Step 1. Similarly one could run **Algorithm I** for every possible distinct pair of vertices and take the minimum of the results to compute $\lambda(D)$. Since the ordering of the vertices does matter for digraphs, then for a digraph D with $|V(D)| = n$, there are $n(n-1)$ distinct vertex pairs that must be computed. Since minimizing the number of calls to maximum flow algorithms optimizes computational speeds, the focus on further connectivity algorithms is reducing the number of distinct vertex pairs needed to be checked.

One method that can be employed in minimizing the number of distinct vertex pairs is to look at the structure of a cut, which we introduced in the previous section. Let G be a nontrivial connected graph. Let $C = (X, \overline{X})$ be a minimal edge cut set of G . By definition, $|C| = \kappa'(G)$. Let $x \in X$ and $y \in \overline{X}$. Since C is a minimal cut, then $\kappa'(x, y) = \kappa'(G)$. If this was not the case, then $\kappa'(x, y) > \kappa'(G)$ or $\kappa'(x, y) < \kappa'(G)$. If

$\kappa'(x, y) > \kappa'(G)$, then there exists an (x, y) -path in $G - C \rightarrowleftarrow$ as C is a cut of G . If $\kappa'(x, y) < \kappa'(G)$, then there exists a smaller cut of $G \rightarrowleftarrow$ as C is minimal. Let $u \in V(G)$. Then $u \in X$ or $u \in \overline{X}$. WLOG, assume $u \in X$. If we had a way to pick a $v \in \overline{X}$, then we know $\kappa'(u, v) = \kappa'(G)$.

While determining a vertex in \overline{X} given a vertex $u \in X$ is difficult for an arbitrary minimal cut, Even and Tarjan [44] noticed that by setting $C = (\{u\}, V(G) - \{u\})$, one could easily pick a vertex in $V(G) - \{u\}$. Unfortunately, this does not guarantee that C is a minimal cut. However, they were still able to develop a new algorithm using this idea that was an improvement over **Algorithm I**.

Algorithm II (Even and Tarjan [44])

Input: A connected nontrivial graph G with $|V(G)| = n$

Output: $\kappa'(G)$

(Step 1) Set $k = n - 1$. Select an arbitrary $x \in V(G)$ and let $X = V(G) - \{x\}$.

(Step 2) Enumerate the vertices in X $\{v_1, v_2, \dots, v_{n-1}\}$.

(Step 3) While $i < n$

Compute $\kappa'(x, v_i)$ using **Algorithm I**

(Step 3.1) **updating** set $k := \min\{k, \kappa'(x, v_i)\}$

In **Algorithm II**, notice we only need $n - 1$ calls to the maximum flow algorithm. For larger graphs, this is a considerable reduction from $\frac{n(n-1)}{2}$.

Recall Whitney [113] proved that $\kappa'(G) \leq \delta(G)$. Esfahanian and Hakimi [41] give a useful lemma for determining the size of the vertex partitions from a cut when $\kappa'(G) < \delta(G)$.

Lemma 3.1.15 (*Esfahanian and Hakimi [41]*) *Let G be a nontrivial connected graph. Let $C = (X, \overline{X})$ be a minimum cut of G . If $\kappa'(G) < \delta(G)$, then $|X| > \delta(G)$ and $|\overline{X}| > \delta(G)$.*

From this lemma, we have the following corollary:

Corollary 3.1.16 (*Esfahanian and Hakimi [41]*) *Let G be a nontrivial connected graph such that $\kappa'(G) < \delta(G)$. Let $C = (X, \overline{X})$ be a minimum cut of G . Then*

- (i) Both X and \overline{X} contain at least one vertex that is not incident to any edges in C .
- (ii) Both X and \overline{X} contain at least one vertex with degree greater than 1 (a non-leaf vertex) of every spanning tree in G .

Using this corollary, we have the following algorithm:

Algorithm III (Esfahanian and Hakimi [41])

Input: A connected nontrivial graph G with $|V(G)| = n$

Output: $\kappa'(G)$

(Step 1) Set $k = n - 1$. Select a spanning tree T of G , and let Y be the set of all non-leaves of T .

(Step 2) Select an arbitrary $x \in Y$ and let $X = Y - \{x\}$.

(Step 3) Enumerate the vertices in X $\{v_1, v_2, \dots, v_{n-1}\}$.

(Step 4) While $i < n$

Compute $\kappa'(x, v_i)$ using **Algorithm I**

(Step 4.1) **updating** set $k := \min\{k, \kappa'(x, v_i)\}$

(Step 5) Assign $\kappa'(G) := \min\{k, \delta(G)\}$. **Stop**.

If $|V(G)| = 2$, then as G is simple and connected, $C = K_2$, **Algorithm III** would only run through once before terminating. Assume that G contains a nontrivial spanning tree T . Claim: T contains at least two leaves. Assume T contains at most one leaf. If T has no leaves, then $d_G(v) \geq 2$ for every $v \in V(G)$. Then T contains a cycle. $\rightarrow\leftarrow$ as T is acyclic. Assume that T contains one leaf. Let P be a path of maximal length and enumerate the vertices in P to be v_0, v_1, \dots, v_m . (If P did not contain the leaf, then P would be a cycle as P is maximal in length and $d_G(v_i) \geq 2$). WLOG assume, $d_G(v_0) = 1$. Consider v_m , which $d_G(v_m) \geq 2$. Let $v \in V(G)$ be a neighbor of v_m other than v_{m-1} . Then $v \in P$ or $v \notin P$. If $v \notin P$, then we can extend P to a longer path. $\rightarrow\leftarrow$ as P is maximal. If $v \in P$, then P contains a cycle. $\rightarrow\leftarrow$ as T is acyclic. Thus every nontrivial tree contains at least two leaves.

Since a tree has at least two leaves, then in Step 2 of **Algorithm III** we have $|X| \leq |V(G)| - 2$, which guarantees that we will have fewer calls to a maximum flow algorithm. However, Gary and Johnson [56] has shown that finding a spanning tree that has a

maximal amount of leaves is an NP-hard problem. Esfalian and Hakimi [41] created an algorithm that builds a spanning tree that will contain a large number of leaves. They modified **Algorithm III** by replacing Step 1 (selecting any spanning tree) with the algorithm they created. The modified algorithm reduces the size of X to less than or equal to $\frac{n}{2}$, which will only make at most $\frac{n}{2}$ calls to a maximum flow algorithm.

Let G be a graph and $D \subseteq V(G)$. D is said to be a dominating set if for every $v \in V(G)$, $v \in D$ or $v \in N_G(u)$ for some $u \in D$. Let $C = (A, \bar{A})$ be a minimum cut. Let $Y \subseteq V(G)$. If $Y \cap A \neq \emptyset$ and $Y \cap \bar{A} \neq \emptyset$, then Y is called a κ' -covering. Using dominating sets, Matula [88] was able to improve on the algorithm just discussed by utilizing the following lemma:

Lemma 3.1.17 (Matula [88]) *Let G be a graph having $\kappa'(G) < \delta(G)$. Let $C = (X, \bar{X})$ be a minimum cut. If D is a dominating set, then D contains vertices of A and \bar{A} .*

Notice that Lemma 3.1.7 states that if D is a dominating set, then D is a κ' -covering. The lemma can be proved by using Lemma 3.1.15. Since C is a cut, X and \bar{X} partition the vertices of G . Let $D \subseteq V(G)$ be a dominating set. Then for every $x \in D$, $x \in X$ or $x \in \bar{X}$. For D to be a κ' -covering we need vertices in both X and \bar{X} . Assume by contradiction that $D \subseteq X$ or $D \subseteq \bar{X}$. Without loss of generality, assume that $D \subseteq V(X)$. Since D is dominating, for every $y \in \bar{X}$, y is incident to a vertex in D . From Lemma 3.1.15, we know $|\bar{X}| > \delta(G)$. Since $D \subseteq V(X)$, every y is incident to a vertex in D , and $|\bar{X}| > \delta(G)$, then there are at least $\delta(G)$ edges in C . Since C is a minimum cut, $\kappa'(G) = |C| \geq \delta(G)$. $\rightarrow \leftarrow$ as $\kappa'(G) < \delta(G)$.

Algorithm IV (Matula [88])

Input: A connected nontrivial graph G

Output: $\kappa'(G)$

(Step 1) Select a dominating set D of G .

(Step 2) Select an arbitrary vertex $x \in D$ and let $X = D - \{x\}$.

(Step 3) Use **Algorithm I** to compute $\kappa'(x, y)$ for every $y \in X$.

(Step 4) Assign $k := \min\{\kappa'(x, y) : y \in X\}$.

(Step 5) Assign $\kappa'(G) := \min\{k, \delta(G)\}$. **Stop**.

The smaller the dominating set, the fewer calls to the maximum flow function. Unfortunately, finding a minimal dominating set is NP-hard [56]. Matula was able to develop an algorithm that produces small dominating sets though they may not be minimal.

Algorithm V (Matula [88])

Input: A connected nontrivial graph G

Output: A dominating set D of G .

(Step 1) Select a vertex $x \in V(D)$ and let $D = \{x\}$

(Step 2) While $V(G) - (D \cup N_G(D)) \neq \emptyset$

Select a vertex $y \in V(G) - (D \cup N_G(D))$

(Step 2.1) **updating** set $D := D \cup \{y\}$

Using **Algorithm V** in place of Step 1 in **Algorithm IV**, and a few other enhancements, Matula [88] was able to create the fastest algorithm finding $\kappa'(G)$ presented; it runs in time complexity $O(mn)$.

While **Algorithm I** works for digraphs, **Algorithm II** cannot be used for digraphs. This is because $\lambda(u, v) \neq \lambda(v, u)$ for all $u, v \in V(D)$ except when $D = K_n^*$ for some positive integer n . When we take a minimal cut $C = (A, \bar{A})$, the cut is only minimal for those vertices in A . If $x \notin \bar{A}$, $\lambda(x, w) \geq |C|$ for every $w \in V(D) - \{x\}$, and equality may not exist.

Previously we defined κ' -covering of G to be any vertex set $K \subseteq V(G)$ such that K contains a vertex from both vertex partitions of a minimum cut. We can similarly define λ -coverings for digraphs. Let $C = (X, \bar{X})$ be a minimum cut of D . Let $K \subset V(D)$. K is a λ -covering of D if K contains at least one vertex from X and one vertex from \bar{X} . Schnorr [101] created a new algorithm based on a λ -covering.

Lemma 3.1.18 *Let $C = (X, \bar{X})$ be a minimum cut of a digraph D . Let $K \subseteq V(D)$ be a λ -covering of D and enumerate $K = \{v_1, v_2, \dots, v_m\}$. Then*

$$\lambda(D) = \min\{\lambda(v_1, v_2), \lambda(v_2, v_3), \dots, \lambda(v_{m-1}, v_m), \lambda(v_m, v_1)\}.$$

Upon first glance, the validity for the lemma may not appear clear. Let us look at why this lemma is indeed true. We know that K is a λ -covering, so there exists an $x \in K$

such that $x \in X$ and a $y \in K$ such that $y \in \overline{X}$. Then $v_1 \in X$ or $v_1 \in \overline{X}$. WLOG, assume $v_1 \in \overline{X}$. For each v_i , $v_i \in X$ or $v_i \in \overline{X}$. Since $v_1 \in \overline{X}$, then there exists a vertex v_j such that $v_j \in X$. In the worst case scenario, all $v_i \in X$ for $j \leq i \leq m$. However, since $\lambda(v_m, v_1)$ is also included, and $v_m \in X$ and $v_1 \in \overline{X}$, and C is a minimum cut, then $\lambda(v_m, v_1) = \lambda(D)$. While this was the worst case scenario, it shows that at some point, we must have the sequence $v_i \in X$, $v_{i+1} \in \overline{X}$, which gives guarantees that some $\lambda(v_i, v_{i+1}) = \lambda(D)$.

Schnorr [101] created an algorithm from the idea of the previous lemma. By letting $K = V(D)$, then we can see that for any cut C , K will contain at least one vertex from each of the vertex partition sets. Hence, for any minimal cut, K is a λ -covering of D . By using this, we have eliminated some of the ordered pairs that needed to be checked in **Algorithm I**.

Algorithm VI (Schnorr [101])

Input: A strong digraph D

Output: $\lambda(D)$

(Step 1) Let $V = V(D) = \{v_1, v_2, \dots, v_n\}$

(Step 2) Using **Algorithm I**, compute $\lambda(v_1, v_2), \lambda(v_2, v_3), \dots, \lambda(v_{m-1}, v_m), \lambda(v_m, v_1)$

(Step 3) **updating** set $\lambda(D) := \min\{\lambda(v_1, v_2), \lambda(v_2, v_3), \dots, \lambda(v_{m-1}, v_m), \lambda(v_m, v_1)\}$

There have been further improvements and many are comparable to translating the previous results from graphs into digraphs. Mansour and Schieber's [86] algorithms have reduced the time complexity to $O(\min(mn, n\lambda^2))$ for a digraph with n vertices and m arcs. We will finish out this section by looking at some advancements made at computing the vertex connectivity of a graph.

We have defined vertex connectivity to be the minimum number of vertices that must be removed in order to disconnect G . Let $x, y \in V(G)$. Let $\kappa(x, y) = n - 1$ if $xy \in E(G)$, and for $xy \notin E(G)$, $\kappa(x, y)$ is the minimum number of vertices that must be removed so that no (x, y) -path exists. Using this, we can create an equivalent definition for $\kappa(G)$: $\kappa(G) = \min\{\kappa(x, y) : x \text{ and } y \text{ are distinct vertices in } G\}$.

Even [43] was able to compute $\kappa(x, y)$ for distinct $x, y \in V(G)$ from a maximum flow

problem as edge-connectivity has been solved using maximum flow. The algorithm runs in time complexity $O(mn^{2/3})$.

Algorithm VII (Even [43])

Input: A graph G with x, y distinct vertices

Output: $\kappa(G)$

(Step 1) Replace each edge $uv \in E(G)$ with arcs (u, v) and (v, u) to turn G into a digraph

(Step 2) For $v \in V(G)$ and $v \notin \{x, y\}$, replace v with two new vertices v_1 and v_2 . Add (v_1, v_2) . For every arc of the form (w, v) replace those arcs with (w, v_1) and for every arc of the form (v, w) replace those arcs with (v_2, w) .

(Step 3) Assign x as the source vertex and y as the sink vertex.

(Step 4) Assign the capacity of each arc to 1, and let H be the resulting network.

(Step 5) Find a maximum flow function f in H .

(Step 6) Set $\kappa(x, y)$ equal to the total flow of f . **Stop**.

To improve on the time complexity of $\kappa(G)$, we will look at vertex cuts, much the same as we used edge cuts to make time complexity improvements on $\kappa'(G)$. Consider a minimum vertex cut set S . Unlike edge cut sets which only partitions $V(G)$ into two sets, a vertex cut set may partition $V(G)$ into more than two sets. Let X be one component of $G - S$, and let $\bar{X} = V(G) - (X \cup S)$, which could be the empty set. If this is the case, then we ignore \bar{X} . If $x \in X$ and $y \in \bar{X}$, then $\kappa(x, y) = \kappa(G)$ since S is a minimum vertex cut set and every path from x to y must go through the vertices of S .

Notice that we cannot simply select an arbitrary vertex x and compute $\kappa(x, y)$ such that $y \in V(G)$ and y is not adjacent to x . This is because for some graphs if the arbitrary vertex x is selected in S , $\kappa(x, y) > \kappa(G)$. We know from Whitney's Theorem [113] that $\kappa(G) \leq \delta(G)$. By taking a vertex set $Y \subset V(G)$ such that $|Y| > \delta(G)$, then no matter what minimum cut S in G that is chosen, there will be a vertex $x \in Y$ such that x is not in S . Thus we can compute $\kappa(x, y)$ for each $x \in Y$ and $y \in V(G) - \{x\}$ such that y is not adjacent to x in G . Then we set $\kappa(G)$ to be the minimum of all such $\kappa(x, y)$.

Algorithm VIII (Even and Tarjan [44])

Input: A graph G

Output: $\kappa(G)$

(Step 1) Assign $i = 1$, $N := n - 1$, and let $V = \{v_1, v_2, \dots, v_n\}$.

(Step 2) For each j , $j = i + 1, i + 2, \dots, n$

(Step 2.1) If $i > N$ go to (Step 4)

(Step 2.2) If v_i and v_j are not adjacent in G , compute $\kappa(v_i, v_j)$ using **Algorithm VIII** and assign $N := \min\{N, \kappa(v_i, v_j)\}$.

(Step 3) **updating** set $i := i + 1$

(Step 4) Set $\kappa(G) = N$. **Stop**.

Even and Tarjan [44] found that their algorithm makes $O((n - \delta(G) - 1)\kappa(G))$ calls to a maximum flow problem. Esfahanian and Hakimi [41] were able to make improvements and lower the amount of calls needed. They did this by observing for an arbitrary $x \in V(G)$, either $x \notin S$ for some minimum vertex cut and we use **Algorithm VIII** or $x \in S$ and so at least two neighbors of x are outside of S , and $\kappa(G) = \min\{\kappa(u, v) : x, y \in N_G(X) \text{ and } u, v \text{ are not neighbors}\}$.

Algorithm IX (Esfahanian and Hakimi [41])

Input: A graph G

Output: $\kappa(G)$

(Step 1) Select a vertex v such that $d_G(v) = \delta(G)$.

(Step 2) Compute $k_1 = \min\{\kappa(v, w) : w \in V(G) - \{v\}, w \notin N_G(v)\}$

(Step 3) Compute $k_2 = \min\{\kappa(x, y) : w \in N_G(v), x \notin N_G(y)\}$

(Step 4) Assign $\kappa(G) := \min\{k_1, k_2\}$. **Stop**.

Esfahanian and Hakimi [41] found $O(n - \delta(G) + 1 + \delta(G)(\delta(G) - 1)/2)$ calls are needed to the maximum flow problem. Other refinements can be made and seen in [41]. These algorithms can be made applicable to directed graphs.

3.2 Minimax Theorems in Subdigraph Density Measures

Let $k \geq 0$ be an integer. A digraph D is **k -arc-strong** if $\lambda(D) \geq k$, or equivalently, for any proper non-empty subset $\emptyset \neq X \subset V(D)$, we always have $|\partial_D^+(X)| \geq k$. Thus in this sense, every digraph D is 0-arc-strong, and $\lambda(D) = 0$ if and only if D is not 1-arc-strong. Let D be a digraph and let D_1 and D_2 be two subdigraphs of D . Define $D_1 \cup D_2$ to be the subdigraph of D with $V(D_1 \cup D_2) = V(D_1) \cup V(D_2)$ and $A(D_1 \cup D_2) = A(D_1) \cup A(D_2)$. We start with some elementary properties.

Lemma 3.2.1 (*Anderson et al.*) *Let D_1, D_2, \dots, D_n be subdigraphs of a digraph D such that $\bigcup_i D_i$ is strongly connected. Then $\lambda(\bigcup_i D_i) \geq \min_i \lambda(D_i)$.*

Proof. Let $H = \bigcup_{i=1}^n D_i$, and $k = \lambda(H)$. Then $k \geq 1$. Hence, without loss of generality, we may assume that $\lambda(D_n) \geq \dots \geq \lambda(D_2) \geq \lambda(D_1) > 0$. Since $\lambda(H) = k$, there exists a non-empty proper edge subset X with $\emptyset \neq X \subset V(H)$ such that $|\partial_H^+(X)| = k$. If there exists an i with $1 \leq i \leq n$ such that both $X \cap V(D_i) \neq \emptyset$ and $V(D_i) - X \neq \emptyset$, then we have that $|\partial_H^+(X)| \geq |\partial_{D_i}^+(X \cap V(D_i))| \geq \lambda(D_i) \geq \lambda(D_1)$, and so the lemma holds.

Hence we may assume that

$$\text{for every } i \text{ if } X \cap V(D_i) \neq \emptyset, \text{ then } V(D_i) \subseteq X. \quad (3.1)$$

Let $U = \bigcup \{V(D_i) : X \cap V(D_i) \neq \emptyset\}$. By (3.1), $U \subseteq X$. For any vertex $x \in X$, as $x \in V(H) = \bigcup_{i=1}^n V(D_i)$, we have $x \in V(D_\ell)$ for some ℓ with $1 \leq \ell \leq n$. Thus $X \cap V(D_\ell) \neq \emptyset$, and so by (3.1), $V(D_\ell) \subseteq U$, implying $x \in U$. It follows that $X \subseteq U$, and so we have that $X = U$. Since H is strongly connected, there exists an $(u, v) \in (X, V(H) - X)_H$ with $u \in X$ and $v \in V(H) - X$. By the definition of H , $(u, v) \in A(D_i)$ for some i with $1 \leq i \leq n$. But then, as $u \in X$, we have $V(D_i) \cap X \neq \emptyset$ and so (3.1) implies that $V(D_i) \subseteq X$, contrary to the fact that $v \in V(H) - X$. This contradiction indicates that (3.1) does not hold, and so the lemma follows. \square

Theorem 3.2.2 (Geller and Harary [57]) *Let D be a strong digraph. Then*

$$\kappa(D) \leq \lambda(D) \leq \min\{\delta^+(D), \delta^-(D)\}.$$

Lemma 3.2.3 (Anderson et al.) *Let D be a strong digraph. Then $\bar{\kappa}(D) \leq \bar{\lambda}(D) \leq \min\{\bar{\delta}^+(D), \bar{\delta}^-(D)\}$.*

Proof. Let $L \subseteq D$ such that $\bar{\kappa}(D) = \kappa(L)$. By (3.2.2), $\bar{\kappa}(D) = \kappa(L) \leq \lambda(L) \leq \bar{\lambda}(D)$. To show that $\bar{\lambda}(D) \leq \min\{\bar{\delta}^+(D), \bar{\delta}^-(D)\}$, we take a subdigraph $H \subseteq D$ such that $\bar{\lambda}(D) = \lambda(H)$. Let $v \in H$ such that $d_H^+(v) = \delta^+(H)$. As $H - \partial_H^+(v)$ is not strong, we have $\bar{\lambda}(D) = \lambda(H) \leq \delta^+(H) \leq \bar{\delta}^+(D)$. Similarly, we also have $\bar{\lambda}(D) \leq \bar{\delta}^-(D)$. \square

3.2.1 Slicing and Proof of Theorem 3.1.2(i)

Throughout this subsection, we assume that D is a digraph with $\bar{\lambda}(D) > 0$. An arc subset W of D is a **direct cut** of D if there exists a non-empty proper vertex subset X such that $W = (X, V(D) - X)_D$ with $W \neq \emptyset$. We present a formal definition of digraph slicing below.

Definition 3.2.4 *Let D be a digraph with $\lambda(D) > 0$. Set $D_1 = D$.*

(i) *A **slicing** of D is a sequence $S = (C_1, C_2, \dots, C_s)$ of arc subsets of D with $s \geq 2$ such that each of the following holds:*

(i-1) C_1 is a direct cut of D_1 .

(i-2) Define $D_2 = D - C_1$. For $2 \leq i \leq s - 1$, D_i is not acyclic, C_i is a non-empty direct cut of D_i and set $D_{i+1} = D_i - C_i$.

(i-3) $D_s = D - \bigcup_{i=1}^{s-1} C_i$ is acyclic.

(ii) *If, for each i with $1 \leq i \leq s - 1$, C_i is a minimum direct cut of a nontrivial strong component of D_i , then the slicing $S = (C_1, C_2, \dots, C_s)$ is a **narrow slicing**.*

(iii) *The **width** of a slicing $S = (C_1, C_2, \dots, C_s)$ is $w(S) = \max\{|C_i|, 1 \leq i \leq s - 1\}$.*

(iv) *The collection of all slicings of D is denoted by $\mathcal{S}(D)$.*

Proof of Theorem 3.1.2(i). Let $k = \bar{\lambda}(D)$ and $k' = \min\{w(S) : S \in \mathcal{S}(D)\}$. By the assumption of Theorem 3.1.2(i), we have $k > 0$.

Suppose first that H is a subdigraph of D with $k = \lambda(H)$. Let $S = (C_1, C_2, \dots, C_m)$ be a slicing of D . By the definition of a slicing, $D_m = D - \bigcup_{i=1}^{m-1} C_i$ is acyclic. Since $\lambda(H) = k \geq 1$, H is not a subdigraph of D_m . Hence there must be a smallest index ℓ with $1 \leq \ell < m$ such that the arc subset $C_\ell \cap A(H) \neq \emptyset$. It follows that $C_\ell \cap A(H)$ is a direct cut of H , so $w(S) \geq |C_\ell| \geq |C_\ell \cap A(H)| \geq \lambda(H) = k$. Since S was arbitrary, we have $k' = \min\{w(S) : S \in \mathcal{S}(D)\} \geq k = \bar{\lambda}(D)$.

Conversely, let $k'' = \min\{w(S) : S \text{ is a narrow slicing of } D\}$. Then as the collection of all narrow slicings of D is a subset of $\mathcal{S}(D)$, it follows by definition that $k'' \geq k'$. We will show that $k \geq k''$, which implies the desired $k \geq k'$. Arguing by contradiction, we assume that $k'' > k$. Then there exists a narrow slicing $S = (C_1, C_2, \dots, C_m)$ such that $k'' = w(S) \geq k + 1$. Hence there exists a smallest i with $1 \leq i \leq m - 1$ such that $|C_i| = w(S) \geq k + 1$. Since S is a narrow slicing, by Definition 3.2.4(ii), C_i is a minimum direct cut of a strong component L of D_i . It follows that $\lambda(D_i) = |C_i| = w(S) \geq k + 1 > \bar{\lambda}(D) \geq \lambda(D_i)$. This contradiction implies that we must have $k \geq k'' \geq k'$. This establishes Theorem 3.1.2(i). \square

The argument deployed in the proof of Theorem 3.1.2(i) suggests some computational useful ways of determining $\bar{\lambda}(D)$, as stated in the following results.

Lemma 3.2.5 (*Anderson et al.*) *Let D be a digraph with $A(D) \neq \emptyset$, and let $\mathcal{C}(D) = \{C \subset A(D) : C \text{ is a minimal direct cut of a strong component of } D\}$. Then*

$$\bar{\lambda}(D) = \max_{C \in \mathcal{C}(D)} \{|C|, \bar{\lambda}(D - C)\}.$$

Proof. Let $H_1 \subset D$ be a digraph such that $\bar{\lambda}(D) = \lambda(H_1)$, and let $C_1 \in A(H_1)$ be a direct cut such that $|C_1| = \lambda(H_1)$. Then $C_1 \in \mathcal{C}(D)$, and hence $\bar{\lambda}(D) = \lambda(H_1) = |C_1| \leq \max_{C \in \mathcal{C}(D)} \{|C|, \bar{\lambda}(D - C)\}$.

Conversely, let $k = \max_{C \in \mathcal{C}(D)} \{|C|, \bar{\lambda}(D - C)\}$. First assume $k = |C|$ for some $C \in \mathcal{C}(D)$. Let $C_2 \in \mathcal{C}(D)$ be such that $|C_2| = k$. Since $C_2 \in \mathcal{C}(D)$, D has a strong

component H' such that C_2 is a minimal direct cut of H' . Then $|C_2| = \lambda(H') \leq \bar{\lambda}(D)$. Then $\bar{\lambda}(D) \geq \max_{C \in \mathcal{C}(D)} \{|C|, \bar{\lambda}(D - C)\}$.

Now we assume $k = \bar{\lambda}(D - C)$ for some $C \in \mathcal{C}(D)$. Let $C_2 \in \mathcal{C}(D)$ be such that $k = \bar{\lambda}(D - C_2)$. Then $\bar{\lambda}(D) \geq \bar{\lambda}(D - C_2) = k$. Thus $\bar{\lambda}(D) \geq \max_{C \in \mathcal{C}(D)} \{|C|, \bar{\lambda}(D - C)\}$.

Since $\bar{\lambda}(D) \geq \max_{C \in \mathcal{C}(D)} \{|C|, \bar{\lambda}(D - C)\}$ when $k = |C|$ for some $C \in \mathcal{C}(D)$ and when $k = \bar{\lambda}(D - C)$ for some $C \in \mathcal{C}(D)$, Then $\bar{\lambda}(D) \geq \max_{C \in \mathcal{C}(D)} \{|C|, \bar{\lambda}(D - C)\}$ holds in general.

Thus $\bar{\lambda}(D) = \max_{C \in \mathcal{C}(D)} \{|C|, \bar{\lambda}(D - C)\}$. \square

Lemma 3.2.6 (*Anderson et al.*) *Let D be a digraph with $A(D) \neq \emptyset$. If $S = (C_1, C_2, \dots, C_s)$ is a narrow slicing of D , then $\bar{\lambda}(D) = \max_{1 \leq i \leq s-1} \{|C_i|\}$.*

Proof. Let $S = (C_1, C_2, \dots, C_s)$ be a narrow slicing of D . Since a narrow slicing is a slicing, it follows by Theorem 3.1.2(i) that $\bar{\lambda}(D) \leq \max_{1 \leq i \leq s-1} \{|C_i|\}$. Conversely, let ℓ be an integer with $1 \leq \ell \leq s - 1$ satisfying $|C_\ell| = \max_{1 \leq i \leq s-1} \{|C_i|\}$. By the definition of a narrow slicing, there exists a subdigraph D_ℓ of D such that C_ℓ is a direct cut of D_ℓ with $|C_\ell| = \lambda(D_\ell)$. It follows that $\bar{\lambda}(D) = \max\{\min |\partial_H^+(X)| : \emptyset \neq X \subset V(H) : H \subseteq D\} \geq \lambda(D_\ell) = |C_\ell| = \max_{1 \leq i \leq s-1} \{|C_i|\}$. Thus $\bar{\lambda}(D) = \max_{1 \leq i \leq s-1} \{|C_i|\}$. \square

3.2.2 The Study of δ^+ -slicing, δ^- -slicing

Throughout this subsection, we assume that D is a digraph with $A(D) \neq \emptyset$. For a digraph D , let $G(D)$, called the **underlying graph** of D , be the graph obtained from D by erasing all the orientation of the arcs of D and removing any duplicate edges created. A digraph D is **weakly connected** if $G(D)$ is connected. A subdigraph H of D is a **weak component** of D if $G(H)$ is a component of $G(D)$ with $|A(H)| > 0$. (Thus an isolated vertex of D is not a weak component.) As in the previous subsection, we start with a formal definition of a δ^+ -slicing (δ^- -slicing).

Definition 3.2.7 Let D be a digraph with $A(D) \neq \emptyset$.

(i) A sequence of disjoint arc subsets $S = (C_1, C_2, \dots, C_s)$ of D is a δ^+ -slicing (or δ^- -slicing, respectively) of D if each $C_i \neq \emptyset$ for $1 \leq i \leq s$, and if each of the following holds:

(i-1) Let $D_1 = D$. There exists a vertex $v_1 \in V(D_1)$ such that $C_1 = \partial_{D_1}^+(v_1)$

($C_1 = \partial_{D_1}^-(v_1)$, respectively).

(i-2) For $2 \leq i \leq s$, set $D_i = D_{i-1} - C_{i-1}$, and there exists a vertex $v_i \in V(D_i)$ such that $C_i = \partial_{D_i}^+(v_i)$ ($C_i = \partial_{D_i}^-(v_i)$, respectively).

(i-3) $A(D_s) - C_s = \emptyset$.

(ii) A δ^+ -slicing (or a δ^- -slicing, respectively) $S = (C_1, C_2, \dots, C_s)$ is **minimal** if for each i with $1 \leq i \leq s$, there exists a weak component L_i of D_i such that $|C_i| = \delta^+(L_i)$ (or $|C_i| = \delta^-(L_i)$, respectively).

(iii) Let $\mathcal{S}^+(D)$ and $\mathcal{S}^-(D)$ denote the collections of all δ^+ -slicing and all δ^- -slicing of D , respectively.

By Definition 3.2.7, if $S = (C_1, C_2, \dots, C_m)$ is a δ^+ -slicing of D , then for each $i = 1, 2, \dots, m$,

there exists a weak component D'_i of D_i and a vertex $v_i \in V(D'_i)$ such that $C_i = \partial_{D'_i}^+(v_i)$.

(3.2)

Proof of Theorem 3.1.2(ii) and (iii). By symmetry, it suffices to prove Theorem 3.1.2(ii). Let $h = \min\{\max\{|C_i| : 1 \leq i \leq s\} : S = (C_1, C_2, \dots, C_s) \in \mathcal{S}^+(D)\}$. By Definition 3.2.7, there exists a subdigraph $H \subseteq D$ such that $\bar{\delta}^+(D) = \delta^+(H)$.

Let $S = (C_1, C_2, \dots, C_s) \in \mathcal{S}^+(D)$ be an arbitrary δ^+ -slicing of D . By (3.2), there exists a vertex $z \in V(H)$ such that $C_j \cap A(H) = \partial_H^+(z)$. It follows that $\max_{1 \leq i \leq m}\{|C_i|\} \geq |C_j| \geq |C_j \cap A(H)| \geq \delta^+(H) = \bar{\delta}^+(D)$. Since $S = (C_1, C_2, \dots, C_s) \in \mathcal{S}^+(D)$ is arbitrary, we have

$$h = \min\{\max\{|C_i| : 1 \leq i \leq s\} : S = (C_1, C_2, \dots, C_s) \in \mathcal{S}^+(D)\} \geq \bar{\delta}^+(D).$$

Conversely, let $h' = \min\{\max\{|C_i| : 1 \leq i \leq s\} : S = (C_1, C_2, \dots, C_s) \in \mathcal{S}^+(D) \text{ is a minimal } \delta^+\text{-slicing of } D\}$. Thus by definition, we have $h \leq h'$. We will show that $h' \leq \bar{\delta}^+(D)$, which would imply that $h \leq h' \leq \bar{\delta}^+(D)$ to complete the proof. Let $S = (C_1, C_2, \dots, C_s)$ be an arbitrary minimal δ^+ -slicing of D . By Definition 3.2.7 (i-3),

$A(D) - \bigcup_{i=1}^s C_i = D_s - C_s = \emptyset$. As $H \neq \emptyset$, we observe that $A(H) \subseteq \bigcup_{i=1}^s C_i$, and so there exists a smallest integer j such that $C_j \cap A(H) \neq \emptyset$. By (3.2), for each i , with $1 \leq i \leq m$, there exists a weak component L_i of D_i and a vertex $v_i \in V(L_i)$ such that $|C_i| = \delta_{L_i}^+(v_i) \leq \bar{\delta}^+(D)$. It follows that $\max\{|C_i| : 1 \leq i \leq m\} \leq \bar{\delta}^+(D)$. Since

$h' = \min\{\max\{|C_i| : 1 \leq i \leq s\} : S = (C_1, C_2, \dots, C_s) \in \mathcal{S}^+(D) \text{ is a minimal slicing of } D\}$,

then $h' \leq \bar{\delta}^+(D)$. This proves that Theorem 3.1.2(ii) must hold. The proof for Theorem 3.1.2 (iii) is similar and will be omitted. \square

The arguments deployed in the proof of Theorem 3.1.2(ii) and (iii) also suggest some computational useful ways of determining $\bar{\delta}^+(D)$ and $\bar{\delta}^-(D)$, as stated in the following results.

Lemma 3.2.8 (*Anderson et al.*) *Let v be a vertex with minimum out-degree in a non-trivial digraph D , then each of the following holds:*

$$(i) \bar{\delta}^+(D) = \max\{\delta^+(D), \bar{\delta}^+(D - v)\}.$$

$$(i) \bar{\delta}^-(D) = \max\{\delta^-(D), \bar{\delta}^-(D - v)\}.$$

Proof. By symmetry, it suffices to prove Lemma 3.2.8(i). As D and $D - v$ are subdigraphs of D , it follows that $\bar{\delta}^+(D) \geq \max\{\delta^+(D), \bar{\delta}^+(D - v)\}$.

Conversely, let H be a subdigraph of D such that $\bar{\delta}^+(D) = \delta^+(H)$. If $v \notin H$, then H is also a subdigraph of $D - v$. Thus, $\bar{\delta}^+(D) = \bar{\delta}^+(D - v)$. If $v \in H$, then $\bar{\delta}^+(D) = \delta^+(H) \leq d_H^+(v) \leq d_D^+(v) = \delta^+(D)$. Thus, $\bar{\delta}^+(D) \leq \max\{\delta^+(D), \bar{\delta}^+(D - v)\}$. This implies the lemma. \square

Lemma 3.2.9 (*Anderson et al.*) *Let D be a digraph with $A(D) \neq \emptyset$.*

(i) *If $S = (C_1, C_2, \dots, C_s)$ is a minimal δ^+ -slicing of D , then*

$$\bar{\delta}^+(D) = \max_{1 \leq i \leq s} \{|C_i|\}.$$

(ii) *If $S = (C_1, C_2, \dots, C_s)$ is a minimal δ^- -slicing of D , then*

$$\bar{\delta}^-(D) = \max_{1 \leq i \leq s} \{|C_i|\}.$$

Proof. By symmetry, it suffices to prove Lemma 3.2.9(i). Let $S = (C_1, C_2, \dots, C_s)$ be a minimal δ^+ -slicing of D . Since a minimal δ^+ -slicing is also a δ^+ -slicing, it follows from Theorem 3.1.2(ii) that

$$\bar{\delta}^+(D) = \max\{\min\{d_H^+(v) : v \in V(H)\} : H \subseteq D\} \leq \max_{1 \leq i \leq s} \{|C_i|\}.$$

On the other hand, there exists an ℓ with $1 \leq \ell \leq s$ such that $|C_\ell| = \max_{1 \leq i \leq s} \{|C_i|\}$. By Definition 3.2.7 (ii), there exists a subdigraph L_ℓ of D such that $|C_\ell| = \delta^+(L_\ell)$. It follows that

$$\bar{\delta}^+(D) = \max\{\min\{d_H^+(v) : v \in V(H)\} : H \subseteq D\} \geq \delta^+(L_\ell) = |C_\ell| = \max_{1 \leq i \leq s} \{|C_i|\}.$$

This justifies Lemma 3.2.9(i). The proof for Lemma 3.2.9(ii) is similar and will be omitted.

□

3.2.3 Maximum Subdigraph Strong Connectivity

Throughout this subsection, we assume that D is a digraph which is not spanned by a complete digraph. If $X \subseteq V(D)$ is a subset, then $D[X]$ denotes the subdigraph of D induced by X . Following [8], for a pair of distinct vertices $u, v \in V(D)$, a vertex subset $S \subset V(D) - \{u, v\}$ is a (u, v) -**separator** if $D - S$ contains no directed (u, v) -paths. A subset $S \subset V(D)$ is a (u, v) -**separator** of D if for some $u, v \in V(D)$, S is a (u, v) -separator. A separator of a strong digraph D is **minimum** if $|S|$ is the smallest among all separators of D . Thus by definition, for a strong digraph D , $\kappa(D) = |S|$ for any minimum separator of D .

Lemma 3.2.10 (*Anderson et al.*) *Let D be a strongly connected digraph which is not spanned by a complete digraph, and $S \subset V(D)$ be a minimum separator of D . Let H_1, H_2, \dots, H_c be the strong components of $D - S$. Then $c \geq 2$ and*

$$\bar{\kappa}(D) = \max\{|S|, \max_{1 \leq i \leq c} \{\bar{\kappa}(D[V(H_i) \cup S])\}\}. \quad (3.3)$$

Proof. Since D is a strongly connected digraph and S is a separator, we have $c \geq 2$. Since S is a minimum separator of D , we have $|S| = \kappa(D)$. Since $\bar{\kappa}(D) = \max\{\kappa(H) :$

$H \subseteq D\}$, then $\bar{\kappa}(D) \geq \kappa(H)$ for every $H \subseteq D$. Thus $\bar{\kappa}(D) \geq \kappa(D) = |S|$. For each i with $1 \leq i \leq c$, $\bar{\kappa}(D[V(H_i) \cup S]) = \max\{\kappa(J) : J \subseteq V(D[V(H_i) \cup S])\}$. Since $D[V(H_i) \cup S] \subseteq D$ for each i , then $\bar{\kappa}(D) \geq \bar{\kappa}(D[V(H_i) \cup S])$ for every i . Thus we have

$$\bar{\kappa}(D) \geq \max\{|S|, \max_{1 \leq i \leq c} \{\bar{\kappa}(D[V(H_i) \cup S])\}\}. \quad (3.4)$$

Conversely, let H be a sub-digraph of D such that $\bar{\kappa}(D) = \kappa(H)$. If for some i with $1 \leq i \leq c$, we have $V(H) \subseteq V(H_i) \cup S$, then $H \subseteq D[V(H_i) \cup S]$. In this case, $\bar{\kappa}(D) = \kappa(H) \leq \bar{\kappa}(D[V(H_i) \cup S]) \leq \max_{1 \leq i \leq c} \{\bar{\kappa}(D[V(H_i) \cup S])\}$. Then

$$\bar{\kappa}(D) \leq \max\{|S|, \max_{1 \leq i \leq c} \{\bar{\kappa}(D[V(H_i) \cup S])\}\}.$$

Now assume that for any i with $1 \leq i \leq c$, $V(H) \subseteq V(H_i) \cup S$ does not hold. This implies that there exists some distinct i and j with $1 \leq i < j \leq c$ such that $V(H) \cap V(H_i) \neq \emptyset$ and $V(H) \cap V(H_j) \neq \emptyset$. Thus S contains a separator of H . Hence, in this case, $|S| \geq \kappa(H) = \bar{\kappa}(D)$. Thus we have proved that, in any case,

$$\bar{\kappa}(D) \leq \max\{|S|, \max_{1 \leq i \leq n} \{\bar{\kappa}(\langle X_i \cup S \rangle)\}\}.$$

This, together with (3.4), Lemma 3.2.10 follows. \square

3.3 Applications to Algorithms

In [90], Matula indicated that contrasting with the situation that computing the maximum clique of a graph is an NP-complete problem, the values $\bar{\delta}(G)$, $\bar{\kappa}'(G)$ and $\bar{\kappa}(G)$ of a graph G are polynomially determinable. In this section, we shall show that the corresponding computational problems in digraph also have polynomial time solutions. Throughout this section, we always assume that D is a digraph on n vertices and m edges, for some positive integers m and n . The main results stated in Section 2 can be applied to computationally determine the parameters $\bar{\delta}^+(D)$, $\bar{\delta}^-(D)$, $\bar{\lambda}(D)$ and $\bar{\kappa}(D)$.

To generate an arc subset of the form $\partial_D^+(v)$ for a vertex v satisfying $d_D^+(v) = \delta^+(D)$, it takes n steps of vertex scanning, and such a procedure is referred to as a minimum out-degree search. It takes at most $n - 1$ minimum out-degree search to generate a minimal

$\bar{\delta}^+(D)$ slicing. Thus by Lemma 3.2.9, it takes $O(n^2)$ time to determine $\bar{\delta}^+(D)$. Similarly, determining $\bar{\delta}^-(D)$ also takes $O(n^2)$ time.

Schnorr [101] showed that $\lambda(D)$ can be computed in by $O(n)$ maximum flow calculations. It is known, with the shortest augmentation path algorithm, each maximum flow calculation runs $O(n^2m)$ time and outputs a minimum direct cut $C = \partial_D^+(X)$ for some non-empty proper subset X with $|C| = \lambda(D)$. Thus a narrow slicing of D can be found in $O(n^4m)$ time, and so by Lemma 3.2.6, $\bar{\lambda}(D)$ can be computationally determined in $O(n^4m)$ time.

In the rest of this section, we explain how Lemma 3.2.10 can be applied to obtain a polynomial algorithm to compute $\bar{\kappa}(D)$.

Gabow [54] found an algorithm to determine a separator S of D with $|S| = \kappa(D)$, (and this algorithm will be referred as Gabow's algorithm below). Gabow's algorithm runs in $O(n^{\frac{5}{2}}m)$ -time. Tarjan [103] presented an $O(n + m)$ -algorithm (referred to as Tarjan's algorithm below) to determine the strong components of a digraph D on n vertices and m arcs. Thus by Lemma 3.2.10, a polynomial algorithm to compute $\bar{\kappa}(D)$ can be found by utilizing Gabow's algorithm and Tarjan's algorithm.

An algorithm computing $\bar{\kappa}$

Input: A digraph D with $n = |V(D)| > 0$ and $m = |A(D)| > 0$.

Output: $\bar{\kappa}(D)$

(Step 1) Set $k := 0$; apply Tarjan's algorithm to determine the strong components of D ;
 set $\mathcal{L} := \{L \text{ is a strong component of } D \text{ with } |V(L)| \geq k + 2\}$.

(Step 2) While $\mathcal{L} \neq \emptyset$,

Find $H \in \mathcal{L}$ so that $|V(H)| = \max\{|V(H_i)| : H_i \in \mathcal{L}\}$.

(Step 2.1) If H is spanned by a complete digraph, then

(Step 2.1.1) **updating k** : set $k := \max\{k, |V(H)| - 1\}$, and

(Step 2.1.2) **updating \mathcal{L}** : set $\mathcal{L} := \mathcal{L} - \{H\}$.

(Step 2.2) If H is not spanned by a complete digraph, then run Gabow's algorithm to determine a minimum separator $S \subset V(H)$ of H .

(Step 2.2.1) **updating k** : Set $k := \max\{k, |S|\}$.

(Step 2.2.2) **updating \mathcal{L}** : Apply Tarjan's algorithm to determine the strong

components $H_1, H_2, \dots, H_{c'}$ of $H - S$.

Set $\mathcal{L} := \mathcal{L} - \{H\} \cup \{D[V(H_j) \cup S] : |V(H_j) \cup S| \geq k + 2, \text{ and } 1 \leq j \leq c'\}$.

By the rule we update the value of k at Step 2.1.1 or Step 2.2.1 anytime $k \leq \bar{\kappa}(D)$. By lemma 3.2.10, when the algorithm stops, it will output $k = \bar{\kappa}(D)$ for any digraph D . It suffices to show that the algorithm will stop for any inputting digraph D . Define

$$h(\mathcal{L}) = \sum_{L \in \mathcal{L}} (|V(L)| - (\bar{\kappa}(D) + 2)).$$

Thus $h(\mathcal{L}) \leq |V(D)| - \kappa(D)$. By the rule we update the value of \mathcal{L} at Step 2.1.2 or Step 2.2.2, after each iteration of Step 2, the value of $h(\mathcal{L})$ is reduced by at least 1, and so it takes at most $h(\mathcal{L}) \leq n - \kappa(D)$ iterations executing Step 2. This implies that the algorithm must stop.

At each Step 2 iteration, Gabow's algorithm runs in $O(n^{\frac{5}{2}}m)$ -time. and Tarjan's algorithm runs in $O(n + m)$ -time. As there will be $O(n)$ -time Step 2 iteration, it follows that this algorithm will run in $O(n^{\frac{7}{2}}m)$ -time.

The main purpose of this section is to indicate that there exist polynomial algorithms to computationally determine $\bar{\delta}^+(D)$, $\bar{\delta}^-(D)$, $\bar{\lambda}(D)$ and $\bar{\kappa}(D)$. Efforts have not been spent on finding the fastest algorithms to compute these invariants. Improvement on computational complexity can be made with further discussions. For example, it is known (Even [43], Galil [55]) that given a digraph D with n vertices and m edges, and an integer k , there exists an algorithm to determine if $\lambda(D) \geq k$ in $O(knm)$ time. Using matroid intersection and based on Edmonds branching theorem, Gabow [53] determines $\lambda(D)$ in $O(\lambda(D)m \log(\frac{n^2}{m}))$ time. These algorithms could be applied to improve the complexity of finding a narrow slicing of a digraph D , thereby determining $\bar{\lambda}(D)$ by Lemma 3.2.6.

Chapter 4

Future Work

4.1 Introduction

Complex networks can be modeled as graphs though they usually have features that do not occur in simple graphs. Complex networks include airline routes, power grids, and the Internet to name a few. Robustness of complex networks focuses on the network functioning after failure that could be a random failure, a hardware failure, or sabotage. In terms of graph theory, robustness focuses on the connectivity of a network.

Much of the previous research has focused on what Klau and Weiskircher [76] term as the worst-case which focuses on the minimum number of edges (vertices) that if removed will disable the network. While removing number of edges (vertices) the minimum gives the possibility of disabling the network, but it is not guaranteed. Since disabling the network is not guaranteed, a natural question is to ask what is the probability that removing an arbitrary minimum set of edges (vertices) disables the network?

In Ariannejad and Tusserkani [6], they ask this question for arbitrary spanning trees rather than an arbitrary minimum set. They define an *st*-robustness of a graph G , denoted $STR(G)$, that sums up the expected value of the number of edge-disjoint spanning trees that remain after removing an arbitrary spanning tree. They classify spanning trees

based on how many edge-disjoint spanning trees remain after the tree is removed. Their main results focus on relationships between classical connectivity measures and defined spanning tree types.

The purpose of this chapter is to build the necessary terminology for digraphs and to present future work as investigating if a similar st -robustness for digraphs will be a useful connectivity measure. In the next section, we discuss previous work and in the last section we discuss our current progress on digraphs.

4.2 Previous Work on Spanning Trees

Recall that a tree, T , is a spanning tree of G if $T \subseteq G$ and $V(T) = V(G)$. Let $\tau(G)$ be the number of spanning trees contained in a graph G . Cayley [32] developed a formula for calculating the total number of spanning trees of complete graphs.

Theorem 4.2.1 (*Cayley's Formula [32]*) *Let G be a complete graph. Then $\tau(G) = n^{n-2}$.*

Two spanning trees of G , T_1 and T_2 are edge disjoint spanning trees if $E(T_1) \cap E(T_2) = \emptyset$. The spanning tree number, $STP(G)$, is defined as the maximum number of edge-disjoint spanning trees contained in G . In 1961, both Nash-Williams [93] and Tutte [107] gave independently the first result on $STP(G)$. First, let P be a partition of $V(G)$. Nash-Williams [93] defines $E_P(G)$ to be the set of edges in G that connect vertices between two different members of the partition P . Using this terminology, the following is their classical result:

Theorem 4.2.2 (*Nash-Williams [93], Tutte [107]*) *A graph G has k -edge disjoint spanning trees if and only if $|E_P(G)| \geq k(|P| - 1)$ for every partition P of $V(G)$.*

Catlin [30] made an improvement on the Nash-Williams/Tutte theorem.

Theorem 4.2.3 (Catlin [30]) *The edge connectivity of G satisfies $\kappa'(G) \geq 2k$ if and only if for any set E_k of k edges of G , the subgraph $H = G - E_k$ has k -edge disjoint spanning trees.*

Corollary 4.2.4 ([96], [78], [62]) *If $\kappa'(G) \geq 2k$, then G has k -edge disjoint spanning trees.*

While there has been other studies on the STP number of graph, Ariannejad and Tusserkani developed a new connectivity measure that involves the STP, but doesn't directly try to calculate it.

Let G be a graph with $STP(G) = t$ for some positive integer t . Ariannejad and Tusserkani [6] classified the types of spanning trees by the following definition: A spanning tree T of G is defined to be type ST_i if $STP(G \setminus T) = i - 1$ for $1 \leq i \leq t$. They define $ST_i(G)$ to be the set of all spanning trees of type ST_i . They define the cardinality of $ST_i(G)$ to be denoted $\tau_i(G)$. Let $\mathcal{ST}(G)$ be the set of all spanning trees of a graph G and let $\tau(G)$ be the cardinality of the set.

Ariannejad and Tusserkani's [6] motivation for partitioning the spanning trees is to answer the question "What is the probability that a graph G is disconnected by the random deletion of spanning trees?" Using these partitions, they developed a new connectivity measure called the st-robustness of a graph G .

Definition 4.2.5 (Ariannejad and Tusserkani [6]) *The **st-robustness** of a graph G , denoted $STR(G)$, is the expected value of $STP(G \setminus T)$ summed over every $T \in \mathcal{ST}(G)$. Formally,*

$$STR(G) = \sum_i \frac{\tau_i(G)}{\tau(G)} (i - 1) \text{ for } 1 \leq i \leq t.$$

Theorem 4.2.6 (Ariannejad and Tusserkani [6]) *The set of all $ST_i(G)$'s ($1 \leq i \leq t$) gives a partition of $\mathcal{ST}(G)$ and all equivalence classes are non-empty.*

Ariannejad and Tusserkani [6] was able to give lower bounds for τ_i as seen in the following theorem:

Theorem 4.2.7 (*Ariannejad and Tusserkani [6]*) *Let G be a graph with $|E(G)| = e$. Then*

$$(i) \tau(G) \geq \left\lceil \frac{e}{(n-1)} \right\rceil$$

$$(ii) \tau_i(G) \geq t \text{ for } 2 \leq i \leq t-1$$

$$(iii) \tau_t(G) \geq \left(\binom{t}{2} (2^{n-1} - 2) \right) + t$$

Theorem 4.2.8 (*Ariannejad and Tusserkani [6]*) *Let G be a graph with n vertices. If G has less than $2n - 2$ distinct cycles, then $\tau_1(G) = \tau(G)$.*

In graph theory, we define an **ear** of a graph G to be a maximal path such that all internal vertices have degree two. Let $H \subseteq G$. West [112] defines an **ear decomposition** of G beginning at H to be a decomposition P_0, P_1, \dots, P_k with $P_0 = H$ and P_i for $i > 0$ is an ear of the union P_0, P_1, \dots, P_i . Ariannejad and Tusserkani [6] define a P_l -ear decomposition of G beginning at H such that all P_i for $i > 0$ the length of P_i is l . Similarly, they define a $K_{1,r}$ -decomposition of G beginning at H to be $P_0 = H$ and every P_i for $i > 0$ is a $K_{1,r}$ star graph. A $K_{1,r}$ star graph is a graph with a single vertex that has r leaves attached. Using ear decompositions they were able to develop lower bounds on the STP of some noncomplete graphs.

Theorem 4.2.9 (*Ariannejad and Tusserkani [6]*) *Let G be a simple connected leafless graph. Then $STP(G) \geq 2$ if and only if G has a P_2 -ear decomposition beginning from a subgraph H , where $STP(H) \geq 2$.*

And a more generalized version of Theorem 4.2.9:

Theorem 4.2.10 (*Ariannejad and Tusserkani [6]*) *Let G be a simple connected leafless graph. Then $STP(G) \geq r$ if and only if G has a $K_{1,r}$ ear decomposition beginning from a subgraph H where $STP(H) \geq r$.*

Let $v \in V(G)$. Let $C = (V - \{v\}, \{v\})$ be a cut of G . Let T_v be the set of all spanning trees of G that is created by extending the edge cut C to a spanning tree [6].

Theorem 4.2.11 (*Ariannejad and Tusserkani [6]*) *Let G be a graph with n vertices and $\delta(G) = k$. If*

$$k \geq \left\lfloor \frac{(n+2)}{2} \right\rfloor,$$

then a spanning tree T is of type ST_1 if and only if there exists a vertex $v \in V$ such that $T \in T_v$, which which we have

$$\left\lceil \frac{e}{(n-1)} \right\rceil \leq \tau_1(G) \leq n$$

.

Theorem 4.2.12 (*Ariannejad and Tusserkani [6]*) *Let $n \geq 6$. Then a spanning tree T of K_n is of type ST_2 if and only if $\Delta(T) = n - 2$.*

Corollary 4.2.13 (*Ariannejad and Tusserkani [6]*) *For $n \geq 3$ we have $\tau_1(K_n) = n$ and for $n \geq 6$ we have $\tau_2(K_n) = n(n-1)(n-2)$.*

Lemma 4.2.14 (*Ariannejad and Tusserkani [6]*) *Let S be a forest subgraph of K_{2n+1} with $|S| \leq n - 1$. Then $STP(K_{2n+1} - S) \geq n - 1$.*

Theorem 4.2.15 (*Ariannejad and Tusserkani [6]*) *Let i and n be positive intgers such that $n \geq 2i + 2$ and consider T to be a spanning tree in K_n . Then $T \in ST_i(K_n)$ if and only if $\Delta(T) = n - i$.*

Conjecture 4.2.16 (*Ariannejad and Tusserkani [6]*)

$$\lim_{n \rightarrow \infty} \frac{STR(K_n)}{n} = 1$$

4.3 Future Work

Let D be a strict strongly connected digraph and let H be a subdigraph of D . By $D \setminus H$ we mean a subdigraph of D obtained by deleting all arcs of H from D . A spanning arborescence T is a subdigraph of D in which for $V(T) = V(D)$, and for some vertex r , denoted the root, any other vertex $v \in V(D)$, $v \neq r$, there exists a unique dipath from r to v . Two spanning arborescences of D , T_1 and T_2 , are said to be arc disjoint if they do not have any common arcs (i.e. $A(T_1) \cap A(T_2) = \emptyset$).

The spanning tree packing number of a graph G , $STP(G)$, can be extended to spanning arborescences. We will define the spanning arborescence packing number of a digraph D , denoted, $SAP(D)$, to be the maximum number of arc disjoint spanning arborescences contained in D . Let $\mathcal{SA}(D)$ denote set of all spanning arborescences of D . We denote $\tau(D)$ to be the cardinality of $\mathcal{SA}(D)$.

While partitioning spanning trees into types was straightforward, partitioning spanning arborescences into types is more complex. One way that we could develop a partition is based on a free root. In this case, we say that a spanning arborescence T of D is of type SA_i ($1 \leq i \leq SAP(D)$) if the maximum number of arc disjoint spanning arborescences contained in $D \setminus A$ is $i - 1$ (i.e. $SAP(D \setminus A) = i - 1$).

Another possibility is based on a fixed root. In this case, we say that a spanning arborescence T rooted at v for some $v \in V(D)$ of D is of type SA_i ($1 \leq i \leq SAP(D)$) if the maximum number of arc disjoint spanning arborescences rooted at v contained in $D \setminus A$ is $i - 1$.

A generalization of the two previous definitions is as follows. Let $S \subseteq V(D)$. We say that a spanning arborescence T rooted at v for some $v \in S$ of D is of type SA_i ($1 \leq i \leq SAP(D)$) if the maximum number of arc disjoint spanning arborescences rooted at any $v \in S$ contained in $D \setminus A$ is $i - 1$. Notice that the first definition is established when $S = V(D)$ and the second definition when $S = \{v\}$. In all of the definitions we let $\tau_i(D)$ to be the cardinality of $SA_i(D)$.

Definition 4.3.1 st-robustness of a directed graph D , $STR(D)$, is defined as the expected value of $SAP(D \setminus A)$, when A ranges over all spanning arborescences of D . Formally,

$$STR(D) = \sum_i \frac{\tau_i(D)}{\tau(D)}(i - 1) \text{ for } 1 \leq i \leq t.$$

It is important to note that strong-connectivity of D is needed. Consider K_3^* . Let $V(K_3^*) = \{v_1, v_2, v_3\}$. Let T be the spanning arborescence rooted at v_1 with $E(T) = \{(v_1, v_2), (v_1, v_3)\}$. Notice that the strong connectivity is broken, as there is no (v_1, v_3) -dipath (or (v_1, v_2) -dipath). However, the graph is still weakly connected. If we used weakly connected, then we could not produce any spanning arborescences of type ST_1 since any spanning arborescence was removed in K_3^* would still be weakly connected.

While Cayley's formula gives an easy way to calculate $\tau(G)$, there is no easy formula for calculating $\tau(D)$. The best that we can currently do is calculate the number of spanning arborescences that are rooted at a vertex.

Theorem 4.3.2 (*Tutte's Directed Matrix-Tree Theorem [106]*) Let D be a digraph with $V(D) = \{v_1, v_2, \dots, v_n\}$ and let L be an $(n \times n)$ matrix whose entries are given by

$$L_{ij} = \begin{cases} \delta^+(v_j) & \text{for } i = j \\ -1 & \text{for } i \neq j \text{ and } (v_i, v_j) \in A(D) \\ 0 & \text{otherwise} \end{cases}$$

then the number N_j of spanning arborescences rooted at v_j is $N_j = \det(\hat{L}_j)$ where \hat{L}_j is the matrix produced by deleting the j th row and column from L .

If we only look at complete digraphs, K_n^* , then we can build the matrix

$$L_{ij} = \begin{cases} n - 1 & \text{for } i = j \\ -1 & \text{otherwise} \end{cases}$$

and the matrix \hat{L}_j is an $(n-1) \times (n-1)$ matrix with $(n-1)$ down the diagonal and -1 for every other entry. Using this information, a program can be written to quickly calculate

the total number of spanning arborescences rooted at v_j . Since D is complete, then $N_i = N_j$ for all i and thus $\tau(K_n^*) = n(N_j)$. Below is a program that can be run in MATLAB that determines $\tau(K_n^*)$ for any vertex size n within computational capabilities.

An algorithm computing $\tau(K_n^*)$ in MATLAB

Input: Vertex size n

Output: $\tau(K_n^*)$

```
n=input('Give the number of vertices: ');
```

```
A=ones(n-1);
```

```
v(1:n-1)=n;
```

```
D=diag(v);
```

```
L=D-A;
```

```
d=n*det(L)
```

Future work is to fully develop *st*-robustness in digraphs and explore which definition will be best suited for this research. Then, we plan to investigate what, if any, theorems extend from graphs to digraphs.

Bibliography

- [1] W. M. P. van der Aalst, B. F. van Dongen, J. Herbst, L. Maruster, G. Schimm, and A. J. M. M. Weijters, Workflow Mining: A Survey of Issues and Approaches, *Data & Knowledge Engineering*, 47 (2003) 237-267.
- [2] J. Akiyama, F. Boesch, H. Era, F. Harary, and R. Tindell, The cohesiveness of a point of a graph, *Networks* 11 (1981) 65-68.
- [3] A. T. Amin and S. L. Hakimi, On the design of reliable network, *Networks* 3 (1973) 241-260.
- [4] I. Anderson, Sufficient conditions for matchings, *Proc. Edinburgh Math. Soc.* 18 (1972) 129-136.
- [5] J. Anderson, H.-J. Lai, X. Lin and M. Xu, On k -maximal strength digraphs, *J. Graph Theory* 84 (2017) 17-25.
- [6] M. Ariannejad and R. Tusserkani, A New Connectivity Measure for Graphs, preprint.
- [7] M. O. Ball, Complexity of network reliability computations, *Networks* 10 (1980) 153-165.
- [8] J. Bang-Jensen and G. Gutin. *Digraphs: Theory, Algorithms and Applications* 2nd Edition. Springer-Verlag, London, 2009.
- [9] D. Bauer, On regular line graphs, *Topics in Graph Theory*, *Annals of the New York Academy of Sciences* 328 (F. Harary, ed.) (1979) 30-31.

- [10] F. T. Boesch, The strongest monotone degree condition for n -connectedness of a graph, *J. Comb. Th. Series B* 16 (1974) 162-165.
- [11] F. T. Boesch, *Large Scale Networks*, IEEE Press, New York, 1976.
- [12] F. T. Boesch and D. Butler, On minimum multigraphs with prescribed lower bounds on local line connectivities, *IEEE Trans. on Circuits and Systems* 23 (1976) 821-823.
- [13] F. T. Boesch and D. Butler, Derivation of the unitary case of minimum multigraphs with prescribed lower bounds on local line-connectivities, *IEEE Trans. on Circuits and Systems* 26 (1980) 642-644.
- [14] F. T. Boesch and S. Chen, A generalization of line-connectivity and optimally invulnerable graphs, *SIAM J. Appl. Math.* 34 (1978) 657-665.
- [15] F. T. Boesch and A. P. Felzer, On the minimum m -degree vulnerability criterion, *IEEE Trans. on Circuit Theory* 18 (1971) 224-228.
- [16] F. T. Boesch and A. P. Felzer, A general class of invulnerable graphs, *Networks* 2 (1972) 261-283.
- [17] F. T. Boesch and A. P. Felzer, On the invulnerability of the regular complete k -partite graphs, *SIAM J. Appl. Math.* 20 (1971) 176-182.
- [18] F. T. Boesch and F. Harary, Unicycle realizability of degree lists, *Networks* 8 (1978) 93-96.
- [19] F. T. Boesch and F. Harary, Line removal algorithms for graphs and their degree lists, *IEEE Trans. on Circuits and Systems* 23 (1976) 778-782.
- [20] F. T. Boesch, F. Harary, and J. A. Kabell, Graphs as models of communication network vulnerability: connectivity and persistence, *Networks* 11 (1981) 57-63.
- [21] F. T. Boesch and J. A. M. McHugh, An edge extremal result for subcohesion. *J. Combinat. Theory Ser. B.* 38 (1985) 1-7.
- [22] F. T. Boesch and J. A. M. McHugh, Synthesis of biconnected graphs, *IEEE Trans. on Circuits and Systems* 21 (1974) 330-334.

- [23] F. T. Boesch and R. E. Thomas, On graphs of invulnerable communication sets, *IEEE Trans. on Circuit Theory* 17 (1970) 183-192.
- [24] B. Bollobás, A problem of the theory of communication networks, *Theory of Graphs* (G. Katona and P. Erdos, eds.), Akad. Kiado, Budapest (1968) 29-36.
- [25] B. Bollobás and F. Harary, Extremal graphs with given diameter and connectivity, *ARS Combinatoria* 1 (1976) 281-296.
- [26] J. A. Bondy, Properties of graphs with constraints on degrees, *Studia Sci. Math. Hungar.* 4 (1966) 473-475.
- [27] J. A. Bondy and U. S. R. Murty, *Graph Theory*. Springer, New York, 2008.
- [28] L. Caccetta, On extremal graphs with given diameter and connectivity, *Topics in Graph Theory*, *Annals of New York Academy of Sciences* 328 (F. Harary, ed.), New York (1979) 76-94.
- [29] L. Caccetta and R. Häggkvist, On diameter critical graphs, *Discrete Math.* 28 (1979) 223-229.
- [30] P. A. Catlin, The reduction of graph families closed under contraction, *Discrete Mathematics*, 160 (1996) 67-80.
- [31] P. A. Catlin, H. J. Lai and Y. Shao, Edge-connectivity and edge-disjoint spanning trees, *Discrete Mathematics*, 309 (2009) 1033-1040.
- [32] A. Cayley, A theorem on trees, *Quart. J. Math.*, 23 (1889) 376-378.
- [33] G. Chartrand, A graph theoretic approach to a communication problem, *SIAM J. Appl. Math.* 14 (1966) 778-781.
- [34] G. Chaty and M. Chein, Minimally 2-edge connected graphs, *J. Graph Theory* 3 (1979) 15-22.
- [35] W. Chou and H. Frank, Survivable communication networks and the terminal capacity matrix, *IEEE Trans. on Circuit Theory* 17 (1970) 192-197.
- [36] V. Chvatal, Tough graphs and hamiltonian circuits, *Discrete Math.* 5 (1973) 215-228.

- [37] G. B. Dantzig, and D. R. Fulkerson, On the max-flow min-cut theorem of networks, Linear Inequalities and Related Systems, Annals of Math. Study 38, Princeton University Press (1956) 215-221.
- [38] G. A. Dirac, Minimally 2-connected graphs, J. Reine Angew. Math. 228 (1967) 204-216.
- [39] J. Edmonds, Existence of k-edge connected ordinary graphs with prescribed degrees, J. or Research of NBS 68 (1964) 73-74.
- [40] A-H. Esfahanian, Connectivity algorithms, Topics in Structural Graph Theory (L. W. Beineke and R. J. Wilson, Eds.), Cambridge University Press (2013) 268-281.
- [41] A-H. Esfahanian and S. L. Hakimi, On computing the connectivities of graphs and digraphs. Networks. 14 (1984) 355-366.
- [42] K. P. Eswaren and R. E. Tarjan, Augmentation problems, SIAM J. Comput. 5 (1976) 653-665.
- [43] S. Even, *Graph Algorithms*, Computer Science Press, Woodland Hills, 1979.
- [44] S. Even and R. E. Tarjan, Network flow and testing graph connectivity, SIAM Journal of Computing 4 (1975), 507-518.
- [45] L. R. Ford and D. R. Fulkerson, Maximal flow through a network. Canadian J. of Math. 8 (1956) 399-404.
- [46] H. Frank, Some new results in the design of survivable networks, Proc. 12th Annual Midwest Circuit Theory Symp., University of Texas (1969) I3.1-I3.8.
- [47] H. Frank, Maximally reliable node weighted graphs, Proc. 3rd Annual Conf. on Information Sciences and Systems, Princeton University (1969) 1-6.
- [48] H. Frank and W. Chou, Connectivity considerations in the design of survivable networks, IEEE Trans. on Circuit Theory 17 (1970) 486-490.
- [49] H. Frank and W. Chou, Topological optimization of computer networks, Proc. IEEE 60 (1972) 1385-1397.

- [50] H. Frank and I. Frisch, *Communication, Transmission and Transportation Networks*, Addison-Wesley, Reading (1971)
- [51] H. Frank and I. T. Frisch, Analysis and design of survivable networks, *IEEE Trans. on Communication Technology* 18 (1970) 501-519.
- [52] I. T. Frisch and M. Malek-Zavavei, Vulnerability of directed communication nets, *IEEE Trans. on Circuit Theory* 19 (1972) 283-284.
- [53] H. N. Gabow, A matroid approach to finding edge-connectivity and packing arborescences, *J. Comput. System Sci.*, 50(2) (1995) 259-273.
- [54] H. N. Gabow, Using expander graphs to find vertex connectivity. *J. Assoc. Comput. Mach.*, 53 (2006) 800-844.
- [55] Z. Galil, Finding the vertex connectivity of graphs, *SIAM J. Comput.* 9(1) (1980) 197-199.
- [56] M. R. Garey and D. S. Johnson, *Computers and Intractability, A guide to the theory of NP-Completeness*, Freeman, San Francisco (1979).
- [57] D. Geller, F. Harary, Connectivity in Digraphs, in: *Recent Trends in Graph Theory, Proceedings of the First New York City Graph Theory Conference, 1970*, Lecture Notes in Mathematics. 186 (1971) 105-115.
- [58] D. L. Goldsmith, On the second-order edge-connectivity of a graph, *Congr. Numer.* 29 (1980) 479-484.
- [59] D. L. Goldsmith and R. C. Entringer, A sufficient condition for equality of edge-connectivity and minimum degree of a graph, *J. Graph Theory* 3 (1979) 251-255.
- [60] R. E. Gomory and T. C. Hu, Multi-terminal network flows, *SIAM J. Appl. Math.* 9 (1961) 551-570.
- [61] G. Günther and B. L. Hartnell, On minimizing the effects of betrayals in a resistance movement, *Proc. of 8th Manitoba Cont. on Numerical Math. and Computing*, University of Manitoba, Winnipeg (1978) 285-306.

- [62] D. Gusfield, Connectivity and edge-disjoint spanning trees, *Inform. Process. Lett.* 16 (1983) 87-89.
- [63] S. L. Hakimi, On an algorithm for construction of the least vulnerable communication network on the graph with maximum connectivity, *IEEE Trans. Circuit Theory* 16 (1969) 229-230.
- [64] S. L. Hakimi, On the existence of graphs with prescribed degrees and connectivity, *SIAM J. Appl. Math.* 26 (1974) 154-164.
- [65] S. L. Hakimi and A. T. Amin, On the design of reliable networks, *Networks* 3 (1973) 241-260.
- [66] S. L. Hakimi and E. F. Schmeichel, Graphs and their degree sequences: a survey, *Proc. Western Mich. Conf. on Graph Theory and Applications*, Springer-Verlag, New York (1976) 225-233.
- [67] R. Halin, A theorem on n -connected graphs, *J. Comb. Th.* 7 (1969) 150-154.
- [68] R. Halin, On the structure of n -connected graphs, *Recent Progress in Combinatorics* (W. T. Tuttle, ed.), Academic Press, New York (1969) 91-102.
- [69] F. Harary, Conditional connectivity. *Networks*. 13 (1983) 346-357.
- [70] F. Harary, *Graph Theory*, Addison-Wesley, Reading (1969).
- [71] F. Harary, The maximum connectivity of graph, *Proc. Nat. Acad. Sci.* 48 (1962) 1142-1146.
- [72] T. E. Harris and F. S. Ross, *Fundamentals of a Method for Evaluating Rail Net Capacities*, 1955.
- [73] G. Hilborn, Measures for distributed processing network survivability, *Proc. National Computer Conference*, Annaheim (1980) 157-163.
- [74] A. M. Hobbs, Survivability of networks under attack, *Applications of Discrete Mathematics* (eds. John G. Michaels and Kenneth H. Rosen), 1991, 332 - 353.

- [75] A. K. Kel'mans, Connectivity of probabilistic networks, *Avtomatika i Telemekhanika* 3 (1966) 98-116.
- [76] G. W. Klau and R. Weiskircher, Robustness and resilience, *Networks. Analysis, LNCS* 3418 (2005) 417-437.
- [77] V. Klee and H. Qualife, Minimum graphs of specified diameter, connectivity and valence, *Math. of Ops. Res.* 11 (1976) 28-31.
- [78] S. Kundu, Bounds on the number of disjoint spanning trees, *J. Combinat. Theory B* 17 (1974) 87-89.
- [79] H.J. Lai, The Size of strength-maximal graphs. *J. Graph Theory.* 14 (1990) 187-197.
- [80] L. Lesniak, Results on the edge-connectivity of graphs, *Discrete Math.* 8 (1974) 351-354.
- [81] P. Li, H.-J. Lai and M. Xu, Disjoint Spanning Arborescences in k -Arc-Strong Digraphs, *Ars Combinatoria*, accepted.
- [82] D. R. Lick, Critically and minimally n -connected graph, *The Many Facets of Graph Theory* (G. Chartrand and S. F. Kapoor, eds.), Springer-Verlag, New York (1969) 199-205.
- [83] X. Lin, S. Fan, H.-J. Lai, and M. Xu, On the lower bound of k -maximal digraphs, *Discrete Math.*, 339 (2016) 2500-2510.
- [84] L. Lovasz, Connectivity in digraphs, *J. Comb. Th. Series B* 15 (1973) 174-177.
- [85] W. Mader, Minimale n -fach kantenzusammenhngende Graphen. *Math. Ann.* 191 (1971) 21-28.
- [86] Y. Mansour and B. Schieber, Finding the edge connectivity of directed graphs, *Journal of Algorithms*, 10 (1989) 76-85.
- [87] D. W. Matula, The cohesive strength of graphs, *The Many Facets of Graph Theory*, *Lecture Notes in Mathematics*, No. 110, G. Chartrand and S. F. Kapoor, eds., Springer-Verlag, Berlin, 1969, 215-221.

- [88] D. W. Matula, Determining edge connectivity in $O(mn)$. Proceedings of 28th Symp. on Foundations of Computer Science. (1987) 249-251.
- [89] D. Matula, K -components, clusters, and slicings in graphs. SIAM J. Appl. Math. 22 (1972) 459-480.
- [90] D. W. Matula, Subgraph connectivity number of a graph, Theory and Applications of Graphs, Lecture Notes in Mathematics, No. 642, Y. Alavi and D. R. Ricks, eds., Springer-Verlag, Berlin, 1976, 371-393.
- [91] K. Menger, Zur allgemeinen Kurventheorie. Fund. Math. 10 (1927) 95-115.
- [92] B. R. Myers, The minimum order 3-connected cubic graphs with specified diameter, IEEE Trans. on Circuits and Systems 27 (1980) 698-709.
- [93] C.St.J.A. Nash-Williams, Edge-disjoint spanning trees of finite graphs, J. London Math. Soc., 36 (1961) 445-450.
- [94] E.M. Palmer, On Spanning Tree Packing Number of a Graph: A Survey, Discrete Mathematics, 230 (2001) 13-21.
- [95] A. N. Patrinos and S. L. Hakimi, Relations between graphs and integer-pair sequences, Discrete Math. 15 (1976) 347-358.
- [96] V. P. Poleskii, A lower bound for the reliability of information, Prob. Inf. Transmission 7 (1971) 165-171.
- [97] A. Rosenthal, Computing the reliability of complex networks, SIAM J. Appl. Math. 32 (1977) 384-393.
- [98] A. Rosenthal and D. Frisque, Transformations for simplifying network reliability calculations, Networks 17 (1977) 97-111.
- [99] S. Schaeffer, Graph clustering, Computer Science Review, 1 (2007) 27-64.
- [100] E. F. Schmeichel and G. S. Bloom, Connectivity, genus and the number of components in vertex-deleted subgraphs, J. Comb. Th. Series B 27 (1979) 198-201.

- [101] C. P. Schnorr, Bottlenecks and edge connectivity in unsymmetrical networks. *SIAM J. Comput.* 8(2) (1979) 265-274.
- [102] G. J. Simmons, Command graphs, *The Proc. of the Internat. Conference on Set Theory and Combinatorics*, Hungarian Academy of Sciences, Keszthely (1973) 1277-1349.
- [103] R. E. Tarjan, Depth-first search and linear graph algorithms, *SIAM Journal of Computing* 2 (1972), 146-160.
- [104] M. Tainiter, A new deterministic network reliability measure, *Networks* 16 (1976) 191-204.
- [105] S. Toueg and K. Steiglitz, The design of small-diameter networks by local search, *IEEE Trans. on Computers* 28 (1979) 537-542.
- [106] W.T. Tutte, The dissection of equilateral triangles into equilateral triangles, *Proc. Cambridge Philos. Soc.*, 44 (1948) 463-482.
- [107] W.T. Tutte, On the problem of decomposing a graph into n connected factor, *J. London Math. Soc.*, 36 (1961) 220-221.
- [108] D. L. Wang, Construction of maximally edge-connected graphs with prescribed degrees, *Studies in Appl. Math.* 55 (1976) 87-92.
- [109] D. L. Wang and D. J. Kleitman, On the existence of n -connected graphs with prescribed degrees, *Networks* 3 (1973) 225-239.
- [110] D. L. Wang and D. J. Kleitman, A note on n -edge connectivity, *SIAM J. Appl. Math.* 26 (1974) 313-314.
- [111] M. E. Watkins, Connectivity of transitive graphs, *J. Comb. Theory* 8 (1970) 23-29.
- [112] D. West, *Introduction to Graph Theory*, 2nd Ed., Prentice Hall, 2001.
- [113] H. Whitney, Congruent graphs and the connectivity of graphs, *Amer. J. Math.* 54 (1932) 150-168.

- [114] R. S. Wilkov, Construction of maximally reliable communication networks with minimum transmission delay, Proc. IEEE International Conf. on Communications 6, IEEE Press, New York (1970) 4210-4215.
- [115] R. S. Wilkov, Analysis and design of reliable computer networks, IEEE Trans. on Communications 20 (1972) 660-678.
- [116] D. R. Woodall, The binding number of a graph and its Anderson number, J. Comb. Theory 15 (1973) 225-255.
- [117] D. R. Woodall, A sufficient condition for hamiltonian circuits, J. Comb. Theory Series B 25 (1978) 184-486.