SPECTRAL AND CHROMATIC PROPERTIES IN VERTEX REMOVAL

GEORGE D. TORRES

Abstract. We analyze a graph that can be constructed a graph G that encodes how G can be disconnected by the process of vertex removal. We call this graph $\mathcal{T}(G)$. Using spectral theory and combinatorial analysis, we prove several theorems about $\mathcal{T}(G)$ concerning the cyclomatic number of G, colorings of G and how connected G is. We will also discuss useful algorithms associated with $\mathcal{T}(G)$. Finally, this paper will cover computational programs that were helpful in approaching this problem.

Contents

1. Introduction and Definitions 1
2. Overview 2
3. Edges and Vertices 4
4. Properties of $\mathcal{T}(G)$ 5
5. Reconstructing G 6
6. Bounding Degrees in $\mathcal{T}(G)$ 7
7. Special Case 11
8. A Cyclomatic Corollary 11
9. $\mathcal{T}(G)$ and the Connectedness of G 12
10. Distances in $\mathcal{T}(G)$ and Coloring G 15
11. Computational Methods 16
12. Future Directions 16
Acknowledgments 16
References 17

1. Introduction and Definitions

Many topics in graph theory deal largely with the study of how connected a graph is. Perhaps a rather simple notion of this is whether or not a graph can be separated into multiple components. A less trivial notion of connectedness, however, arises when we have a graph with only one component (i.e. a completely connected graph). Though we may have two completely connected graphs, one may be more "disconnectable" than the other. In this context, the term "connected," is a superlative, and is difficult to approach; finding an exact measure that detects the degree to which a graph is connected isn't a problem with one answer. Since there are several criteria to measure connectedness by, there are also several established measures (see Hararay, Ch. 5 [3]). This paper seeks to further the discussion of

Date: August 12th, 2013.

connectedness of a graph through a formalization and quantification of the process of dismantling a graph. In particular, we will establish a measure of connectedness through the criterion of completely disconnecting a graph.

We will begin by providing a brief overview of this project and its findings.

2. Overview

Definition 2.1. A *graph* is a finite collection of points and unordered[1] pairs of those points. We denote a graph G as (V, E), where V (or $V(G)$) is the set of vertices of G and E (or $E(G)$) is the set of edges of G. In particular, $E(G) \subset V \times V$. We do not consider graphs with self loops. We also only consider connected graphs.

Graphs are often represented pictorially as networks. See Figure 1 for some examples of a graph.

Definition 2.2. The *degree* of a vertex v is the number of edges incident upon it.

Definition 2.3. The *chromatic number* of a graph G, denoted as $\gamma(G)$, is the number of colors necessary to assign to each vertex such that no two adjacent vertices have the same color. The *multiplicity* of a color is the number of vertices of that color.

Definition 2.4. The *cyclomatic number* of a graph G, denoted $c(G)$, is the minimum number of edges necessary to remove from G so that it has no cycles. For any connected graph $c(G) = 1 + |E(G)| - |V(G)|$ (pg. 27, [1]).

Definition 2.5. A *subgraph* of a graph $G = (V, E)$ is a graph $S = (V', E')$ such that $V' \subset V$ and $E' \subset E$.

Definition 2.6. An *adjacency matrix* A_G is a symmetric matrix associated to a graph built in the following way:

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

In particular the number of walks between vertices v_i and v_j of length ℓ is the (i, j) entry of ${A_G}^{\ell}$ (pg 151, [3]).

This paper addresses the process of deconstructing a graph by removing its vertices. When we say that we *remove* a vertex v_i, we precisely mean that we are making the *ith* row and column of A_G all zero. This then means that every new subgraph created by removing a vertex still has the same number of vertices. In graph theoretic notation, a graph G with vertex v_i removed is denoted as $G_{\setminus v_i}$.

Definition 2.7. We define Φ_G to be the set of distinct subgraphs of G attained by removing any number of vertices of G.

With this, we define a graph associated to a graph G, denoted as $\mathcal{T}(G)$, as follows:

Definition 2.8.

$$\mathcal{T}(G) := \begin{cases} V_i \in V(\mathcal{T}(G)) \text{ iff } V_i \in \Phi_G \\ (V_i, V_j) \in E(\mathcal{T}(G)) \text{ iff } V_i = V_{j_{\setminus v}} \text{ for some } v \in V(G). \end{cases}$$

[1]If the combinations are ordered, it is called a directed graph.

We denote the vertex of $\mathcal{T}(G)$ that corresponds to the subgraph of G with no edges as V_0, and we denote the vertex of $\mathcal{T}(G)$ that corresponds to G itself as V_G. In a more intuitive sense, $\mathcal{T}(G)$ is a graph of the ways to deconstruct G by removing vertices. As an example, see Figure 1.

This paper will provide theorems about $\mathcal{T}(G)$ and how it measures the chromatic and cylcomatic numbers of G as well as how connected G is. These theorems will be proven in later sections.

A vertex of high interest in $\mathcal{T}(G)$ is V_0, the vertex corresponding to the completely disconnected graph. The following theorem places bounds on its degree.

Theorem 2.9. *For a graph $G = (V, E)$ with the ith vertex having k_i adjacent vertices that are in a triangle*[2]

$$\sum_{i \in V(G)} \left(2^{deg(i)-k_i} + k_i - 1\right) \le deg(V_0) \le \sum_{i \in V(G)} \left(2^{deg(i)} - 1\right).$$

Moreover, when G is triangle-free, the upper and lower bounds are equal and

$$deg(V_0) = \sum_{i \in V(G)} \left(2^{deg(i)} - 1\right).$$

The triangle-free hypothesis isn't sharply necessary, and in fact there are many graphs with triangles that still satisfy this equality. However, it is indeterminate as to whether or not all graphs will satisfy it.

The next theorem characterizes the chromatic number of $\mathcal{T}(G)$ in terms of local and global properties of G.

Theorem 2.10. *For a triangle-free graph G with cyclomatic number $c(G)$, we have:*

$$\gamma(\mathcal{T}(G)) \le c(G) + \sum_{i \in V(G)} \left(2^{deg(i)} - deg(i)\right).$$

For similar reasons as above, the triangle-free hypothesis isn't sharply necessary here as well. This theorem shows how the chromatic number of $\mathcal{T}(G)$ is dependent on the degrees of vertices in G and how many loops G has.

We will show that $\frac{deg(V_0)}{v-1}$, (where v is the number of vertices in $\mathcal{T}(G)$) is a possible measure of how connected G is. The following theorem places bounds on this quantity based on spectral properties of $\mathcal{T}(G)$.

Theorem 2.11. *For a graph G and where $\mathcal{T}(G)$ has v vertices, μ_1 is the highest eigenvalue of $A_{\mathcal{T}(G)}$, and μ_n is the lowest, we have:*

$$\frac{\mu_n}{v-1} \le \frac{deg(V_0)}{v-1} \le \frac{v\mu_1}{v-1} - \mu_n.$$

[2]That is, the set of vertices other than i contained in a loop of length 3 from i to itself has size k_i.

The final theorem we present concerns how $\mathcal{T}(G)$ can be used in understanding colorings of G.

Theorem 2.12. *Let G be a colored graph with chromatic number $\gamma(G)$. Let Γ_2 be the number of vertices colored with the color of second highest multiplicity. Also suppose the distance between V_0 and V_G in $\mathcal{T}(G)$ is d. Then,*

$$\Gamma_2 \geq \frac{d}{\gamma(G) - 1}.$$

3. Edges and Vertices

Dismantling or disconnecting a graph can be achieved by either removing a vertex, or removing an edge. Our intent is to represent this deconstruction in another graph, and thus we define the edge removal graph and the vertex removal graph:

Definition 3.1. For a graph $G = (V, E)$, the edge removal graph $\mathcal{S}(G)$ is defined as:

$$\mathcal{S}(G) := \begin{cases} V_i \in V(\mathcal{S}(G)) \text{ iff } V_i = (V, E'), \text{ where } E' \subset E \\ (V_i, V_j) \in E(\mathcal{S}(G)) \text{ iff } V_i = V_{j \setminus e} \text{ for some } e \in E(G). \end{cases}$$

Recall Definition 2.8 for the definition of the vertex removal graph. See Figure 1 for examples of $\mathcal{S}(G)$ and $\mathcal{T}(G)$.

Remark 3.2. When two subgraphs of G are connected in $\mathcal{T}(G)$ or $\mathcal{S}(G)$, it precisely means that one can remove a vertex or edge from one and get the other. This carries with it an inherent direction (i.e. $V_i \to V_j$ but not $V_j \to V_i$), which therefore means that $\mathcal{T}(G)$ and $\mathcal{S}(G)$ are actually directed graphs. In this paper, we ignore this and nevertheless consider them to be undirected. If we take $\mathcal{T}(G)$ to be directed, we will explicitly denote it as $\tilde{\mathcal{T}}(G)$.

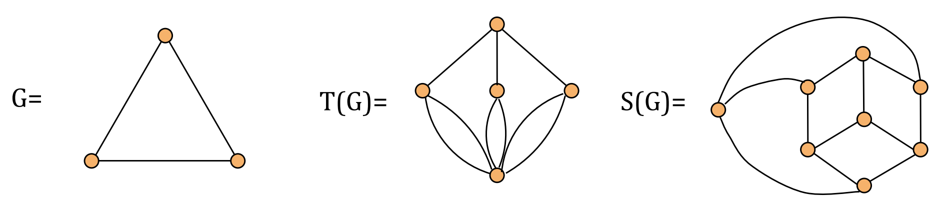

Figure 1. A graph G and its corresponding $\mathcal{T}(G)$ and $\mathcal{S}(G)$. Note the multiple edges between certain vertices in $\mathcal{T}(G)$.

While both edge and vertex removal processes are natural ways to disconnect a graph, this paper will only concern $\mathcal{T}(G)$. Our reasoning for this is immediate from the following definition and proposition.

Definition 3.3. A *graph isomorphism* between two graphs $G = (V_1, E_1)$ and $F = (V_2, E_2)$ is a bijective map $\phi : V_1 \to V_2$ such $\phi(v)$ and $\phi(w)$ are connected in F if and only if v and w are connected in G. If such a map exists between F and G, they are said to be isomorphic (written $G \sim F$). As such, one typically ignores differences between isomorphic graphs.

Proposition 3.4. *For any $G = (V_1, E_1)$ and $F = (V_2, E_2)$ with edge sets of the same size, we have $\mathcal{S}(G) \sim \mathcal{S}(F)$.*

Proof. Since $|E_1| = |E_2| = n$, we can label the edges in both G and F as $\{e_1, ..., e_n\}$. In $\mathcal{S}(G)$, every vertex corresponds to a different subset of its edges (since removing edges commutes). There are 2^n subsets of the edges, and therefore 2^n vertices in both $\mathcal{S}(G)$ and $\mathcal{S}(F)$.

Since both $\mathcal{S}(G)$ and $\mathcal{S}(F)$ have the number of vertices, we can create a bijection ω between the vertices of $\mathcal{S}(G)$ and $\mathcal{S}(F)$ that maps a vertex in $\mathcal{S}(G)$ to the vertex in $\mathcal{S}(F)$ with the same subset of edges. In particular, if V_i and V_j are connected in $\mathcal{S}(G)$, then $\omega(V_i)$ and $\omega(V_j)$ must be connected in $\mathcal{S}(F)$ because they each have the same subset of edges, respectively. The same argument applies in the reverse direction, therefore $(V_i, V_j) \iff (\omega(V_i), \omega(V_j))$ and so $\mathcal{S}(G) \sim \mathcal{S}(F)$.

□

This shows how $\mathcal{S}(G)$ only depends on the number of edges in G, and therefore cannot tell us very much. However, the effect of removing a vertex depends highly on the degree of that vertex as well as other local factors, and thus $\mathcal{T}(G)$ is a much more sophisticated object to merit investigation.

4. Properties of $\mathcal{T}(G)$

The vertex removal graph has various properties which we outline in the two propositions below.

Definition 4.1. A *multigraph* is a graph with duplicate edges.

Definition 4.2. An *isolated edge* is an edge (v_i, v_j) where v_i and v_j aren't in any other edges.

Proposition 4.3. *For any G, the vertex removal graph $\mathcal{T}(G)$ is a multigraph containing at most one duplicate of any of its edges. In particular, an edge connecting $V_i, V_j \in V(\mathcal{T}(G))$ has a duplicate if and only if V_i contains an isolated edge and V_j contains one fewer isolated edges.*

Proof. We will prove the second statement, then the first.

$\Longrightarrow$: If V_i contains an isolated edge, then there are two vertices v_i and v_j that are only connected to each other. Since V_j contains one fewer isolated edge, then we must remove either v_i or v_j in V_i to get to V_j. In particular $V_j = V_{i_{\setminus v_i}}$ and $V_j = V_{i_{\setminus v_j}}$, and thus there are two edges connecting V_i and V_j.

$\Longleftarrow$: In the other direction, suppose V_i and V_j are connected by two edges. Then there must be two distinct vertices in V_i whose removal will yield V_j; call these v_i and v_j. Since removing either must produce the same graph, we see each vertex must be contained in the same set of edges. This is only possible when a single edge connects them.

By similar reasoning, there cannot be three edges connecting some V_i and V_j because that would imply that there are three vertices in V_i that are part of an edge. Therefore, every edge of $\mathcal{T}(G)$ has at most one duplicate. □

Proposition 4.4. *If $G \sim F$, then $\mathcal{T}(G) \sim \mathcal{T}(F)$.*

Proof. If $F \sim G$, then there exists an isomorphism $\phi : V(G) \to V(F)$. Now, define a map $\omega : V(\mathcal{T}(G)) \to V(\mathcal{T}(F))$ such that, if some subgraph $S \in \Phi_G$ is turned into to the graph $S' \in \Phi_F$ by applying ϕ to every vertex in S, then V (the vertex of $\mathcal{T}(G)$ corresponding to S) is mapped to the vertex in $\mathcal{T}(F)$ that corresponds to S'. In particular, ω is bijective because ϕ is bijective.

Now, suppose $V_i, V_j \in V(\mathcal{T}(G))$ are adjacent. Then, by definition, $S_i = S_{j_{\setminus v}}$, where S_i and S_j are subgraphs that correspond to V_i, V_j and v is some vertex in G. Now,

$$\phi(S_i) = \phi(S_{j_{\setminus v}}) = \phi(S_j)_{\setminus \phi(v)}.$$

We see that $\phi(v)$ is some $w \in V(F)$. We also see that $\phi(S_i)$ and $\phi(S_j)$ correspond to $\omega(V_i)$ and $\omega(V_j)$. This means that $\omega(V_i)$ and $\omega(V_j)$ are adjacent. Thus, we have shown that there is a bijective map ω such that, if $(V_i, V_j) \in E(\mathcal{T}(G))$, then $(\omega(V_i), \omega(V_j)) \in E(\mathcal{T}(F))$. This proves the first direction. The second direction can be shown using the same, but reversed, steps. Thus, there is a bijective map $\omega : V(\mathcal{T}(G)) \to V(\mathcal{T}(F))$ such that

$$(V_i, V_j) \in E(\mathcal{T}(G)) \iff (\omega(V_i), \omega(V_j)) \in E(\mathcal{T}(F))$$

and therefore that $\mathcal{T}(G) \sim \mathcal{T}(F)$.

□

Remark 4.5. In the proof of Proposition 4.4, we were careful to differentiate between a vertex of $\mathcal{T}(G)$ and a subgraph of G. For example, we differentiated between the subgraph S_1 and the vertex V_1 that corresponds to that subgraph. While these are essentially the same, we highlighted this difference solely to keep the rigor of constructing a bijective map between vertices (not between subgraphs). In all other cases, we consider a vertex of $\mathcal{T}(G)$ to be both a vertex and a subgraph of G.

This tells us that $\mathcal{T}(G)$ behaves nicely under standard graph invariance and allows us to ignore vertex ordering in a graph and when considering $\mathcal{T}(G)$.

5. Reconstructing G

Here we present an algorithm that allows us to reconstruct G given $\tilde{\mathcal{T}}(G)$ (a directed vertex removal graph).

Theorem 5.1. *For some $\tilde{\mathcal{T}}(G)$, we can reconstruct G through the following steps:*

(1) Locate the vertex with no inward edges on it and the vertex with no outward edges on it. These are V_G and V_0, respectively.
(2) The degree of V_G is the number of vertices in G; label these $\{v_1, ..., v_n\}$. Label the vertices connected to V_G as $\{V_1, .., V_n\}$ so that removing some v_i in G achieves the subgraph V_i.
(3) The number of multi-edges connected to V_0 is the number of edges of G. Label every vertex in $\tilde{\mathcal{T}}(G)$ that is connected to V_0 with a multi edge with $\{P_1, ..., P_q\}$.
(4) For every $P_i \in \{P_1, ..., P_q\}$, find the two vertices $V_i, V_j \in \{V_1, ..., V_n\}$ for which every path from V_0 to V_G containing one of these vertices cannot pass through P_i.
(5) Add (v_i, v_j) to the set of edges of G.

(6) *Every edge and every vertex of G has been identified, and thus G has been determined.*

Before we prove Theorem 5.1, we introduce a few lemmas.

Lemma 5.2. *Given $\tilde{\mathcal{T}}(G)$, the vertex with no inward edges is V_G and the vertex with no outward edges is V_0.*

Proof. The only subgraph in Φ_G that cannot be formed by removing a vertex of G is G itself, therefore, the only vertex in $\tilde{\mathcal{T}}(G)$ with no in-edges is V_G.

Similarly, the only subgraph in Φ_G that cannot form other subgraphs in Φ_G by removing vertices is the empty graph, and so the only vertex in $\tilde{\mathcal{T}}(G)$ that has no outward edges is V_0. □

Lemma 5.3. *Every vertex connected to V_0 with a multi-edge corresponds to a unique edge of G.*

Proof. À la Proposition 4.3, we can say that every vertex connected to V_0 with multiple edges has exactly one more isolated edge than V_0. Since V_0 is empty, it follows that every vertex connected to V_0 with a multi-edge is a subgraph of G containing only one edge of G. We can thus correspond every edge of G to a vertex of $\tilde{\mathcal{T}}(G)$ connected to V_0 with a multi-edge. □

Proof. (of Theorem 5.1)

(1) This is exactly proved by Lemma 5.2
(2) Since the degree of V_G is n, there are n subgraphs that you can attain by removing one vertex from G. This necessarily means that G has exactly n vertices.
(3) This is a direct consequence of Lemma 5.3
(4) Every P_i is a subgraph with one edge. If at any point in a path from V_G to V_0 one removes one of the vertices in this edge, there will be no paths to P_i because this edge has already been disconnected. In particular, if one removes either one of the vertices in this edge as a first step (starting from V_G), there will be no way to get to V_0 through P_i. Therefore, there must be two vertices, v_i and v_j, in G that cannot be removed if one wishes to travel through P_i. Thus, traveling through V_i or V_j (i.e. removes v_i or v_j from G), prohibits a path through P_i.
(5) In identifying these two vertices, we have exactly determined the vertices connected by the edge in P_i, and therefore have found the corresponding edge of G, which is (v_i, v_j).
(6) Continuing this process for every P will generate all edges of G, since every P corresponds directly to every edge of G.

□

It's important to note here the necessity for our proof that $\mathcal{T}(G)$ be directed. The biggest reason for this is from Lemma 5.2, where we depended on knowing the direction of an edge to determine V_0 and V_G.

6. Bounding Degrees in $\mathcal{T}(G)$

For small graphs, the author with A. Misrak used computer programs to generate many examples of $\mathcal{T}(G)$ (see Section 11). A compelling pattern that we noticed was that V_0, the vertex corresponding to the graph with no edges, was always of much higher degree than any other vertices in $\mathcal{T}(G)$. Motivated by this, we present bounds on the degree of V_0 (as stated in Section 2).

Definition 6.1. The *girth* of a graph is the size of the smallest loop. A graph is *triangle free* if it has girth greater than 3.

Theorem 2.9. *For a graph $G = (V, E)$ with the ith vertex having k_i adjacent vertices that are in a triangle,*

$$\sum_{i \in V(G)} \left(2^{deg(i)-k_i} + k_i - 1\right) \leq deg(V_0) \leq \sum_{i \in V(G)} \left(2^{deg(i)} - 1\right).$$

Moreover, when G is triangle-free, the upper and lower bounds are equal and

$$deg(V_0) = \sum_{i \in V(G)} \left(2^{deg(i)} - 1\right).$$

Before we prove Theorem 2.9, we'll introduce a definition and a lemma concerning star graphs.

Definition 6.2. A *star graph* is a graph consisting only of a vertex surrounded by leaves.

Lemma 6.3. *A graph can be completely disconnected by removing only one vertex if and only if it is a star graph.*

Proof.

$\Longrightarrow$: In the first direction, suppose a graph can be completely disconnected by removing v_i. Then every edge must contain v_i, and therefore must be of the form (v_i, w), where w is any other vertex in the graph. Now, we see that there can only be one distinct edge containing each vertex that isn't v_i. This means that every vertex except for v_i is a leaf. Thus, the graph is a vertex surrounded by leaves.

$\Longleftarrow$: The other direction is immediate from our definition of a star graph; any star graph has a vertex that is connected to all other vertices, and all other vertices are only connected to that vertex. Therefore, removing this vertex will disconnect the graph. □

Now we proceed to proving the bounds on V_0.

Proof. (of Theorem 2.9)
By construction, every subgraph connected to V_0 can be disconnected in one step. By the previous lemma, we see that every subgraph connected to V_0 is a star graph. In particular, we can find the degree of V_0 by counting the number of star graphs centered at each vertex in G. Consider the arbitrary vertex v of G shown below.

Without loss of generality, we can assume that every vertex can be isolated as above; that when every edge that does not contain v or one if its neighbors is removed, we are left with a vertex surrounded by leaves and triangles as shown. Suppose there are k_i surrounding vertices in a triangle; then there are $deg(v) - k_i$

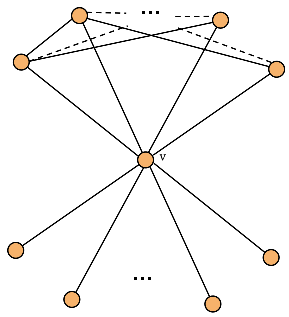

FIGURE 2. An arbitrary vertex v in a graph G with both leaves and triangles.

leaves.

An upper bound for the number of star subgraphs centered on this vertex happens where $k_i = 0$ (i.e. every vertex is a leaf). This means that the number of leaves is $deg(v)$, and that the number of star graphs associated to v is the number of subsets of those leaves, or $2^{deg(v)} - 1$ (we subtract 1 because we do not consider the empty subset of leaves). Thus, if we sum this term over all vertices in G, we obtain the upper bound for the degree of V_0:

$$deg(V_0) \leq \sum_{i \in V(G)} \left(2^{deg(i)} - 1\right).$$

The lower bound can be found in a similar way. From the proof of Theorem 5.1, we know that every edge of G corresponds to a multi-edge containing V_0. In particular, for every edge of G, there will be 2 edges added to the degree of V_0. Thus, a lower bound for $deg(V_0)$ is $2|E(G)|$. Here, we have only counted all star subgraphs of degree 1 (i.e. isolated edges). Using Figure 2, we can improve this bound by counting all star subgraphs of degree 2 or more. The vertex v has $deg(v) - k_i$ leaves, and therefore there are

$$2^{deg(v)-k_i} - (deg(v) - k_i) - 1 = 2^{deg(v)-k_i} - deg(v) + k_i - 1$$

subsets of these leaves of size at least 2. We sum this over all vertices and add $2|E(G)|$ to obtain the lower bound on the degree of V_0:

$$deg(V_0) \geq 2|E(G)| + \sum_{i \in V(G)} \left(2^{deg(i)-k_i} - deg(i) + k_i - 1\right).$$

We can simplify this using $\sum deg(i) = 2|E(G)|$ (see Hararay, pg 14 [3]):

$$deg(V_0) \geq \sum_{i \in V(G)} \left(2^{deg(i)-k_i} + k_i - 1\right).$$

In particular, if G has no triangles, every k_i vanishes, and the lower bound becomes:

$$\begin{aligned} deg(V_0) &\geq \sum_{i \in V(G)} \left(2^{deg(i) - \cancelto{0}{k_i}} + \cancelto{0}{k_i} - 1\right) \\ &\geq \sum_{i \in V(G)} \left(2^{deg(i)} - 1\right) \end{aligned}$$

Thus, the lower and upper bounds are equal in this case.

□

We can apply a similar process to bound the degree of any other vertex in $\mathcal{T}(G)$.

Definition 6.4. The set of *active vertices* in a graph $G = (V, E)$, denoted $\tilde{V}(G)$, is the set of vertices that are contained in at least one edge in E.

Theorem 6.5. *For a graph G and any vertex V_i in $\mathcal{T}(G)$, which corresponds to the subgraph $S_i \in \Phi_G$,*

$$|V(G)| \leq deg(V_i) \leq |\tilde{V}(S_i)| + \sum_{\begin{pmatrix} i \in V(G) \\ i \notin \tilde{V}(S_i) \end{pmatrix}} \left(2^{deg(i)} - 1\right).$$

Proof. Since S_i contains active vertices, we know that there are exactly $|V(S_i)|$ out-paths from V_i. Now we look at all in-paths, ones which create V_i. The only way to create S_i by removing a vertex is to remove a vertex that isn't in the set of active vertices of S_i (i.e. not in $\tilde{V}(S_i)$). In other words, we are looking for the number of star subgraphs centered on vertices not in $\tilde{V}(S_i)$.

A clear lower bound for this is that every vertex only contributes one star. Thus, $C_G(S_i)$ has at least $|V(G)| - |\tilde{V}(S_i)|$ star subgraphs. Adding this to the number of out vertices gives us the lower bound:

$$deg(V_i) \geq |V(G)| - |\tilde{V}(S_i)| + |\tilde{V}(S_i)| = |V(G)|.$$

We can use the same reasoning as in the first proof to find an upper bound. Each vertex will have at most $deg(v)$ leaves, so there are at most $2^{deg(v)} - 1$ star graphs centered on that vertex. Summing over all vertices not in $\tilde{V}(S_i)$ and adding the out-degree then gives the upper bound:

$$\deg(V_i) \leq |\tilde{V}(S_i)| + \sum_{\begin{pmatrix} i \in V(G) \\ i \notin \tilde{V}(S_i) \end{pmatrix}} \left(2^{deg(i)} - 1\right).$$

□

These bounds aren't good enough to allow us to claim that $deg(V_0) \geq deg(V_i)$ for any $\mathcal{T}(G)$. Thus, we only leave this as a conjecture:

Conjecture 6.6. *For a graph G, the maximum degree of $\mathcal{T}(G)$ is $deg(V_0)$.*

7. Special Case

Fortunately, we can assume a special case for which Conjecture 6.6 is true. If we do not allow any triangles in G, then Theorem 2.9 tells us that

$$deg(V_0) = \sum_{i \in V(G)} \left(2^{deg(i)} - 1\right).$$

Splitting the sum over some subgraph $S_i \in \Phi_G$ corresponding to the vertex V_i:

$$deg(V_0) = \sum_{i \in \tilde{V}(S_i)} \left(2^{deg(i)} - 1\right) + \sum_{\begin{pmatrix} i \in V(G) \\ i \notin \tilde{V}(S_i) \end{pmatrix}} \left(2^{deg(i)} - 1\right).$$

If we subtract off the upper bound for the degree of the vertex V_i (corresponding to S_i), given by Theorem 6.5, we have:

$$\begin{aligned} deg(V_0) - deg(V_i) \geq & \sum_{i \in \tilde{V}(S_i)} \left(2^{deg(i)} - 1\right) + \sum_{\begin{pmatrix} i \in V(G) \\ i \notin \tilde{V}(S_i) \end{pmatrix}} \left(2^{deg(i)} - 1\right) - |\tilde{V}(S_i)| \\ & - \sum_{\begin{pmatrix} i \in V(G) \\ i \notin \tilde{V}(S_i) \end{pmatrix}} \left(2^{deg(i)} - 1\right) \\ \geq & \sum_{i \in \tilde{V}(S_i)} \left(2^{deg(i)} - 1\right) - |\tilde{V}(S_i)|. \end{aligned}$$

The smallest that the left sum can be happens when every degree is 1, so $\sum 2^1 - 1 = |\tilde{V}(S_i)|$. Therefore this quantity is always greater than or equal to 0. So,

$$deg(V_0) - deg(V_i) \geq 0,$$

and thus V_0 is the maximal degree.

8. A Cyclomatic Corollary

For a triangle-free graph, not only is V_0 the vertex with the maximum degree, but we have an exact value for this degree. Also, recall Definition 2.4, which says that the cyclomatic number of a graph is the minimum number of edges necessary to remove so that G has no loops and that it can be calculated as $1 + |E(G)| - |V(G)|$. With this, consider the following theorem:

Theorem 2.10. *For a graph G with girth greater than 3 and cyclomatic number $c(G)$, we have:*

$$\gamma(\mathcal{T}(G)) \leq c(G) + \sum_{i \in V(G)} \left(2^{deg(i)} - deg(i)\right).$$

Proof. Brooks shows that, for a graph of maximum degree d, the chromatic number is at most $d + 1$ (pg. 194, [2]). This comes from assigning a color to the vertex of largest degree and all those connected to it. Thus, we start with a bound on the chromatic number of $\mathcal{T}(G)$ of

$$\gamma(\mathcal{T}(G)) \leq 1 + \sum_{i \in V(G)} \left(2^{deg(i)} - 1\right).$$

However, we know that there are $|E(G)|$ multi-edges connected to V_0, and that there are actually only $deg(V_0) - |E(G)|$ adjacent vertices to V_0. Thus,

$$\gamma(\mathcal{T}(G)) \leq 1 - |E(G)| + \sum_{i \in V(G)} \left(2^{deg(i)} - 1\right).$$

We can split up the sum and rewrite:

$$\begin{aligned}\gamma(\mathcal{T}(G)) &\leq 1 - |E(G)| + \sum_{i \in V(G)} 1 + \sum_{i \in V(G)} 2^{deg(i)} + 2|E(G) - 2|E(G)| \\ &\leq 1 + |E(G)| - |V(G)| + \sum_{i \in V(G)} \left(2^{deg(i)} - deg(i)\right).\end{aligned}$$

We recognize the quantity $1 + |E(G)| - |V(G)|$ to be the cyclomatic number of G. Thus,

$$\gamma(\mathcal{T}(G)) \leq c(G) + \sum_{i \in V(G)} \left(2^{deg(i)} - deg(i)\right).$$

□

9. $\mathcal{T}(G)$ and the Connectedness of G

We now pose a new way to measure the connectedness of a graph. Consider the star graph and complete graph, shown below.

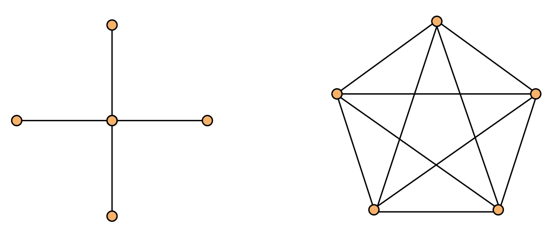

Figure 3. A star graph G (left) and a complete graph F (right), each on 5 vertices

The star graph can be completely disconnected in one step. In fact, every subgraph can be completely disconnected in one step, which means that every vertex in $\mathcal{T}(G)$ is adjacent to V_0. The complete graph, however, cannot be disconnected nearly as easily; very few of the subgraphs in $\mathcal{T}(F)$ can be disconnected in one step, and so V_0 is much less centrally located. In both cases, this property is gauged by the degree of V_0. We define a new quantity that measures this property in a graph:

$$\alpha = \frac{deg(V_0)}{v - 1};$$

where v is the number of vertices in $\mathcal{T}(G)$. This can be thought of as the "normalized degree" of V_0 because it measures the ratio of subgraphs attached to V_0 to the total number of subgraphs.

Remark 9.1. This ratio does't measure the *exact* ratio of subgraphs attached to V_0 to the total number of subgraphs, since there are some subgraphs connected to V_0 with two edges. Instead, it is measuring a *weighted* ratio of subgraphs, because it gives more weight to subgraphs that can be disconnected in more than one way (those with multi-edges to V_0).

The following theorem placed bounds on α using spectral properties of $\mathcal{T}(G)$.

Theorem 2.11. *For a graph G, where $\mathcal{T}(G)$ has v vertices, μ_1 is the highest eigenvalue of $A_{\mathcal{T}(G)}$, and μ_n is the lowest, we have:*

$$\frac{\mu_n}{v-1} \leq \alpha \leq \frac{v\mu_1}{v-1} - \mu_n.$$

Before proving Theorem 2.11, we introduce a definition.

Definition 9.2. For a complex Hermitian matrix M and a non-zero complex vector $\vec{x}$, the *Rayleigh quotient* is:

$$R(M, \vec{x}) := \frac{\vec{x}^\top M \vec{x}}{\vec{x}^\top \vec{x}}.$$

The Courant-Fischer theorem states that, if $\mu_1 \geq \ldots \geq \mu_n$ are the eigenvalues of M, then

$$\mu_n \leq R(M, \vec{x}) \leq \mu_1 \;\; \forall \vec{x} \in \mathbb{C}^n.$$

Proof. (of Theorem 2.11)

Since $A_{\mathcal{T}(G)}$ is a symmetric, $v \times v$ matrix with real entries, it is Hermitian. Therefore, by Courant-Fischer,

$$\frac{\vec{1}^\top A_{\mathcal{T}(G)} \vec{1}}{\vec{1}^\top \vec{1}} \leq \mu_1,$$

where $\vec{1}$ is the vector whose entries are all 1. Right multiplying this by any adjacency matrix yields a vector whose *ith* entry is the degree of the *ith* vertex of the graph. Multiplying this by the transpose of $\vec{1}$ simply sums the degrees of all vertices. Therefore:

$$\frac{\sum_{i \in V(\mathcal{T}(G))} deg(i)}{v} \leq \mu_1.$$

Rewriting:

$$deg(V_0) + \sum_{i \neq V_0} deg(i) \leq v\mu_1$$

$$\frac{deg(V_0)}{v-1} \leq \frac{v\mu_1}{v-1} - \frac{\sum_{i \neq V_0} deg(i)}{v-1}.$$

If we now consider the graph $\mathcal{T}(G)_{\setminus V_0}$, we see that

$$\sum_{i \in V(\mathcal{T}(G)_{\setminus V_0})} deg(i) \leq \sum_{\begin{pmatrix} i \in V(\mathcal{T}(G)) \\ i \neq V_0 \end{pmatrix}} deg(i).$$

And so,

$$\frac{deg(V_0)}{v-1} \leq \frac{v\mu_1}{v-1} - \frac{\sum_{i \in V(\mathcal{T}(G)_{\setminus V_0})} deg(i)}{v-1}. \tag{9.3}$$

Now, we use another Rayleigh quotient: $R(\mathcal{T}(G), \vec{y})$, where $\vec{y}$ is the vector of all one's except for a zero at the entry corresponding to V_0. Again by Courant-Fischer,

$$\mu_n \leq R(\mathcal{T}(G), \vec{y}) = \frac{\vec{y}^\top A_{\mathcal{T}(G)} \vec{y}}{\vec{y}^\top \vec{y}}.$$

Right multiplying $\vec{y}$ by $A_{\mathcal{T}(G)}$ will return a vector whose *ith* entry is the degree of the *ith* vertex in $\mathcal{T}(G)_{\setminus V_0}$. Left multiplying this by $\vec{y}^\top$ sums these degrees. Thus,

$$\mu_n \leq \frac{\sum_{i \in V(\mathcal{T}(G)_{\setminus V_0})} deg(i)}{v-1}.$$

Substituting this into (9.3) leaves:

$$\frac{deg(V_0)}{v-1} \leq \frac{v\mu_1}{v-1} - \mu_n.$$

The lower bound for α can be found by recognizing that $R(\mathcal{T}(G), \vec{1} - \vec{y}) = deg(V_0):$

$$R(\mathcal{T}(G), \vec{1} - \vec{y}) = [0 \ \dots \ 1 \ \dots \ 0] A_{\mathcal{T}(G)} \begin{bmatrix} 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \end{bmatrix} = deg(V_0).$$

Once again by Courant-Fischer:

$$\mu_n \leq R(\mathcal{T}(G), \vec{1} - \vec{y}) = deg(V_0).$$

dividing both sides by $v - 1:$

$$\frac{\mu_n}{v-1} \leq \frac{deg(V_0)}{v-1}.$$

□

Remark 9.4. The number of vertices in $\mathcal{T}(G)$ goes approximately as 2^n, where n the number of vertices in G. Thus, for moderately large graphs G, v is very large, and the upper bound for α becomes nearly $\mu_1 - \mu_n$. So, for larger and larger G, the upper bound for α approaches the size of the spectrum of $\mathcal{T}(G)$ *very* quickly.

10. Distances in $\mathcal{T}(G)$ and Coloring G

Another aspect of $\mathcal{T}(G)$ is how it encodes the possible ways to travel from V_G to V_0 (to completely disconnect a graph). For example, the fewest number of vertices required to completely disconnect G is the lowest value of ℓ for which the $(G, 0)$ entry of ${A_G}^\ell$ is non-zero.

The following theorem addresses the distance between V_G and V_0 and how it relates to coloring G.

Theorem 2.12. *Let G be a colored graph with chromatic number $\gamma(G)$. Let Γ_2 be the number of vertices colored with the color of second highest multiplicity. Also suppose the distance between V_0 and V_G is d. Then,*

$$\Gamma_2 \geq \frac{d}{\gamma(G) - 1}.$$

Proof. Let G be colored with $\gamma(G)$ colors. Now suppose we remove all vertices except for those of a given color. Clearly, G must now be disconnected, since all remaining vertices, by construction, are not connected. Now order the multiplicities of the colors as $\Gamma_1 \geq ... \geq \Gamma_{\gamma(G)}$. Then, removing all but the *kth* color takes at most

$$\sum_{i \neq k}^{\gamma(G)} \Gamma_i$$

steps. This is clearly smallest when $k = 1$. Thus, the fastest way to disconnect G by removing all but one color is at most:

$$\sum_{i=2}^{\gamma(G)} \Gamma_i.$$

This must be bounded below by the actual fewest number of steps required to disconnect G, which is d. Therefore,

$$d \leq \sum_{i=2}^{\gamma(G)} \Gamma_i.$$

This sum is bounded above by the largest value of Γ_i times the interval length, $(\gamma(G) - 1)$. Therefore,

$$d \leq \Gamma_2(\gamma(G) - 1) \Longrightarrow \Gamma_2 \geq \frac{d}{\gamma(G) - 1}.$$

□

Example 10.1.

Suppose you are a test administrator and you are giving a test to a group of students who have already been seated. You don't want any cheating, so you have to make multiple versions of a test and hand them out so that no two adjacent students have the same version. You took a topology course in college, and you were smart to realize that you could represent the seating arrangement in a graph and then find the chromatic number of that graph to find out how many versions of the test you would have to make.

Unfortunately, the copier room is very busy and the most amount of copies of any given test you can make is n. Assuming at least two of the versions of the test get printed n times, how many versions of the test will you need now?

In the terms of our above theorem, we know that $\Gamma_2 = n$, because the two largest multiplicities of the versions will be n. Then, we can write down (or more likely put into a computer) the vertex removal graph $\mathcal{T}(G)$ of the seating arrangement graph G. Once we know the distance between V_G and V_0 is d, then the chromatic number of the graph, γ_2, under this constraint is:

$$n = \Gamma_2 \geq \frac{d}{\gamma_2(G) - 1} \Longrightarrow \gamma_2(G) \geq \frac{d}{n} + 1.$$

We are most interested in the lowest value that $\gamma_2(G)$ must have, so therefore we must make $\frac{d}{n} + 1$ versions of the test (rounded up, if it isn't an integer).

Remark 10.2. It is certainly possible to choose n such that $\frac{d}{n} + 1$ is lower than the actual chromatic number of G; this means we must pick the lowest *allowed* number satisfying the above inequality. In general, the actual new chromatic number is $\max(\gamma(G), \gamma_2(G))$.

11. Computational Methods

We referenced several times the usefulness of computers in this project. Indeed, computing vertex removal graphs is very tedious by hand, especially on graphs exceeding 5 vertices. Appended to this paper is a program written in Python (using a Sage platform) that takes the adjacency matrix of a graph G and returns the adjacency matrix of $\mathcal{T}(G)$. It also has the option of outputting all elements of Φ_G in a list (named K in the code). Due to the exponential nature of $\mathcal{T}(G)$, any program like this will be limited to working with relatively small graphs (~13 vertices) where significant computing power isn't available.

12. Future Directions

There is still a lot of potential for further investigation of $\mathcal{T}(G)$. One compelling topic is the diameter of $\mathcal{T}(G)$ (the closest distance between the furthest two vertices). While we have conjectured some preliminary results about what affects the diameter of $\mathcal{T}(G)$, many examples have shown that it is highly non-trivial.

Another compelling avenue is the question of reconstructing G from an *undirected* $\mathcal{T}(G)$. Recall in Section 4 that we required a directed $\mathcal{T}(G)$, but we haven't yet seen whether this is always necessary. A similar sort of question is whether G is always uniquely associated to $\mathcal{T}(G)$, or if there is some non-isomorphic F such that $\mathcal{T}(F) \sim \mathcal{T}(G)$.

Acknowledgments. My gratitude goes whole-heartedly to my mentors Dr. Farb and Nick Salter, both of whom provided excellent advice and guidance. My thanks also goes to Abel Misrak, who worked jointly with me on this project. Finally, I would like to acknowledge the National Science Foundation and the Leadership Alliance for funding and facilitating this research.

References

[1] Berge, Claude. *The Theory of Graphs*. Courier Dover Publications, pg. 27, 2001
[2] Brooks, R. L.; Tutte, W. T. *On Colouring the Nodes of a Network*. Proceedings of the Cambridge Philosophical Society 37 (2), 1941.
[3] Harary, Frank. *Graph Theory*. Addison-Wesley Publishing, Menlo Park, California, 1969.

Appendix

Program to compute the adjacency matrix of $\mathcal{T}(G)$:

```
def Tau(G):
       n=len(G.rows())
       for k in range(n):
              G[k,k]=k+1
       L=[0]*(2^n)
       K=[0]
       K[0]=G
       L[0]=G
       C=Combinations(G.rows())
       for i in range(n):
              H=copy(G)
              H[i]=0
              H[:,i]=0
              L[i+1]=H
       for i in range(2^n-n-1):
              j=i+n+1
              H=copy(G)
              for k in range(len(C[j])):
                     for m in range(n):
                            if C[j][k]==H[m]:
                                   H[m]=0
              L[j]=H
       for i in range(2^n):
              for m in range(n):
                     if L[i][m]==0:
                            L[i][:,m]=0
       for q in range(2^n):
       for p in range(n):
                     L[q][p,p]=0
       for i in range(2^n):
              c=0;
              for m in range(len(K)):
                     if L[i]==K[m]:
                            c=c+1
              if c==0:
                     K.append(L[i])
       m=len(K)
       T=Matrix(QQ,m,[0]*(m^2))
       for i in range(m):
                     for p in range(n):
                     B=copy(K[i])
                     B[p]=0
                     B[:,p]=0
                     for q in range(m):
                            if B==K[q]:
```

```
                                  T[i,q]=T[i,q]+1
                                  T[q,i]=T[q,i]+1
for i in range(m):
        T[i,i]=0
return T
```