WILLIAM & MARY
CHARTERED 1693

The Minimum Number of Multiplicity 1 Eigenvalues among Real Symmetric Matrices whose Graph is a Tree

Wenxuan "Olivia" Ding
Advisor: Dr. Charles R. Johnson

College of William & Mary

February 19, 2021

Introduction

Let $A = (a_{ij})$ be an $n \times n$ real symmetric matrix. The **graph** of A, denoted $G(A)$, is the simple undirected graph on n vertices with an edge between i and j iff the entry $a_{ij} \neq 0$ (no restriction on the diagonal entries). Given a graph G, we define $\mathcal{S}(G)$ to be the set of all real symmetric matrices A such that $G(A) = G$.

Introduction

Let $A = (a_{ij})$ be an $n \times n$ real symmetric matrix. The **graph** of A, denoted $G(A)$, is the simple undirected graph on n vertices with an edge between i and j iff the entry $a_{ij} \neq 0$ (no restriction on the diagonal entries). Given a graph G, we define $\mathcal{S}(G)$ to be the set of all real symmetric matrices A such that $G(A) = G$.

A **tree**, T, is a minimally connected undirected graph, i.e. a connected acyclic graph on n vertices with $n - 1$ edges.

Introduction

Let $A = (a_{ij})$ be an $n \times n$ real symmetric matrix. The **graph** of A, denoted $G(A)$, is the simple undirected graph on n vertices with an edge between i and j iff the entry $a_{ij} \neq 0$ (no restriction on the diagonal entries). Given a graph G, we define $\mathcal{S}(G)$ to be the set of all real symmetric matrices A such that $G(A) = G$.

A **tree**, T, is a minimally connected undirected graph, i.e. a connected acyclic graph on n vertices with $n - 1$ edges.

$$T = 1 - 2 - 3 - 4$$

Introduction

Let $A = (a_{ij})$ be an $n \times n$ real symmetric matrix. The **graph** of A, denoted $G(A)$, is the simple undirected graph on n vertices with an edge between i and j iff the entry $a_{ij} \neq 0$ (no restriction on the diagonal entries). Given a graph G, we define $\mathcal{S}(G)$ to be the set of all real symmetric matrices A such that $G(A) = G$.

A **tree**, T, is a minimally connected undirected graph, i.e. a connected acyclic graph on n vertices with $n-1$ edges.

$$T = \text{①}-\text{②}-\text{③}-\text{④}$$

$$A = \begin{bmatrix} a_{11} & a_{12} & 0 & 0 \\ a_{12} & a_{22} & a_{23} & 0 \\ 0 & a_{23} & a_{33} & a_{34} \\ 0 & 0 & a_{34} & a_{44} \end{bmatrix}$$

Introduction

Every matrix $A \in M_n(\mathbb{R}) \in \mathcal{S}(T)$ has a **multiplicity list**, which is a partition of n listing multiplicities of the eigenvalues of A. The multiplicities can be summarized in two ways: **ordered** and **unordered**.

Introduction

Every matrix $A \in M_n(\mathbb{R}) \in \mathcal{S}(T)$ has a **multiplicity list**, which is a partition of n listing multiplicities of the eigenvalues of A. The multiplicities can be summarized in two ways: **ordered** and **unordered**.

For example, when $n = 15$, and the eigenvalues are

$$-3, -1, -1, 2, 4, 4, 4, 5, 5, 6, 8, 8, 10, 11, 25$$

An ordered multiplicity list is $(1, 2, 1, 3, 2, 1, 2, 1, 1, 1)$, while an unordered multiplicity list is $(3, 2, 2, 2, 1, 1, 1, 1, 1, 1)$.

Introduction

Every matrix $A \in M_n(\mathbb{R}) \in \mathcal{S}(T)$ has a **multiplicity list**, which is a partition of n listing multiplicities of the eigenvalues of A. The multiplicities can be summarized in two ways: **ordered** and **unordered**.

For example, when $n = 15$, and the eigenvalues are

$$-3, -1, -1, 2, 4, 4, 4, 5, 5, 6, 8, 8, 10, 11, 25$$

An ordered multiplicity list is $(1, 2, 1, 3, 2, 1, 2, 1, 1, 1)$, while an unordered multiplicity list is $(3, 2, 2, 2, 1, 1, 1, 1, 1, 1)$.

The **catalog** of a tree, denoted $\mathcal{L}(T)$, is the collection of all multiplicity lists that occur among the matrices in $\mathcal{S}(T)$.

Introduction

Every matrix $A \in M_n(\mathbb{R}) \in \mathcal{S}(T)$ has a **multiplicity list**, which is a partition of n listing multiplicities of the eigenvalues of A.
The multiplicities can be summarized in two ways: **ordered** and **unordered**.
For example, when $n = 15$, and the eigenvalues are

$$-3, -1, -1, 2, 4, 4, 4, 5, 5, 6, 8, 8, 10, 11, 25$$

An ordered multiplicity list is $(1, 2, 1, 3, 2, 1, 2, 1, 1, 1)$, while an unordered multiplicity list is $(3, 2, 2, 2, 1, 1, 1, 1, 1, 1)$.
The **catalog** of a tree, denoted $\mathcal{L}(T)$, is the collection of all multiplicity lists that occur among the matrices in $\mathcal{S}(T)$.
We define $U(T)$ to be the minimum number of 1's among the lists in $\mathcal{L}(T)$.

Introduction

Theorem

[3] If T is a tree, the largest and smallest eigenvalue of each $A \in \mathcal{S}(T)$ have multiplicity 1. That is, $U(T) \geq 2$.

Introduction

Theorem

[3] If T is a tree, the largest and smallest eigenvalue of each $A \in \mathcal{S}(T)$ have multiplicity 1. That is, $U(T) \geq 2$.

Question

When will the multiplicity lists in $\mathcal{L}(T)$ have $U(T) > 2$?

Introduction

Theorem

[3] If T is a tree, the largest and smallest eigenvalue of each $A \in \mathcal{S}(T)$ have multiplicity 1. That is, $U(T) \geq 2$.

Question

When will the multiplicity lists in $\mathcal{L}(T)$ have $U(T) > 2$?

$U(T)$ can be much greater than 2. For example, $U(P_n) = n$ for the path P_n on n vertices.

Introduction

Theorem

[3] If T is a tree, the largest and smallest eigenvalue of each $A \in \mathcal{S}(T)$ have multiplicity 1. That is, $U(T) \geq 2$.

Question

When will the multiplicity lists in $\mathcal{L}(T)$ have $U(T) > 2$?

$U(T)$ can be much greater than 2. For example, $U(P_n) = n$ for the path P_n on n vertices.

There has been much interest, and progress on determining $\mathcal{L}(T)$ for each tree T. The maximum multiplicity, $M(T)$, is the path cover number $P(T)$ [7], and the minimum number of distinct eigenvalues is at least the diameter $d(T)$ [7]. Similarly, precise information about $U(T)$ would further narrow the possibilities for the catalog $\mathcal{L}(T)$.

Background

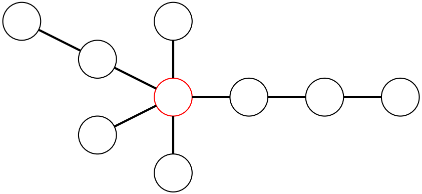
For our purpose, we consider three degree possibilities of a vertex in a tree: a pendent vertex (degree 1), a degree 2 vertex, or **a high degree vertex (HDV)** if its degree is at least 3.

Background

For our purpose, we consider three degree possibilities of a vertex in a tree: a pendent vertex (degree 1), a degree 2 vertex, or **a high degree vertex (HDV)** if its degree is at least 3.
A **generalized star (g-star)** is a tree with at most one HDV; moreover, the HDV, if it exists, is called the **central vertex** of the g-star. A g-star has a number of paths (**arms**) hanging from the central vertex.

Background

For our purpose, we consider three degree possibilities of a vertex in a tree: a pendent vertex (degree 1), a degree 2 vertex, or **a high degree vertex (HDV)** if its degree is at least 3. A **generalized star (g-star)** is a tree with at most one HDV; moreover, the HDV, if it exists, is called the **central vertex** of the g-star. A g-star has a number of paths (**arms**) hanging from the central vertex. For example,

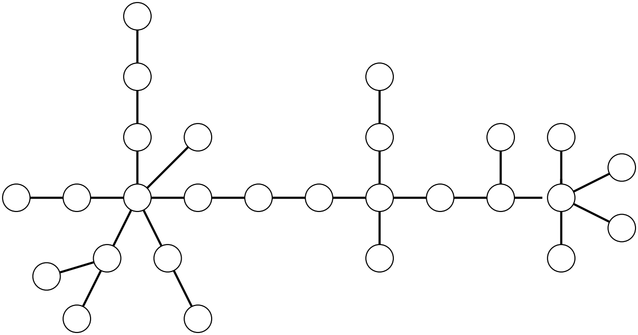


Background

A tree is a **linear tree** if all its HDV's lie on a single induced path. A linear tree with k HDV's is called **k-linear**. And a linear tree can be viewed as the composition of g-stars and connecting paths, i.e. $T = L(T_1, s_1, \ldots, s_{k-1}, T_k)$ [2].

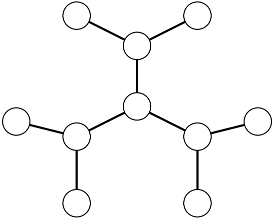
Background

A tree is a **linear tree** if all its HDV's lie on a single induced path. A linear tree with k HDV's is called **k-linear**. And a linear tree can be viewed as the composition of g-stars and connecting paths, i.e. $T = L(T_1, s_1, \ldots, s_{k-1}, T_k)$ [2].
For example, the following tree is a 4-linear tree.



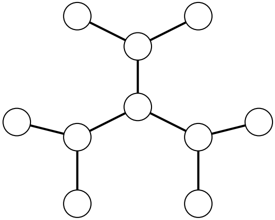
Background

The smallest **nonlinear** tree is on 10 vertices, and by 25 vertices, half of the trees are linear.



Background

The smallest **nonlinear** tree is on 10 vertices, and by 25 vertices, half of the trees are linear.



The **diameter** is defined to be the length of the longest induced path in T, measured in vertices. For example, the diameter for this smallest nonlinear tree is 5.

Background

Auxiliary results for g-stars and their upward multiplicity lists:

WILLIAM & MARY
CHARTERED 1693

Background

Auxiliary results for g-stars and their upward multiplicity lists: Let G be a graph and fix vertex v. Let $A \in \mathcal{S}(G)$. We say that λ is an **upward eigenvalue of** A **at** v if $m_{A(v)}(\lambda) = m_A(\lambda) + 1$. In this case, the multiplicity of this eigenvalue λ is called an **upward multiplicity of** A **at** v, denoted $\hat{q}_i$.

Background

Auxiliary results for g-stars and their upward multiplicity lists:
Let G be a graph and fix vertex v. Let $A \in \mathcal{S}(G)$. We say that λ is an **upward eigenvalue of** A **at** v if $m_{A(v)}(\lambda) = m_A(\lambda) + 1$. In this case, the multiplicity of this eigenvalue λ is called an **upward multiplicity of** A **at** v, denoted $\hat{q}_i$.
Therefore, for a g-star T, we have the following lemma:

Lemma

[1] *Let T be a g-star with the central vertex v. If $A \in \mathcal{S}(T)$ and λ is an eigenvalue of $A(v)$, then $m_{A(v)}(\lambda) = m_A(\lambda) + 1$.*

Background

Auxiliary results for g-stars and their upward multiplicity lists: Let G be a graph and fix vertex v. Let $A \in \mathcal{S}(G)$. We say that λ is an **upward eigenvalue of** A **at** v if $m_{A(v)}(\lambda) = m_A(\lambda) + 1$. In this case, the multiplicity of this eigenvalue λ is called an **upward multiplicity of** A **at** v, denoted $\hat{q}_i$.

Therefore, for a g-star T, we have the following lemma:

Lemma

[1] *Let T be a g-star with the central vertex v. If $A \in \mathcal{S}(T)$ and λ is an eigenvalue of $A(v)$, then $m_{A(v)}(\lambda) = m_A(\lambda) + 1$.*

From this lemma, we can say that the upward eigenvalues of A (including those with multiplicity $\hat{0}$) are exactly the eigenvalues of $A(v)$. The **complete upward multiplicity lists** of a g-star have the form $(1, \hat{q}_1, 1, \hat{q}_2, 1, \cdots, \hat{q}_r, 1)$, in which r upward multiplicities are "bookended" by $r + 1$ non-upward 1's.

Background

Linear superposition principle (LSP):

Background

Linear superposition principle (LSP):

Theorem

[2] [5] *Let $T_1, \cdots, T_k$ be g-stars and $s_1, \cdots, s_{k-1}$ be nonnegative integers. Given $\hat{b}_i$, a complete upward multiplicity list for T_i (relative to the central vertex), $i = 1, \cdots, k$, and $\hat{c}_j$, a list of s non-upward 1's, $j = 1, \cdots, k-1$, construct augmented lists b_i^+, $i = 1, \cdots, k$ and c_j^+ subject to the following:*

1. *all b_i^+ and c_j^+ are the same length;*
2. *each b_i^+ and c_j^+ is obtained from $\hat{b}_i$ ($\hat{c}_j$) by inserting nonupward 0s;*
3. *for each l, the l^{th} element of the augmented lists, denoted $b_{i,l}^+$ and $c_{j,l}^+$, are not all nonupward 0s; and*
4. *for each l, arranging the $b_{i,l}^+$'s and $c_{j,l}^+$'s in the order $b_{1,l}^+, c_{1,l}^+, b_{2,l}^+ c_{2,l}^+, \ldots, b_{k,l}^+$, there is at least one upward multiplicity between any two non-upward ones.*

Then $\sum_{i=1}^{k} b_i^+ + \sum_{j=1}^{k-1} c_j^+$,where the addition is termwise, is a multiplicity list for $LT(T_1, s_1, \ldots, s_{k-1}, T_k)$ generated by the LSP. For any k -linear tree $T = L(T_1, s_1, T_2, s_2, \ldots, s_{k-1}, T_k)$, $\mathcal{L}_o(T)$ is equal to the set of all candidate multiplicity lists generated by the LSP for T.

Background

Example

Let $T = L(T_1, 2, T_2)$ and let $\hat{b} = (1, \hat{2}, 1)$ and $\hat{c} = (1, \hat{1}, 1, \hat{1}, 1, \hat{1}, 1)$ be upward multiplicity lists of T_1 and T_2, respectively. The following are two ways superimposing to get ordered multiplicity lists for T:

$$\begin{array}{cccccccc} 0 & 1 & \hat{2} & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 1 & \hat{1} & 1 & \hat{1} & 1 & \hat{1} & 1 & 0 \\ \hline 1 & 2 & 3 & 2 & 1 & 2 & 1 & 1 \end{array}$$

and

$$\begin{array}{cccccccccc} 0 & 0 & 1 & 0 & \hat{2} & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & \hat{1} & 0 & 1 & \hat{1} & 1 & \hat{1} & 1 \\ \hline 1 & 1 & 1 & 2 & 2 & 1 & 2 & 1 & 1 & 1 \end{array}.$$

Background

Example

Let $T = L(T_1, 2, T_2)$ and let $\hat{b} = (1, \hat{2}, 1)$ and $\hat{c} = (1, \hat{1}, 1, \hat{1}, 1, \hat{1}, 1)$ be upward multiplicity lists of T_1 and T_2, respectively. The following are two ways superimposing to get ordered multiplicity lists for T:

$$\begin{array}{cccccccc}
0 & 1 & \hat{2} & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\
1 & \hat{1} & 1 & \hat{1} & 1 & \hat{1} & 1 & 0 \\
\hline
1 & 2 & 3 & 2 & 1 & 2 & 1 & 1
\end{array}$$

and

$$\begin{array}{cccccccccc}
0 & 0 & 1 & 0 & \hat{2} & 0 & 1 & 0 & 0 & 0 \\
1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & \hat{1} & 0 & 1 & \hat{1} & 1 & \hat{1} & 1 \\
\hline
1 & 1 & 1 & 2 & 2 & 1 & 2 & 1 & 1 & 1
\end{array}.$$

The following superposition is not valid, since it violates the condition **4:**

$$\begin{array}{ccccccc}
0 & 1 & \hat{2} & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 1 & 0 \\
1 & \hat{1} & 1 & \hat{1} & 1 & \hat{1} & 1 \\
\hline
1 & 2 & 3 & 3 & 1 & 2 & 1
\end{array}.$$

Background

Existing results about $U(T)$:

Background

Existing results about $U(T)$:

1 The diameter lower bound:

Theorem

[3] *If T is a tree on n vertices, then $U(T) \geq 2d - n$.*

Background

Existing results about $U(T)$:

1. The diameter lower bound:

Theorem

[3] *If T is a tree on n vertices, then $U(T) \geq 2d - n$.*

2. The degree 2 vertices upper bound:

Theorem

[3] *For a linear tree T, $U(T) \leq 2 + D_2$, where $D_2(T)$ denotes the number of degree 2 vertices in T.*

An Overview of Results

An Overview of Results

I. Incremental changes of $U(T)$ for linear trees

An Overview of Results

I. Incremental changes of $U(T)$ for linear trees

- The change of $U(T)$ is bounded by 1 upon adding a vertex

An Overview of Results

I. Incremental changes of $U(T)$ for linear trees

- The change of $U(T)$ is bounded by 1 upon adding a vertex
- Discussion of how $U(T)$ changes upon different ways of adding a vertex

An Overview of Results

I. Incremental changes of $U(T)$ for linear trees

- The change of $U(T)$ is bounded by 1 upon adding a vertex
- Discussion of how $U(T)$ changes upon different ways of adding a vertex
- A new upper bound and a refined upper bound

An Overview of Results

I. Incremental changes of $U(T)$ for linear trees

- The change of $U(T)$ is bounded by 1 upon adding a vertex
- Discussion of how $U(T)$ changes upon different ways of adding a vertex
- A new upper bound and a refined upper bound

II. A formula for $U(T)$ for 2-linear trees (With M. Ingwersen)

An Overview of Results

I. Incremental changes of $U(T)$ for linear trees

- The change of $U(T)$ is bounded by 1 upon adding a vertex
- Discussion of how $U(T)$ changes upon different ways of adding a vertex
- A new upper bound and a refined upper bound

II. A formula for $U(T)$ for 2-linear trees (With M. Ingwersen)

III. Some preliminary results on $U(T)$ for nonlinear trees

I. Incremental changes of $U(T)$ for linear trees

Given a tree, we can add a vertex and obtain a larger tree via either adding a pendent vertex, in which a new edge and a vertex pendent at an existing vertex are added, or **edge subdivision**, in which a new vertex of degree 2 is positioned along an existing edge. When a tree is linear and the resulting tree upon vertex addition is also linear, we can use the LSP to prove that the change of $U(T)$ is bounded by 1.

WILLIAM & MARY
CHARTERED 1693

I. Incremental changes of $U(T)$ for linear trees

Theorem

Let T be a linear tree, and let T' be a linear tree obtained by adding a vertex to T, through either adding a pendent vertex or edge subdivison. Then $|U(T') - U(T)| \leq 1$.

I. Incremental changes of $U(T)$ for linear trees

Theorem

Let T be a linear tree, and let T' be a linear tree obtained by adding a vertex to T, through either adding a pendent vertex or edge subdivison. Then $|U(T') - U(T)| \leq 1$.

Proof. Given a linear tree $T = L(T_1, s_1, \cdots, s_k, T_{k+1})$, we may add a vertex in the following 6 ways while maintaining the linearity.

1 Add a pendent vertex to the endpoint of an arm of some T_i;

I. Incremental changes of $U(T)$ for linear trees

Theorem

Let T be a linear tree, and let T' be a linear tree obtained by adding a vertex to T, through either adding a pendent vertex or edge subdivison. Then $|U(T') - U(T)| \leq 1$.

Proof. Given a linear tree $T = L(T_1, s_1, \cdots, s_k, T_{k+1})$, we may add a vertex in the following 6 ways while maintaining the linearity.

1 Add a pendent vertex to the endpoint of an arm of some T_i;

2 Add a pendent vertex to the central vertex of some T_i;

I. Incremental changes of $U(T)$ for linear trees

Theorem

Let T be a linear tree, and let T' be a linear tree obtained by adding a vertex to T, through either adding a pendent vertex or edge subdivison. Then $|U(T') - U(T)| \leq 1$.

Proof. Given a linear tree $T = L(T_1, s_1, \cdots, s_k, T_{k+1})$, we may add a vertex in the following 6 ways while maintaining the linearity.

1 Add a pendent vertex to the endpoint of an arm of some T_i;

2 Add a pendent vertex to the central vertex of some T_i;

3 Subdivide an edge on an arm of a g-star (equivalent to **1**);

I. Incremental changes of $U(T)$ for linear trees

Theorem

Let T be a linear tree, and let T' be a linear tree obtained by adding a vertex to T, through either adding a pendent vertex or edge subdivison. Then $|U(T') - U(T)| \leq 1$.

Proof. Given a linear tree $T = L(T_1, s_1, \cdots, s_k, T_{k+1})$, we may add a vertex in the following 6 ways while maintaining the linearity.

1. Add a pendent vertex to the endpoint of an arm of some T_i;
2. Add a pendent vertex to the central vertex of some T_i;
3. Subdivide an edge on an arm of a g-star (equivalent to **1**);
4. Subdivide an edge on a connecting path s_i for some $1 \leq i \leq k$;

I. Incremental changes of $U(T)$ for linear trees

Theorem

Let T be a linear tree, and let T' be a linear tree obtained by adding a vertex to T, through either adding a pendent vertex or edge subdivison. Then $|U(T') - U(T)| \leq 1$.

Proof. Given a linear tree $T = L(T_1, s_1, \cdots, s_k, T_{k+1})$, we may add a vertex in the following 6 ways while maintaining the linearity.

1. Add a pendent vertex to the endpoint of an arm of some T_i;
2. Add a pendent vertex to the central vertex of some T_i;
3. Subdivide an edge on an arm of a g-star (equivalent to **1**);
4. Subdivide an edge on a connecting path s_i for some $1 \leq i \leq k$;
5. Add a pendent vertex to a vertex on a connecting path s_i;

I. Incremental changes of $U(T)$ for linear trees

Theorem

Let T be a linear tree, and let T' be a linear tree obtained by adding a vertex to T, through either adding a pendent vertex or edge subdivison. Then $|U(T') - U(T)| \leq 1$.

Proof. Given a linear tree $T = L(T_1, s_1, \cdots, s_k, T_{k+1})$, we may add a vertex in the following 6 ways while maintaining the linearity.

1 Add a pendent vertex to the endpoint of an arm of some T_i;

2 Add a pendent vertex to the central vertex of some T_i;

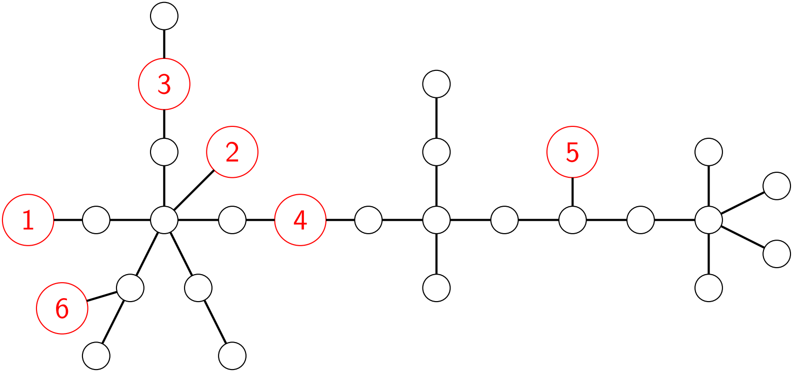
3 Subdivide an edge on an arm of a g-star (equivalent to **1**);

4 Subdivide an edge on a connecting path s_i for some $1 \leq i \leq k$;

5 Add a pendent vertex to a vertex on a connecting path s_i;

6 Add a pendent vertex to a non-pendent vertex on an arm of a peripheral g-star (i.e T_1 or T_{k+1})

I. Incremental changes of $U(T)$ for linear trees



I. Incremental changes of $U(T)$ for linear trees

Empirical evidence from the database [3]:

Table: Changes in $U(T)$ after adding a pendent vertex [4]

	-1	0	$+1$
Isolated	0	0	1
Pendent	221	430	554
Degree 2	936	85	0
HDV	226	594	0
Total	1383	1109	555

Table: Changes in $U(T)$ after edge subdivision

-1	0	$+1$
225	490	909

When a vertex is added in a particular way

When a vertex is added in a particular way, we can determine the exact set of possible values of $U(T') - U(T)$.

When a vertex is added in a particular way

When a vertex is added in a particular way, we can determine the exact set of possible values of $U(T') - U(T)$.

Table: Possible changes of $U(T)$ upon addition of a vertex

Vertex addition		+1	+0	-1
At an HDV		No	Yes	Yes
At a degree 2 vertex		No	Yes	Yes
At a pendent vertex	d increases by 1	Yes	Yes	Yes
	d stays the same	No	Yes	Yes
Edge subdivision		Yes	Yes	Yes

A new bound and a refined bound

A new diameter upper bound:

Corollary

For any linear tree T, $U(T) \leq d(T)$; moreover, $U(T) \leq d(T) - 1$ unless T is a path.

A new bound and a refined bound

A new diameter upper bound:

Corollary

For any linear tree T, $U(T) \leq d(T)$; moreover, $U(T) \leq d(T) - 1$ unless T is a path.

A refined $2 + D_2$ upper bound:

Theorem

For a linear tree $T = L(T_1, s_1, \cdots, s_k, T_{k+1})$, $U(T) = 2 + D_2$ if and only if T is **depth 1** *and either of the following is true:*

(a) $s_i = 0$ *for all* $1 \leq i \leq k$; *or*

(b) $deg_{T_j} = 3$ *for all* $1 < j < k + 1$.

A new bound and a refined bound

A new diameter upper bound:

Corollary

For any linear tree T, $U(T) \leq d(T)$; moreover, $U(T) \leq d(T) - 1$ unless T is a path.

A refined $2 + D_2$ upper bound:

Theorem

For a linear tree $T = L(T_1, s_1, \cdots, s_k, T_{k+1})$, $U(T) = 2 + D_2$ if and only if T is **depth 1** *and either of the following is true:*

(a) $s_i = 0$ for all $1 \leq i \leq k$; or

(b) $deg_{T_j} = 3$ for all $1 < j < k + 1$.

Corollary

If a linear tree T is not depth 1, then $U(T) \leq D_2 + 1$.

II. A formula for $U(T)$ for 2-linear trees

II. A formula for $U(T)$ for 2-linear trees

$U(T)$ has been characterized for 1-linear trees, i.e. g-stars.

Prior result

[1] Let T be a g-star with arm lengths $l_1 \geq \cdots \geq l_a$. Then

$$U(T) = \max\{1 + l_1, 2d(T) - n\}.$$

II. A formula for $U(T)$ for 2-linear trees

$U(T)$ has been characterized for 1-linear trees, i.e. g-stars.

Prior result

[1] Let T be a g-star with arm lengths $l_1 \geq \cdots \geq l_a$. Then

$$U(T) = \max\{1 + l_1, 2d(T) - n\}.$$

Efforts [6] have been made to determine multiplicity lists, hence $U(T)$, for special cases of 2-linear trees, such as double paths.

II. A formula for $U(T)$ for 2-linear trees

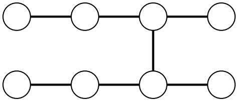
$U(T)$ has been characterized for 1-linear trees, i.e. g-stars.

Prior result

[1] Let T be a g-star with arm lengths $l_1 \geq \cdots \geq l_a$. Then

$$U(T) = \max\{1 + l_1, 2d(T) - n\}.$$

Efforts [6] have been made to determine multiplicity lists, hence $U(T)$, for special cases of 2-linear trees, such as double paths. For example,



II. A formula for $U(T)$ for 2-linear trees

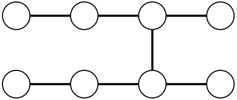
$U(T)$ has been characterized for 1-linear trees, i.e. g-stars.

Prior result

[1] Let T be a g-star with arm lengths $l_1 \geq \cdots \geq l_a$. Then

$$U(T) = \max\{1 + l_1, 2d(T) - n\}.$$

Efforts [6] have been made to determine multiplicity lists, hence $U(T)$, for special cases of 2-linear trees, such as double paths. For example,



Now, moving one step further, we consider 2-linear trees in general.

II. A formula for $U(T)$ for 2-linear trees

Theorem

Let $T = L(T_1, s, T_2)$ be a 2-linear tree. Let n_i be the number of vertices in T_i, $i = 1, 2$. Let $l_1 \geq l_2 \geq \cdots \geq l_a$ (resp., $m_1 \geq m_2 \geq \cdots \geq m_b$) be the length of the arms of T_1 (resp., T_2). We also define $z(T_1) = \max\{0, l_1 - \sum_{i=2}^{i=a} l_i\}$ (resp., $z(T_2) = \max\{0, m_1 - \sum_{j=2}^{j=b} m_j\}$.) Then

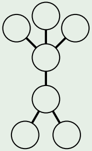
$$U(T) = \max \begin{cases} 2 + z(T_1) + z(T_2) + s & \text{(upward } \hat{0} \text{ bound)} \\ l_1 + 1 - \lfloor \frac{n_2 - z(T_2) - 1}{2} \rfloor & \text{(difference bound for } T_1\text{)} \\ m_1 + 1 - \lfloor \frac{n_1 - z(T_1) - 1}{2} \rfloor & \text{(difference bound for } T_2\text{)} \\ 2d(T) - n_1 - n_2 - s & \text{(diameter bound)} \end{cases}$$

II. A formula for $U(T)$ for 2-linear trees

Example

II. A formula for $U(T)$ for 2-linear trees

Example



Upward $\hat{0}$ bound: **2**

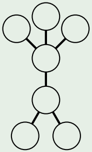
Difference bounds: 1

Diameter bound: 1

$U(T) = 2$

II. A formula for $U(T)$ for 2-linear trees

Example

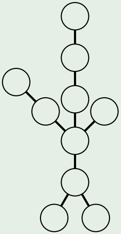


Upward $\hat{0}$ bound: **2**

Difference bounds: 1

Diameter bound: 1

$U(T) = 2$



Upward $\hat{0}$ bound: 2

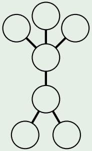
Difference bounds: **3**

Diameter bound: 2

$U(T) = 3$

II. A formula for $U(T)$ for 2-linear trees

Example

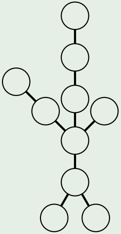


Upward $\hat{0}$ bound: **2**

Difference bounds: 1

Diameter bound: 1

$U(T) = 2$

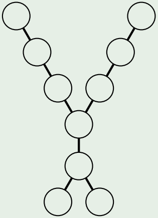


Upward $\hat{0}$ bound: 2

Difference bounds: **3**

Diameter bound: 2

$U(T) = 3$



Upward $\hat{0}$ bound: 2

Difference bounds: 3

Diameter bound: **4**

$U(T) = 4$

III. Nonlinear trees

We know a lot about linear trees, but little is known about nonlinear trees and their multiplicity lists. The LSP cannot be easily extended.

III. Nonlinear trees

We know a lot about linear trees, but little is known about nonlinear trees and their multiplicity lists. The LSP cannot be easily extended.

We classify nonlinear trees in terms of their diameter (≥ 5). Define **cores** of diameter d nonlinear trees to be the minimal nonlinear trees with that diameter.

III. Nonlinear trees

We know a lot about linear trees, but little is known about nonlinear trees and their multiplicity lists. The LSP cannot be easily extended.

We classify nonlinear trees in terms of their diameter (≥ 5). Define **cores** of diameter d nonlinear trees to be the minimal nonlinear trees with that diameter.

The characterization of cores is that, a nonlinear tree T is a core for diameter d nonlinear trees if and only if $n(T) = d + 5$.

III. Nonlinear trees

What are nice about cores:

III. Nonlinear trees

What are nice about cores:

1. For a given diameter, there are finitely many cores.

III. Nonlinear trees

What are nice about cores:

1. For a given diameter, there are finitely many cores.

2. Each core generates an infinite family of nonlinear trees via a sequence of pendent vertex additions while the diameter remains the same.

III. Nonlinear trees

What are nice about cores:

1. For a given diameter, there are finitely many cores.

2. Each core generates an infinite family of nonlinear trees via a sequence of pendent vertex additions while the diameter remains the same.

3. The union of the families of all the cores for d is the set of all nonlinear trees with diameter d.

III. Nonlinear trees

What are nice about cores:

1. For a given diameter, there are finitely many cores.
2. Each core generates an infinite family of nonlinear trees via a sequence of pendent vertex additions while the diameter remains the same.
3. The union of the families of all the cores for d is the set of all nonlinear trees with diameter d.
4. The families of two different cores for some d might overlap with one another.

III. Nonlinear trees

What are nice about cores:

1. For a given diameter, there are finitely many cores.
2. Each core generates an infinite family of nonlinear trees via a sequence of pendent vertex additions while the diameter remains the same.
3. The union of the families of all the cores for d is the set of all nonlinear trees with diameter d.
4. The families of two different cores for some d might overlap with one another.
5. Cores of any diameter d nonlinear trees only contain 4 HDV's.

III. Nonlinear trees

Conjecture

Let T be a core of diameter d nonlinear trees, then

$$U(T) = \begin{cases} 2 & \text{if } d \leq 7 \\ d - 5 & \text{if } d \geq 8 \end{cases}.$$

Moreover, for any T' in the family generated by T, $U(T') \leq U(T)$.

III. Nonlinear trees

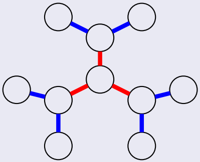
How can we enumerate all the cores of a given diameter d?

III. Nonlinear trees

How can we enumerate all the cores of a given diameter d?

Algorithm

Start with the 10-vertex nonlinear tree, say T. There are two types of edges in T: R(ed) edges that are adjacent to the central vertex and B(lue) edges that are adjacent to some pendent vertex.

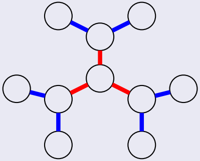


III. Nonlinear trees

How can we enumerate all the cores of a given diameter d?

Algorithm

Start with the 10-vertex nonlinear tree, say T. There are two types of edges in T: R(ed) edges that are adjacent to the central vertex and B(lue) edges that are adjacent to some pendent vertex.



Since the cores are minimal nonlinear trees of diameter d, they are obtained by adding $d - 5$ vertices to T in a way such that the addition of each vertex increases the diameter by 1.

III. Nonlinear trees

Algorithm (Cont.)

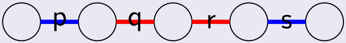
In fact, up to isomorphism, it equates to doing edge subdivision $d-5$ times on one diameter of T. So, generating cores boils down to what we can do on one diameter of T.

III. Nonlinear trees

Algorithm (Cont.)

In fact, up to isomorphism, it equates to doing edge subdivision $d-5$ times on one diameter of T. So, generating cores boils down to what we can do on one diameter of T.

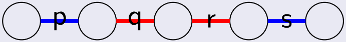
Wlog, assume the diameter is the following 5-path, where p, q, r, and s denote the number of edge subdivisions operated on respective edges.



Algorithm (Cont.)

In fact, up to isomorphism, it equates to doing edge subdivision $d-5$ times on one diameter of T. So, generating cores boils down to what we can do on one diameter of T.

Wlog, assume the diameter is the following 5-path, where p, q, r, and s denote the number of edge subdivisions operated on respective edges.



The set of all the cores of diameter d equals all possible nonnegative integer string partitions of $d-5$, i.e. $p+q+r+s=d-5$, up to isomorphism.

How many cores are there given a diameter d?

III. Nonlinear trees

How many cores are there given a diameter d?

Proposition

Define $c(d)$ to be the number of cores for diameter d nonlinear trees. Then,

$$c(d) = \begin{bmatrix} p_1 & p_2 & p_3 & p_4 & p_5 \end{bmatrix} \begin{bmatrix} 1 & 2 & 4 & 6 & 12 \end{bmatrix}^T$$

where $p_i, 1 \leq i \leq 5$, denotes the number of non-isomorphic partitions of one of the following patterns: $(a, a, a, a), (a, b, b, b), (a, a, b, b), (a, b, b, c)$, and (a, b, c, d).

III. Nonlinear trees

How many cores are there given a diameter d?

Proposition

Define $c(d)$ to be the number of cores for diameter d nonlinear trees. Then,

$$c(d) = \begin{bmatrix} p_1 & p_2 & p_3 & p_4 & p_5 \end{bmatrix} \begin{bmatrix} 1 & 2 & 4 & 6 & 12 \end{bmatrix}^T$$

where $p_i, 1 \leq i \leq 5$, denotes the number of non-isomorphic partitions of one of the following patterns: $(a, a, a, a), (a, b, b, b), (a, a, b, b), (a, b, b, c)$, and (a, b, c, d).

In fact, the generating function of $c(d)$ is the expansion of

$$c(x+5) = \frac{(1+x^2)}{(1-x)^2(1-x^2)^2}.$$

III. Nonlinear trees

d	$c(d)$
5	1
6	2
7	6
8	10
9	19
10	28
11	44
12	60
13	85
14	110
15	146

WILLIAM & MARY
CHARTERED 1693

[1] C.R. Johnson, A. Leal Duarte, and C. Saiago, *Inverse eigenvalue problems and lists of multiplicities of eigenvalues for matrices whose graph is a tree: the case of generalized stars and double generalized stars*, Linear Algebra Appl. **373** (2003), 311–330.

[2] C.R. Johnson, A. Li, and A. Walker, *Ordered multiplicity lists for eigenvalues of symmetric matrices whose graph is a linear tree*, Discrete Mathematics **333** (2014), 39–55.

[3] C.R. Johnson and C.M. Saiago, *Eigenvalues, Multiplicities and Graphs*, Cambridge University Press, Cambridge, 2018.

[4] M. Ingwesen, *2018 Matrix REU report* (2018).

[5] C.R. Johnson and T. Wakhare, *The inverse eigenvalue problem for linear trees*, Submitted, Discrete mathematics.

[6] C.R. Johnson and A. Leal Duarte, *On the possible multiplicities of the eigenvalues of an Hermitian matrix whose graph is a given tree*, Linear Algebra Appl. **348** (2002), 7-21.

[7] C.R. Johnson and A. Leal Duarte, *The maximum multiplicity of an eigenvalue in a matrix whose graph is a tree.*, Linear and Multilinear Algebra **46** (1999), 139-144.