# Characterization Results for the L(2,1)-Labeling Problem on Trees

Christopher A. Wood

September 22, 2012

# Agenda

1. Background and Motivation
   - Problem Statement
   - $L(2,1)$-Labeling
   - Main Result
2. Infinite Classes of Forbidden Subtrees
   - Label Forcing
   - $\Delta(T) = 3$ Algorithm
   - $\Delta(T) = 4$ Algorithm
   - $\Delta(T) \geq 5$ Algorithm
3. Concluding Remarks

# $L(h,k)$-Labeling Problem

The $L(h,k)$ *labeling* of a graph $G$ is a vertex labeling $f : V(G) \to \{0\} \cup \mathbb{Z}^+$ such that

1. $|f(u) - f(v)| \geq h$ for all $uv \in E(G)$,
2. $|f(u) - f(v)| \geq k$ if $d(u,v) = 2$.

# $L(h,k)$-Labeling Span

- The *span of an* $L(h,k)$ *labeling* $f$ on a graph $G$ is the maximum $f(u)$ for all $u \in V(G)$.
- The $L(h,k)$ *span of a graph* $G$, denoted $\lambda_{h,k}(G)$, is the minimum span of all $L(h,k)$ labelings on $G$.

An $L(h,k)$ labeling $f$ on $G$ whose span is equal to the span of $G$ is called a span labeling of $G$.

# $L(2,1)$-Labeling

- The $L(2,1)$-labeling problem on trees is a special case of this class of labeling problems.
- Griggs and Yeh showed that $\lambda_{2,1}(T) \in \{\Delta(T)+1, \Delta(T)+2\}$ for all trees $T$, and further concluded that the problem of recognizing the two classes of trees is NP-hard [2].
- Chang and Kuo have since provided a polynomial-time algorithm that can decide whether or not the label span for a tree $T$ is $(\Delta(T)+1)$ [1].
- The problem of determining the label span of a tree $T$ based solely on its structural properties has been found to be very difficult.

# Definitions

## Critical Labels

*The **critical labels** for an $L(2,1)$-labeling $f$ are the maximum and minimum possible labels for a tree $T$ such that the span of $f$ is equal to $(\Delta(T)+1)$. The maximum label value, $(\Delta(T)+1)$, is the upper critical value and the minimum label value, $0$, is the lower critical value.*

## $\Delta$-Path Segment

*A **$\Delta$-path segment** is defined as a path $P$ between two major vertices $v_i$ and $v_j$ such that all internal vertices are minor vertices.*

# Definitions

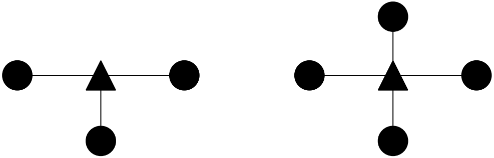
## Forbidden Subtree

*A **forbidden subtree** is defined as a subgraph $T'$ of a tree $T$ such that $\lambda(T') = (\Delta(T) + 2)$.*

## Structural Class of Trees

*A **structural class of trees**, denoted $\langle T \rangle$, is a set of trees for which each tree $T_i$ in the set has the same structure. That is, only major vertices can vary their neighbors by adding or removing pendant vertices with the limitation that the structure of the graph does not change. Furthermore, the $L(2,1)$-labeling span of all trees $T_i$ in the class $\langle T \rangle$ is the same.*

# Structural Class Example



Two trees in the same structural class where $\Delta(T) = 3$ and $\Delta(T) = 4$.

# Main Result

**Theorem**

*There exists an infinite number of forbidden subtree structural classes $\langle T \rangle$ such that $\lambda(T_i) = (\Delta(T_i) + 2)$ for all trees $T_i$ in $\langle T \rangle$.*

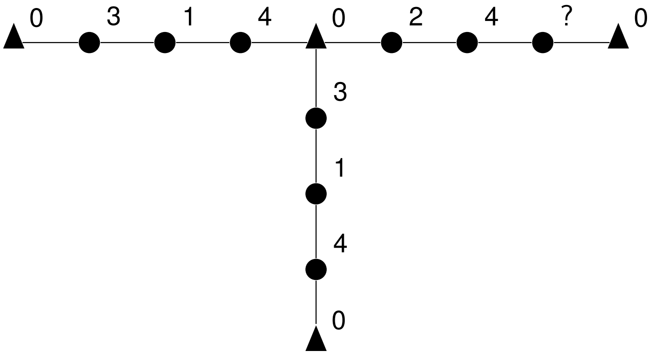
# Label Forcing

We can construct infinitely many forbidden subtrees by *forcing* their label span.

- Choose trees $T$ with known $L(2,1)$ label spans.
- Join copies of $T$ together at specific nodes such that the label possibilities can be controlled.

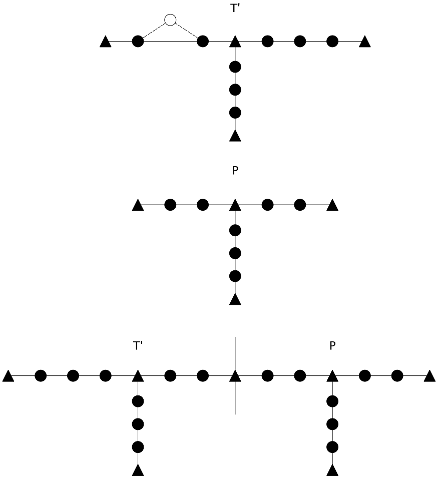
# $\Delta(T) = 3$ Algorithm

The base tree $T$ with a label span of $(\Delta(T) + 2)$.



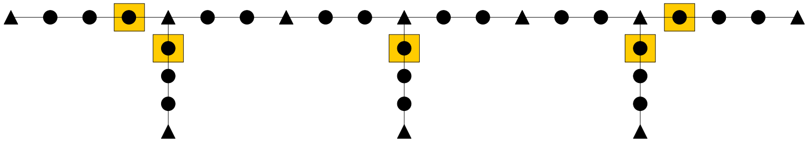
# $\Delta(T) = 3$ Algorithm Steps

- $T^*$ is initialized as a copy of $T$.
- $T'$ is the result of removing one node from $T^*$.
- $P$ is an intermediate tree that is recursively appended to copies of $T^*$.
- A final copy of $T'$ is appended to $T^*$ to force a label span of $(\Delta(T^*) + 2)$.



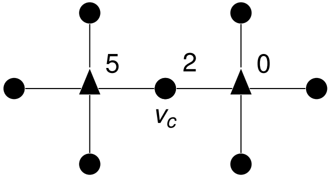
# $\Delta(T) = 3$ Proof Idea

- Highlighted vertices cannot be given a label of 2.
- $\Delta$-path segments of length 4 must be labeled with $\langle 03140 \rangle$, $\langle 04130 \rangle$, or $\langle 04204 \rangle$.
- $\langle 0240 \rangle$, $\langle 0314 \rangle$, $\langle 4204 \rangle$, and $\langle 4130 \rangle$ are the only possible labels for $\Delta$-path segments of length 3 on copies of $P$.
- $P$ additions do not change the label span until the final $T'$ is appended.



# $\Delta(T) = 4$ Algorithm

The base tree $T$ (a joint star) with a label span of $(\Delta(T)+1)$, which we can copy and merge together to force the label span of the resulting graph.

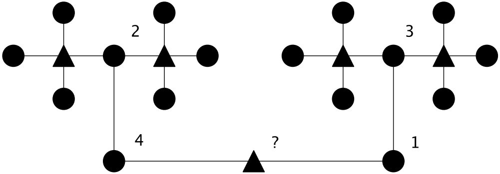


# $\Delta(T) = 4$ Algorithm

ALGORITHM 1: $JOIN_4(T_1, T_2)$

1: Let $v_c^1$ and $v_c^2$ be the centers of two trees $T_1$ and $T_2$.
2: Add a path $P = v_c^1 v_1 v_2 v_3 v_c^2$ to $T^+ = T_1 \cup T_2$, where $v_2$ is a major vertex.
3: **return** $T^+$

# $JOIN_4(T_1, T_2)$ Result

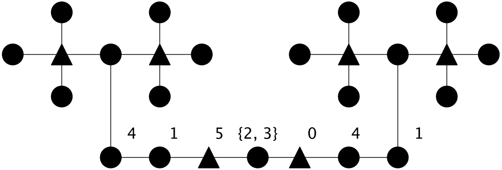


# $\Delta(T) = 4$ Algorithm

ALGORITHM 2: *TRANSFORM*$(T^+)$

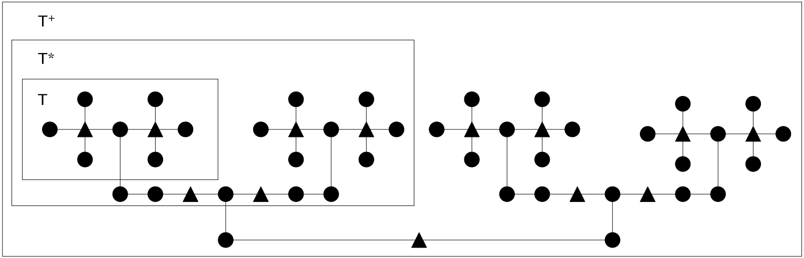
1: Let $P = v_i, v_j, v_k$, the path of vertices between the two joint star centers.
2: Replace $P = v_1 v_2 v_3$ in $G$ with another path $P' = v_i, v_1, v_2, v_j, v_4, v_5, v_k$, where $v_2$ and $v_4$ are major vertices, and let $T^*$ be the resulting tree.
3: **return** $T^*$.

# *TRANSFORM*$(T^{+})$ Result



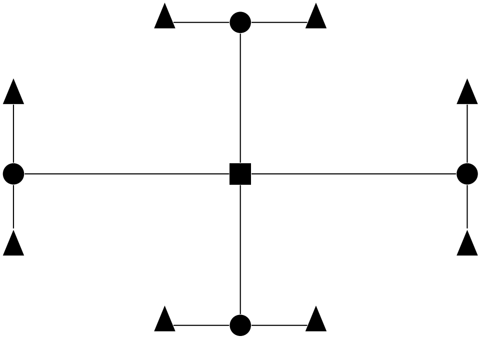
# $\Delta(T) = 4$ Algorithm - Join and Transform

Recursively join copies of $T$ together, transform the resulting tree to form a similar tree with a larger order, and then repeat until iterations are complete.



# $\Delta(T) \geq 5$ Algorithm

An example higher order tree $T$ with $\Delta(T) = 7$.



# Concluding Remarks

- We found infinitely many forbidden subtree classes.
- Finding a complete structural characterization is more difficult than originally thought.
- How feasible is it to find a characterization for smaller order trees?

# References

Gerard J. Chang and David Kuo.
The l(2, 1)-labeling problem on graphs.
*SIAM J. Discrete Math.*, 9(2):309–316, 1996.

Jerrold R. Griggs and Roger K. Yeh.
Labelling graphs with a condition at distance 2.
*SIAM J. Discrete Math.*, 5(4):586–595, 1992.