Recognizing the Overlap Graphs of Leaf-Limited and Particular Trees is NP-Complete

or

What I did in Prague

Jessica Enright, Martin Pergel

Graduate Seminar, University of Alberta, April 30, 2008
Outline

1. Introduction
2. Overlap Graphs of Trees with 3 Leaves
3. Overlap Graphs of Trees with $k$ Leaves
4. Overlap Graphs of Particular Trees
5. Wrap-up
Set Representations of Graphs

Usually, we represent a graph as a list or matrix of adjacencies.

But there are many other ways to represent a graph.

**Set Representation**: each vertex is assigned a set, and two vertices are adjacent if and only if their sets have some relationship.

\[ R = (S, T) \]
Sets Overlap = Vertices Adjacent

{ 2, 3, 4}, {2, 3}, {1, 2}, {5, 3}
The sets we’ve looked at so far have been unrestricted.

All graphs have an overlap representation.

Restricting the nature of the sets restricts the class we can represent.
I’m interested in the overlap graphs of subtrees in a tree.

Subtree overlap representation of G

| a = {2, 3, 4} |
| b = {2, 3}   |
| c = {1, 2}   |
| d = {5, 3}   |
Some related classes very well known

- **Interval Graphs**: Intersection Graphs of Intervals on a Line
- **Chordal Graphs**: Intersection Graphs of Subtrees in a Tree
- **Caterpillar Overlap Graphs**: Overlap Graphs of Subcaterpillars in a Caterpillar
- **Subtree Overlap Graphs**: Overlap Graphs of Subtrees in a Tree
- **Cocomparability Graphs**: Overlap Graphs of Subtrees in a Tree where all Subtrees have a Point in Common
- **Comparability Graphs**: Containment Graphs of Subtrees in a Tree
- **Spider Graphs**: Intersection Graphs of Polygons in a Circle
- **Caterpillar Overlap Graphs**: Overlap Graphs of Subcaterpillars in a Caterpillar
- **Interval Graphs**: Intersection Graphs of Intervals on a Line
- **Permutation Graphs**: Containment Graphs of Intervals on a Line
- **Circle Graphs**: Intersection Graphs of Chords on a Circle

References:
- Golumbic and Scheinerman, 1989
- Pneuli, Even and Lempel, 1971
- Golumbic, 2004
- Cenek, 1998
- Golumbic, 2004
- Pneuli, Even and Lempel, 1971
- Golumbic, 2004
- Cenek, 1998
I went to Prague!
We couldn’t prove hardness of recognition of general SOGs
Overlap graphs of restricted trees is a good - and interesting - start!

Define three problems:

- **$T_3 – OGR$:** Is a given graph the overlap graph of subtrees of a tree with three leaves
- **$k – SOGR$:** For $k \geq 5$, is a given graph the overlap graph of subtrees of a tree with $k$ leaves
- **$T – OGR$:** For an arbitrary tree $T$ with more than two leaves, is a given graph the overlap graph of subtrees of a subdivided $T$

We’ll show that all three of these are NP-complete.
Ideas Behind the Proofs

- From colouring: produce a graph that is representable iff the original is colourable
- Do this by restricting where one can represent the subtrees corresponding to different vertices
- Use sneakiness to block off much of a tree
- Have parts of a tree correspond to colour classes
Tree-Restricted OGs

Jessica Enright, Martin Pergel

Introduction
Overlap Graphs of Trees with 3 Leaves
Overlap Graphs of Trees with $k$ Leaves
Overlap Graphs of Particular Trees
Wrap-up

$T_3$
Rosgen Lemma 3.14

We will use a simplified version:
Graph $G = (V, E)$ is the overlap graph of set family $S$. Consider $v_i, v_j \in V$ such that $(v_i, v_j) \notin E$.

If $s_i | s_j$, then every subtree corresponding to a vertex reachable from $v_i$ avoiding the neighborhood of $v_j$ is also disjoint from $s_j$.

If $s_i \subset s_j$, then every subtree corresponding to a vertex reachable from $v_i$ avoiding the neighborhood of $v_j$ is also contained in $s_j$. 
$k$-Strand graphs
Using Lemma 3.14 (Rosgen), we can tell a few things about any SOR of a $k$-Strand graph.

- Either $t_b \subset t_s$ or $t_s \subset t_b$ - let’s assume the second
- Both $t_s$ and $t_b$ have at least $k$ boundary nodes
Representing a $k$-Strand Graph on a $T_3$

Can only represent for $k = 3$.
The central node of $T_3$ is in $t_s$ and $t_b$. 
3-CON-k-COL

3-Connected: A graph is 3-connected if there is no set of 2 vertices that could be removed to disconnect the graph.

- Problem: Let \( G \) be a 3-connected graph. Can \( G \) be coloured with \( k \)-colours?
- This is NP-Complete

Given a connected graph \( G \), one can always produce a 3-connected graph with the same colouring number as \( G \) in polynomial time.
Constructing a Graph

Goal: Given a graph $G$, produce a graph $G''$ that can be represented on $T_3$ if and only if $G$ is 3-colourable.
Constructing a Graph

Start with 3-connected graph $G = (V, E)$

Graph $G' = (V', E')$ is the disjoint union of four copies of $G$

For later convenience, let's call the vertex sets of the four components of $G'$ $V_a$, $V_b$, $V_c$ and $V_d$
Constructing a Graph

Let’s define four vertex sets:

- \( V_1 = V' \)
- \( V_2 = E' \)
- \( V_3 = \) a set of size \(|V'|\), but disjoint from \( V_1, V_2, V_4 \)
- \( V_4 = \) the vertices of the 3-Strand graph.

Then \( V'' = V_1 \cup V_2 \cup V_3 \cup V_4 \)

Note about \( V_3 \): each member of \( V_3 \) is a *holder* for exactly one member of \( V_1 \)
Constructing a Graph

Let’s also define an edge set

- $E_1 = (v_e, v_x) | v_e \in V_2$ and there exists $v_y \in V'$ such that $v_e = (v_x, v_j) \in E'$
- $E_2 = (v_i, v_j) | v_j \in V_2$ and $v_j$ is the holder of $v_i$
- $E_3 = (v_i, v_j) | v_i \in V_2 \cup V_3$ and $v_j \in V_2 \cup V_3$
- $E_4 = \text{the edges of the 3-Strand graph}$

Then $E'' = E_1 \cup E_2 \cup E_3 \cup E_4$
Representing the Graph

Twig: Let $T$ be a tree. Then the twigs of $G$ are the paths from leaves to just before nodes of degree greater than 2

Nicely Represented: A subset of $V_1$ is nicely represented if they are only on the twigs of the tree, and no two on the same twig are adjacent in $G'$
First Task:

Show that if $G$ is 3-colourable, then $G''$ can be represented on $T_3$
Representing the Graph

Assume $G$ is 3-colourable - then $G'$ is 3-colourable. Let $C_1, C_2, C_3$ be the three colour classes of $G'$.

Assign one twig to each colour class.
Define subtrees $T''_4$ corresponding to vertices in $V_4$: 

![Diagram of subtrees]

Representing the Graph
Representing the Graph

Define subtrees $\mathcal{T}_1''$ corresponding to vertices in $V_1$

Three-vertex subtrees, all disjoint, on the twig corresponding to colour class:
Representing the Graph

We will construct the subtrees corresponding to vertices in $V_2$, $V_3$ in two phases:

- First we will build them - correct adjacency outside $V_2 \cup V_3$
- Then correct adjacency within $V_2 \cup V_3$ - *a clique!*
For each vertex $v_i \in V_2$
- Recall that $v_i$ is an edge in $G'$
- $(v_j, v_k) = v_i$ and $v_j, v_k \in V_1$
- We've already made $t_j, t_k \in \mathcal{T}_1$
- $t_i$ is the path between the inward nodes of $t_j, t_k$
Representing the Graph

For each vertex $v_i \in V_3$

- Recall that there’s a vertex $v_j \in V_1$ associated with $v_i$
- We’ve already made $t_j$
- $t_i$ is the path from the middle node of $t_j$ to central node of $T_3$

$v_k$ is the holder of $v_i$
Representing the Graph

Now we will adjust so that members $\mathcal{T}_2'' \cup \mathcal{T}_3''$ all pairwise overlap.

Sort all members of $\mathcal{T}_2'' \cup \mathcal{T}_3''$ by decreasing size.

Add nodes inside $t_s$ - more to ones later in order.
Now comes the hard part!

Remaining:
- $G''$ is representable on $T_3 \rightarrow G$ is 3-colourable
G” is representable $\rightarrow$ G is 3-colourable

Assume $G''$ is the OG of $T''$, subtrees of $T_3$
G” is representable $\rightarrow$ G is 3-colourable

Sketch of proof:
- Define "nicely represented"
- Recall that in $V_1$ there are four "copies" of vertices in $V$
- Only three copies may be not nicely represented
- At least one copy is nicely represented
- Show that if some vertices in $V_1$ are nicely represented, implies a 3-colouring of those in $G'$
G” is representable $\rightarrow$ G is 3-colourable

Nicely Represented: A subset of $V_1$ is nicely represented if they are only on the twigs of the tree, and no two on the same twig are adjacent in $G'$
We know something about the representation of the 3-Strand.

Both $t_s$ and $t_b$ contain central node.
G” is representable $\rightarrow$ G is 3-colourable

How $V_a, V_b, V_c, V_d$ be not nice?

- Two subtrees on same twig, vertices adjacent in $G'$
- Subtree containing central node $q$
  - Contained in $t_s$
  - Containing $t_s$
Illegal Pair

- Two subtrees on the same twig, vertices adjacent in $G'$

$(t_i, t_j)$ in $E'$, are on the same twig
Illegal Pairs

Some Lemmas:

- In every illegal pair one subtree must be inside $t_s$, the other outside $t_b$
- Cannot represent all vertices from one of $V_a, V_b, V_c, V_d$ as illegal pairs
- All illegal pairs must be on the same twig
- At most two illegal subtrees can have legal neighbors
- At most one illegal pair from each of $V_a, V_b, V_c, V_d$
- At most one of $V_a, V_b, V_c, V_d$ has an illegal pair

There is only one illegal pair.
Subtrees from $V_1$ containing $q$

There are at most two:
- One contained in $t_s$
- One containing $t_b$
G” is representable $\rightarrow$ G is 3-colourable

Only:
- At most one of $V_a, V_b, V_c, V_d$ has an illegal pair
- At most two of $V_a, V_b, V_c, V_d$ have a subtree not contained in a twig

If one of $V_a, V_b, V_c, V_d$ has neither of above, it is nicely represented.
Therefore, one of $V_a, V_b, V_c, V_d$ is nicely represented.
G” is representable $\rightarrow$ G is 3-colourable

Let $V_i$ be a member of $\{V_a, V_b, V_c, V_d\}$ that is nicely represented.

Let $C_1, C_2, C_3$ be the sets of vertices of $V_i$ represented on each of the three twigs of $T_3$.

$C_1, C_2, C_3$ are colour classes for $G'[V_i]$, isomorphic to $G$. 
Good News!

On to overlap graphs of trees with $k \geq 5$ leaves!

Very similar to $T_3$ proof.
Problem

Reminder!

\( k - SOGR: \) For \( k \geq 5 \), is a given graph the overlap graph of subtrees of a tree with \( k \) leaves
Constructing a Graph

Goal: Given a graph $G$, produce a graph $G''$ that can be represented on a tree with $k$ leaves if and only if $G$ is $k$-colourable.
Modified $k$-Strand graphs
Representing a Modified $k$-Strand Graph on a Tree of $k$ Leaves

We can prove a few things:

- $t_{s2} \subset t_{b2}$
- $t_s \mid t_{b2}$
- $t_{b2} \subset t_b$
Representing a Modified $k$-Strand Graph on a Tree of $k$ Leaves

If a Mod-$k$-Strand Graph is represented on a tree $T$ with $k$ leaves:

- There is at least one node of $T$ with degree 3.
- One node of degree 3 is contained in $t_s$.
- All other nodes of $t_s$ are degree less than 3.
- All other nodes of degree greater than 2 are contained in $t_{s2}$ and $t_{b2}$.
- All nodes of degree greater than 2 are contained in $t_b$. 

![Diagram showing $t_{s2}$, $t_{b2}$, and $t_s$ nodes containing $t_b$]
Goal: Given a graph $G$, produce a graph $G''$ that can be represented on a tree with $k$ leaves if and only if $G$ is $k$-colourable.
Constructing a Graph

Start with 3-connected graph $G = (V, E)$

Graph $G' = (V', E')$ is the disjoint union of four copies of $G$. For later convenience, let's call the vertex sets of the four components of $G'$ $V_a$, $V_b$, $V_c$ and $V_d$. 
Constructing a Graph

Let’s define four vertex sets:

- $V_1 = V'$
- $V_2 = E'$
- $V_3 = \text{a set of size } |V'|, \text{ but disjoint from } V_1, V_2, V_4$
- $V_4 = \text{the vertices of the Mod-k-Strand graph.}$

Then $V'' = V_1 \cup V_2 \cup V_3 \cup V_4$
Let’s also define an edge set

- \( E_1 = (v_e, v_x) | v_e \in V_2 \) and there exists \( v_y \in V' \) such that \( v_e = (v_x, v_j) \in E' \)
- \( E_2 = (v_i, v_j) | v_j \in V_2 \) and \( v_i \) is the holder of \( v_j \)
- \( E_3 = (v_i, v_j) | v_i \in V_2 \cup V_3 \) and \( v_i \in V_2 \cup V_3 \)
- \( E_4 = \) the edges of the 3-Strand graph

Then \( E'' = E_1 \cup E_2 \cup E_3 \cup E_4 \)
Representing the Graph

Assume $G$ is $k$-colourable - then $G'$ is $k$-colourable.
Let $C_1...C_k$ be the colour classes of $G'$.

Assign one twig to each colour class.
Representing the Graph

Build subtrees effectively as before, but some members of $\mathcal{T}_2'' \cup \mathcal{T}_3''$ contain extra nodes of $T''$. 

\[
\begin{align*}
C_1 & \quad C_2 \\
C_3 & \quad C_4 \\
C_5 & \quad \vdots \\
& \quad C_k
\end{align*}
\]
Remaining:

- \( G'' \) is \( k \) - SOGR \( \rightarrow \) \( G \) is k-colourable
G” is representable $\rightarrow$ G is k-colourable

Assume $G''$ is the OG of $T''$, subtrees of a tree with $k$ leaves
Sketch of proof:

- Recall that in $V_1$ there are four "copies" of vertices in $V$
- Only three copies may be not nicely represented
- At least one copy is nicely represented
- Show that if vertices in $V_1$ are nicely represented, implies a $k$-colouring for those in $G'$
Only:

- At most one of $V_a, V_b, V_c, V_d$ has an illegal pair
- At most two of $V_a, V_b, V_c, V_d$ have a subtree not contained in a twig

If one of $V_a, V_b, V_c, V_d$ has neither of above, it is nicely represented.

Therefore, one of $V_a, V_b, V_c, V_d$ is nicely represented.
G” is representable → G is 3-colourable

Let \( V_i \in \{ V_a, V_b, V_c, V_d \} \) be nicely represented.

Let \( C_1, C_2, C_3 \) be the sets of vertices of \( V_i \) represented on each of the three twigs of \( T_3 \).

\( C_1 \ldots C_k \) are colour classes for \( G'[V_i] \), isomorphic to \( G \).
The proof is almost exactly the same as the previous one!

Except we base the modified Strand Graph on the Tree.
Let $T$ be the tree we want to represent things on.

- If $T$ has only one node of degree greater than two - analogous to the $T_3$
- If $T$ has more than one node of degree greater than two:
  - Find the smallest degree node such that:
    - It has degree $\geq 3$
    - Only removing only one of its neighbors disconnects it from all other nodes of degree $\geq 3$
  - Let the degree of that node be $\delta_m$
Modified Strand Graph
Then we make the Modified Strand Graph with \((\delta_m)\) and (number leaves of \(T - \delta_m + 1\)) strands:

This leaves us with a "uniqueish" distribution of the nodes of degree \(\geq 3\) between \(t_s\) and \(t_{b2}\)
Representing the Modified Strand Graph
From there it’s basically the same as the other two proofs.
Hopes and Dreams

- Can I combine these proofs?
- SOG recognition in general
- Complicacy measures of a SOG
Thanks!

- Martin Pergel, Jan Kratochvil and KAM
- Lorna Stewart
- NSERC
- iCORE