Uniqueness in tree-depth labelings of graphs

Michael D. Barrus

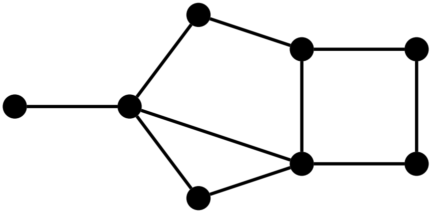
Department of Mathematics
University of Rhode Island

MAA Northeastern Section Spring Meeting
Norwich University • June 3, 2017

Joint work with John Sinkovic (University of Waterloo)

Tree-depth

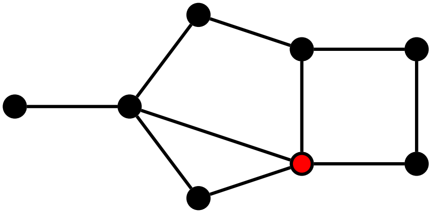
(aka (vertex) ranking number, ordered coloring number, ...)



Tree-depth td(G)**:** The minimum number of steps needed to delete all of G, where in each step at most one vertex is deleted from each connected component.

Tree-depth

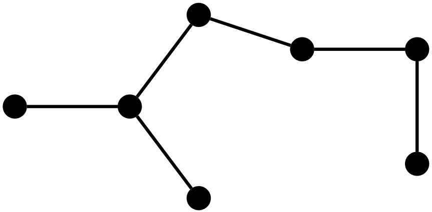
(aka (vertex) ranking number, ordered coloring number, ...)



Tree-depth td(G)**:** The minimum number of steps needed to delete all of G, where in each step at most one vertex is deleted from each connected component.

Tree-depth

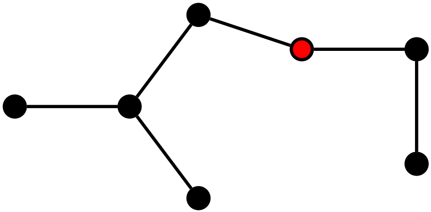
(aka (vertex) ranking number, ordered coloring number, ...)



Tree-depth td(G)**:** The minimum number of steps needed to delete all of G, where in each step at most one vertex is deleted from each connected component.

Tree-depth

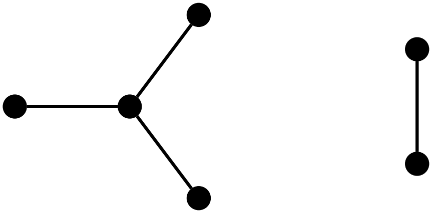
(aka (vertex) ranking number, ordered coloring number, ...)



Tree-depth td(G)**:** The minimum number of steps needed to delete all of G, where in each step at most one vertex is deleted from each connected component.

Tree-depth

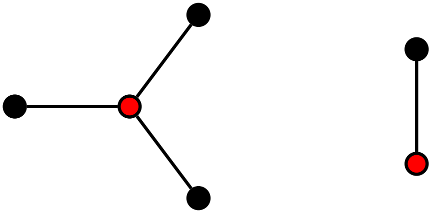
(aka (vertex) ranking number, ordered coloring number, ...)



Tree-depth td(G)**:** The minimum number of steps needed to delete all of G, where in each step at most one vertex is deleted from each connected component.

Tree-depth

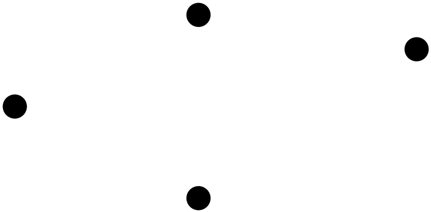
(aka (vertex) ranking number, ordered coloring number, ...)



Tree-depth td(G)**:** The minimum number of steps needed to delete all of G, where in each step at most one vertex is deleted from each connected component.

Tree-depth

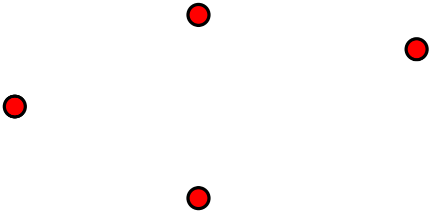
(aka (vertex) ranking number, ordered coloring number, ...)



Tree-depth td(G)**:** The minimum number of steps needed to delete all of G, where in each step at most one vertex is deleted from each connected component.

Tree-depth

(aka (vertex) ranking number, ordered coloring number, ...)



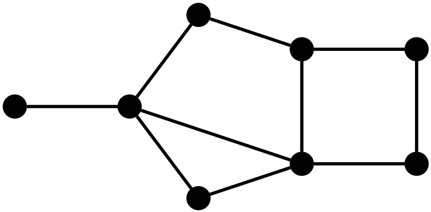
Tree-depth td(G)**:** The minimum number of steps needed to delete all of G, where in each step at most one vertex is deleted from each connected component.

Tree-depth
(aka (vertex) ranking number, ordered coloring number, ...)

Tree-depth td(G)**:** The minimum number of steps needed to delete all of G, where in each step at most one vertex is deleted from each connected component.

Tree-depth

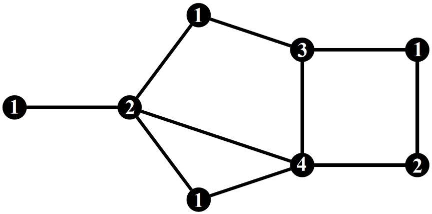
(aka (vertex) ranking number, ordered coloring number, ...)



Tree-depth td(G)**:** The minimum number of steps needed to delete all of G, where in each step at most one vertex is deleted from each connected component. (Here, td(G) $= 4$)

Tree-depth

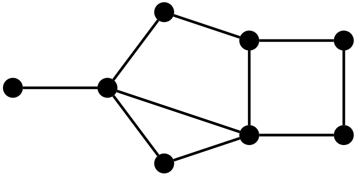
(aka (vertex) ranking number, ordered coloring number, ...)



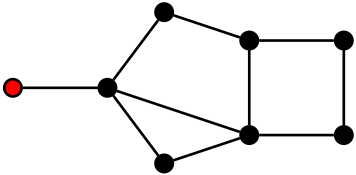
Tree-depth td(G)**:** The minimum number of steps needed to delete all of G, where in each step at most one vertex is deleted from each connected component. (Here, td(G) $= 4$)

Equivalently, the smallest number of labels needed in a labeling where every path with equal endpoints also has a higher label.

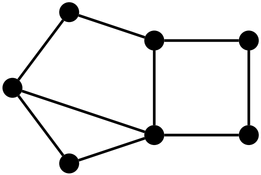
Tree-depth and criticality (with respect to minors)



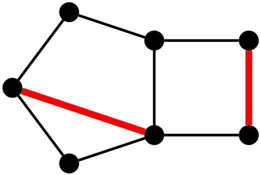
Tree-depth and criticality (with respect to minors)



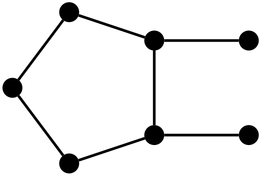
Tree-depth and criticality (with respect to minors)



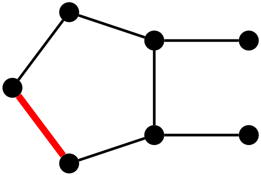
Tree-depth and criticality (with respect to minors)



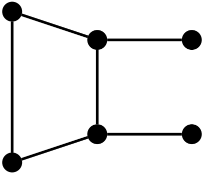
Tree-depth and criticality (with respect to minors)



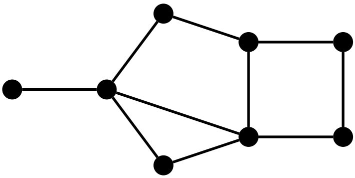
Tree-depth and criticality (with respect to minors)

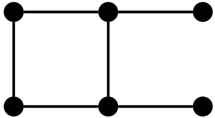


Tree-depth and criticality (with respect to minors)



Tree-depth and criticality (with respect to minors)

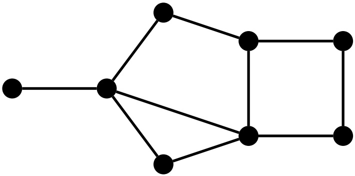


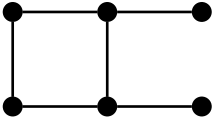


Theorem

If G contains H as a minor, then $\mathrm{td}(G) \geq \mathrm{td}(H)$.

Tree-depth and criticality (with respect to minors)





Theorem

If G contains H as a minor, then $\mathrm{td}(G) \geq \mathrm{td}(H)$.

Idea: A graph is **critical** if every proper minor has lower tree-depth.

Critical minors determine the tree-depth.

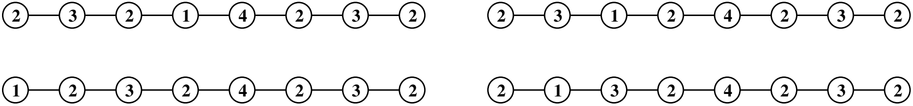
An observation and a new concept

In a **critical** graph, deleting any vertex or edge, or contracting any edge, lowers the tree-depth.

An observation and a new concept

In a **critical** graph, deleting any vertex or edge, or contracting any edge, lowers the tree-depth.

A graph is **1-unique** if for every vertex *v* there exists an optimal tree-depth labeling where *v* is the only vertex assigned the label 1.



Why 1-uniqueness? It's like criticality!

Just like critical graphs, 1-unique graphs are tree-depth-critical with respect to

- **vertex deletions**,

Why 1-uniqueness? It's like criticality!

Just like critical graphs, 1-unique graphs are tree-depth-critical with respect to

- **vertex deletions**,
- **edge contractions**, and

Why 1-uniqueness? It's like criticality!

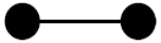
Just like critical graphs, 1-unique graphs are tree-depth-critical with respect to

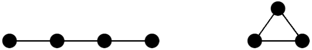
- **vertex deletions**,
- **edge contractions**, and
- **deletions of edge cuts**.

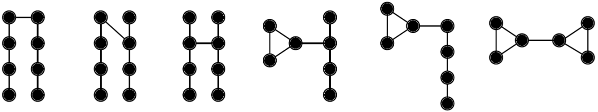
Why 1-uniqueness? It's like criticality!

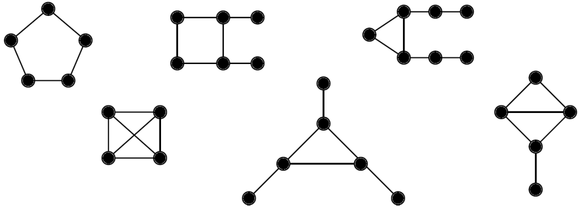
Critical graphs with small tree-depth: Dvořák–Giannopoulou–Thilikos, '09, '12

1: 

2: 

3: 

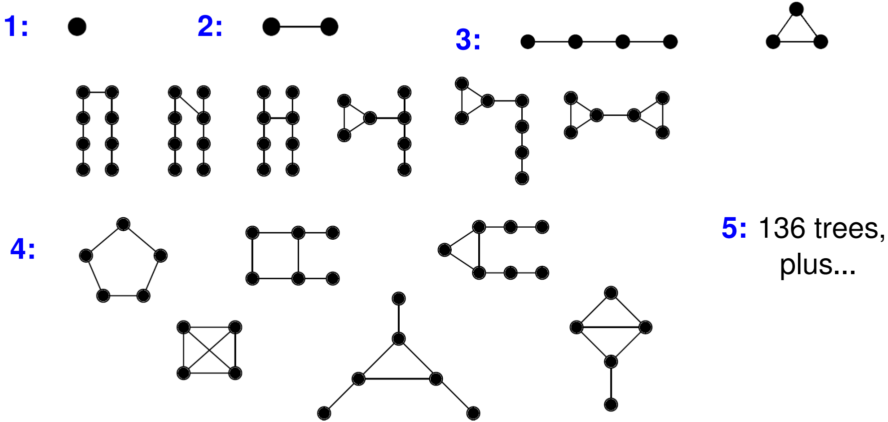


4: 

5: 136 trees,
plus...

Why 1-uniqueness? It's like criticality!

Critical graphs with small tree-depth: Dvořák–Giannopoulou–Thilikos, '09, '12

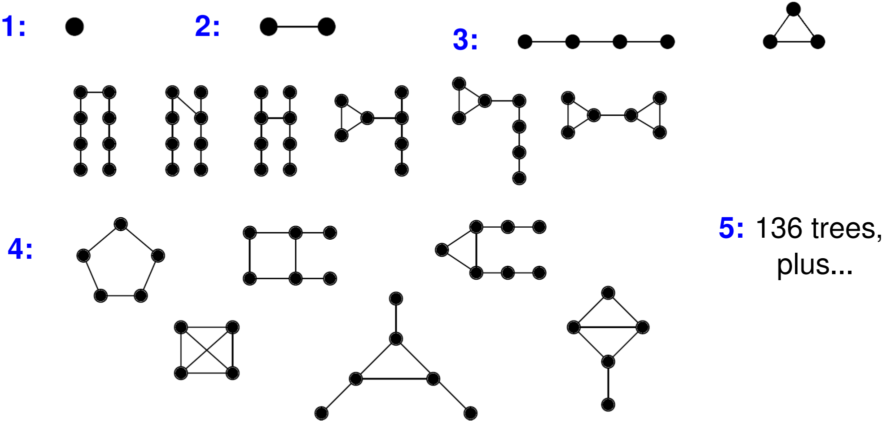


Conjecture: (B, Sinkovic, 2016)

All critical graphs are 1-unique!

Why 1-uniqueness? It's like criticality!

Critical graphs with small tree-depth: Dvořák–Giannopoulou–Thilikos, '09, '12

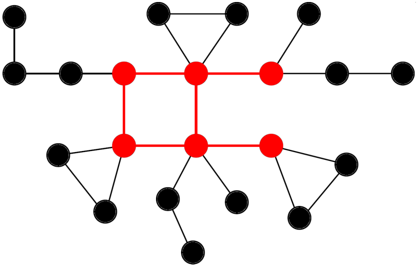


(Ultimately hasty) Conjecture: (B, Sinkovic, 2016)

All critical graphs are 1-unique!

Why 1-uniqueness? It's useful!

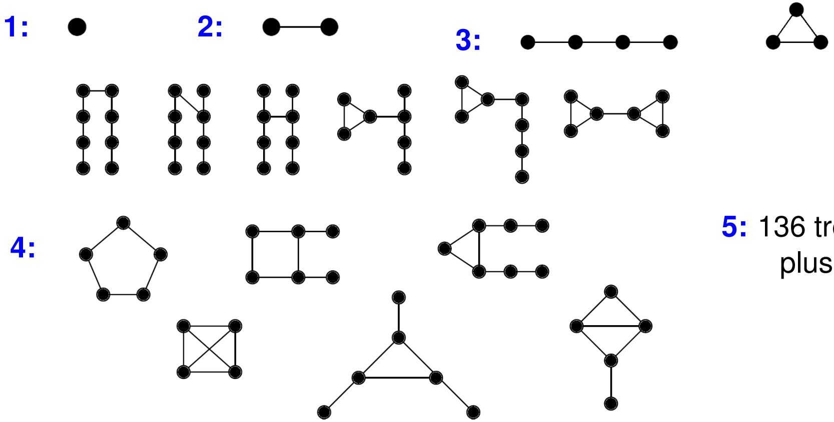
Hang k-critical "appendages" off every vertex of an ℓ-critical graph...



Theorem (B, Sinkovic, 2016)

Graphs constructed in this way are $(k + \ell - 1)$-critical if the appendages are 1-unique.

Why 1-uniqueness? It's useful!



5: 136 trees,
plus...

B, Sinkovic, 2016

Conjecture: If G is k-critical, then $\Delta(G) \leq k - 1$

Proposition: If G is 1-unique and $\text{td}(G) = k$, then $\Delta(G) \leq k - 1$.

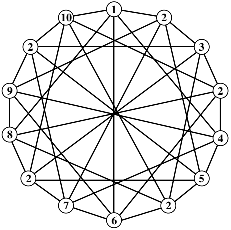
Back to that conjecture

(B, Sinkovic, 2016, 2017+)

Is every critical graph 1-unique?

Yes for

- every k-critical graph for $k \in \{1, 2, 3, 4\}$,
- every critical tree,
- every critical cycle,
- every Andrasfai graph,
- ...



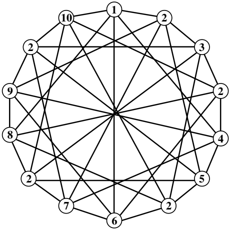
Back to that conjecture

(B, Sinkovic, 2016, 2017+)

Is every critical graph 1-unique?

Yes for

- every k-critical graph for $k \in \{1, 2, 3, 4\}$,
- every critical tree,
- every critical cycle,
- every Andrasfai graph,
- ...



Theorem

*If G is an n-vertex critical graph **and** $\mathrm{td}(G) \geq n-1$, then G is 1-unique.*

Resolution

Conjecture

For any k, if G is k-critical, then G is 1-unique.

Known true for

$$k = 1, 2, 3, 4, \qquad\qquad n - 1, n$$

Resolution

Conjecture

For any k, if G is k-critical, then G is 1-unique.

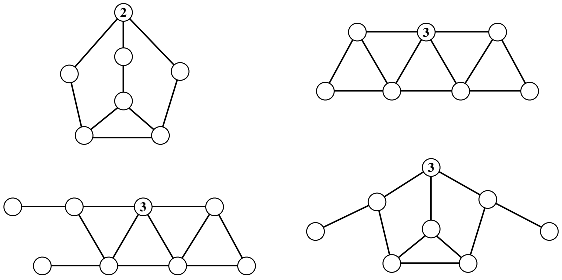
Known true for

$$k = 1, 2, 3, 4, \qquad\qquad\qquad n-1, n$$

False for

$$k = \qquad\qquad 5, 6, \ldots, n-2$$

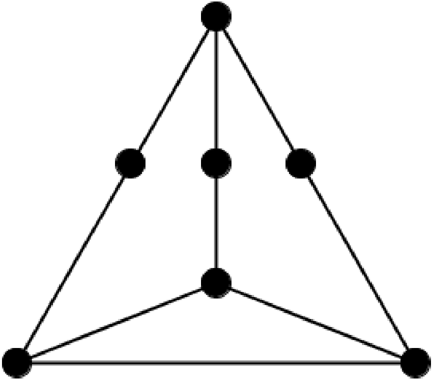
Finding counterexamples



Computer search using SageMath's graph database and functions

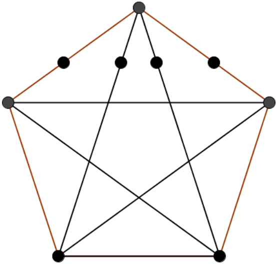
- Iterate through proper colorings of a graph.
- Identify colorings with a unique lowest label; identify 1-unique (or nearly 1-unique) graphs.
- Use 1-uniqueness to produce candidates for criticality testing.

Counterexample



A 7-vertex critical graph G with $\text{td}(G) = 5 = n(G) - 2$ that is not 1-unique!

Counterexamples



For $t \geq 2$, form H_t by subdividing all edges incident with a vertex of K_{t+2}.

Here, $\text{td}(H_t) = n(H_t) - t$.

The graph H_t is critical, but in **no** optimal tree-depth labeling can the vertex at the subdivided edges' center receive the unique 1.

Resolution

Conjecture

For any k, if G is k-critical, then G is 1-unique.

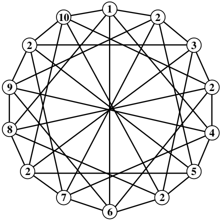
Known true for

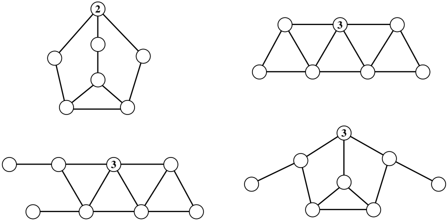
$k = 1, 2, 3, 4, \qquad n-1, n$

False for

$k = \qquad 5, 6, \ldots, n-2$

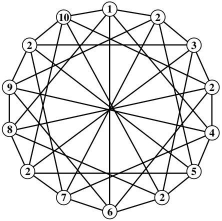
Remaining questions

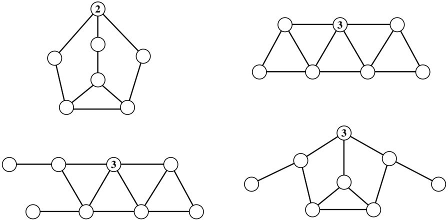




- Is “critical implies 1-unique” true for large special classes?

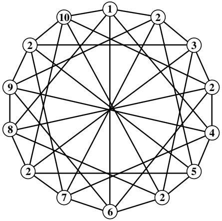
Remaining questions

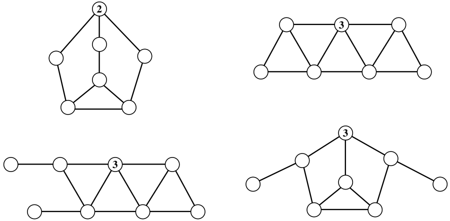




- Is “critical implies 1-unique” true for large special classes?
- Empirically, it appears that when a k-critical non-1-unique graph has a unique “problem vertex,” deleting it yields a $(k-1)$-critical graph. Is this always the case?

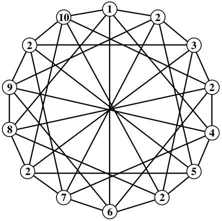
Remaining questions

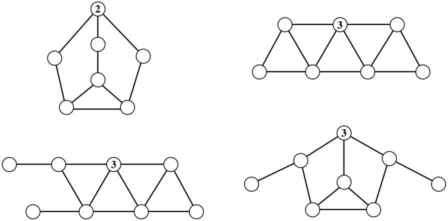




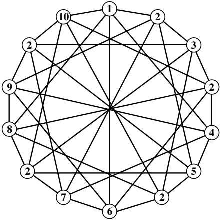
- Is “critical implies 1-unique” true for large special classes?
- Empirically, it appears that when a k-critical non-1-unique graph has a unique “problem vertex,” deleting it yields a $(k-1)$-critical graph. Is this always the case?
- What fraction of critical graphs are 1-unique?

Remaining questions





- Is "critical implies 1-unique" true for large special classes?
- Empirically, it appears that when a k-critical non-1-unique graph has a unique "problem vertex," deleting it yields a $(k-1)$-critical graph. Is this always the case?
- What fraction of critical graphs are 1-unique?
- How else, instead, can we prove/disprove that critical graphs satisfy $$\Delta(G) \leq \text{td}(G) - 1?$$



Thank you!