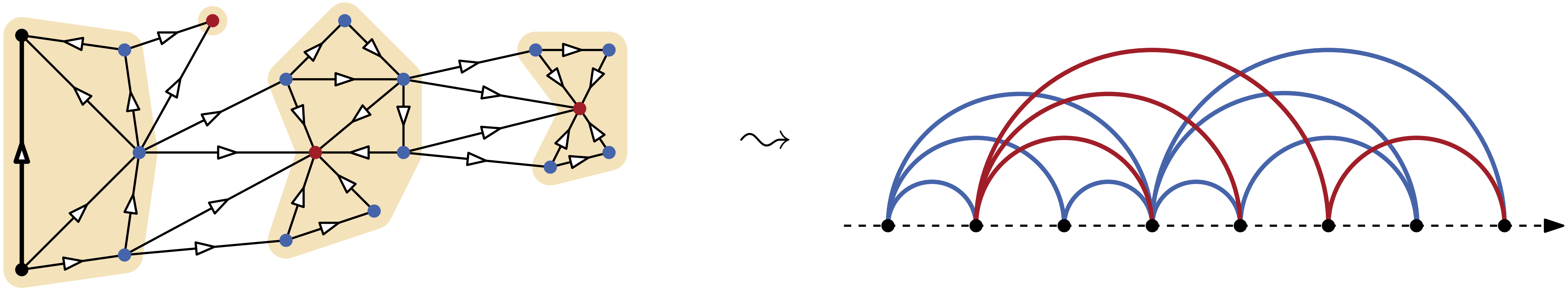
Directed Acyclic Outerplanar Graphs Have Constant Stack Number

FOCS 2023 · 9th of November 2023

Paul Jungeblut, Laura Merker, Torsten Ueckerdt

Karlsruhe Institute of Technology, Germany

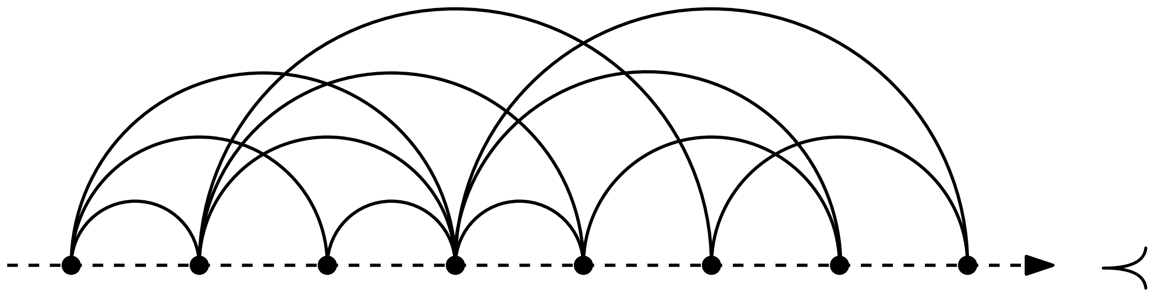


Stack Number: $\mathrm{sn}(G)$

Input: Graph G

Want:
- linear ordering $\prec$ of vertices
- k-coloring of edges, s.t:
  same color $\rightsquigarrow$ crossing-free

Stack Number: $\mathrm{sn}(G) := \min_{\prec} k$

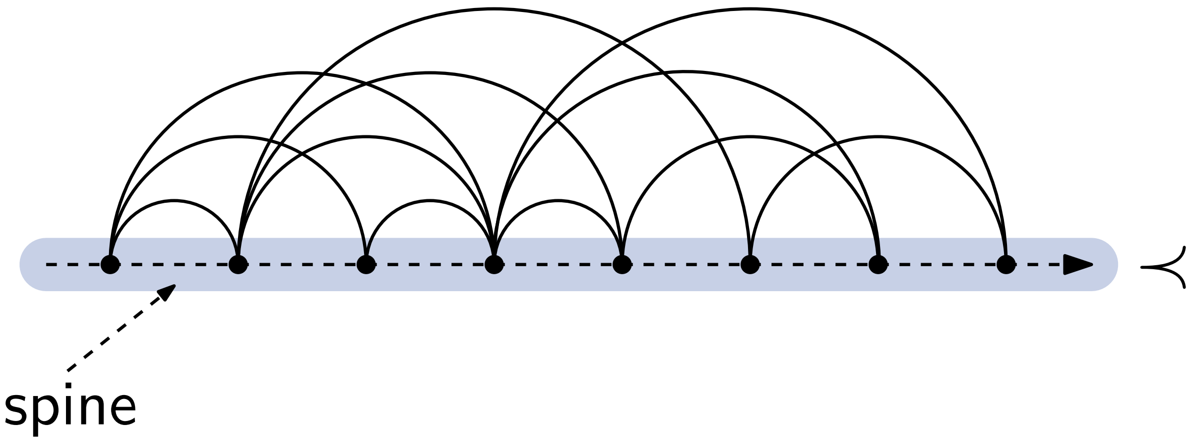


Stack Number: $\mathrm{sn}(G)$

Input: Graph G

Want:
- linear ordering $\prec$ of vertices
- k-coloring of edges, s.t:
  same color $\rightsquigarrow$ crossing-free

Stack Number: $\mathrm{sn}(G) := \min_{\prec} k$

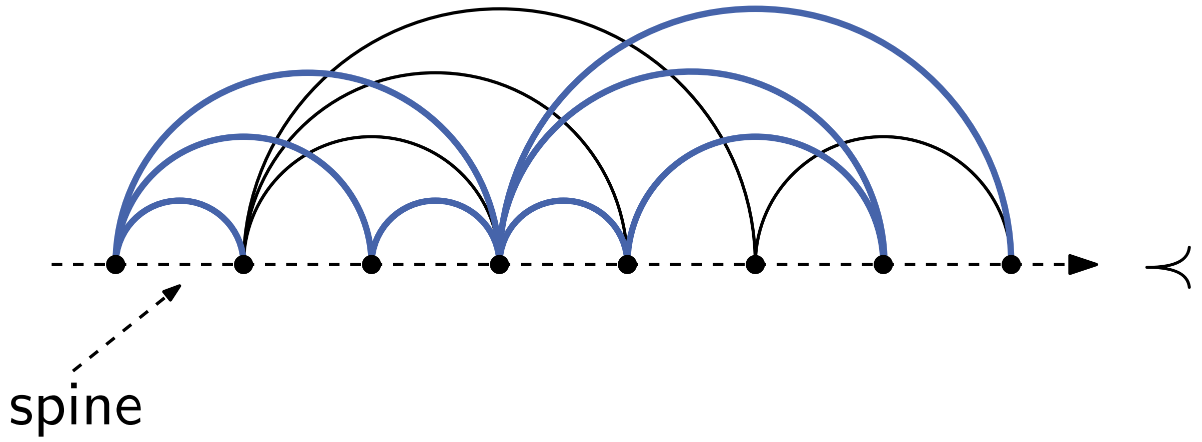


Stack Number: $\mathrm{sn}(G)$

Input: Graph G

Want:
- linear ordering $\prec$ of vertices
- k-coloring of edges, s.t:
  same color $\leadsto$ crossing-free

Stack Number: $\mathrm{sn}(G) := \min_{\prec} k$

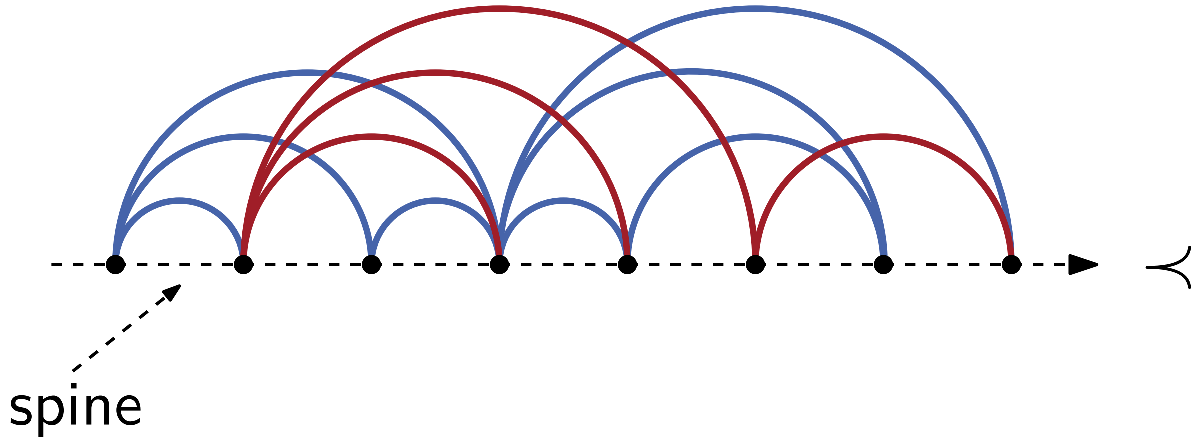


Stack Number: $\mathrm{sn}(G)$

Input: Graph G

Want:
- linear ordering $\prec$ of vertices
- k-coloring of edges, s.t:
  same color $\rightsquigarrow$ crossing-free

Stack Number: $\mathrm{sn}(G) := \min_{\prec} k$

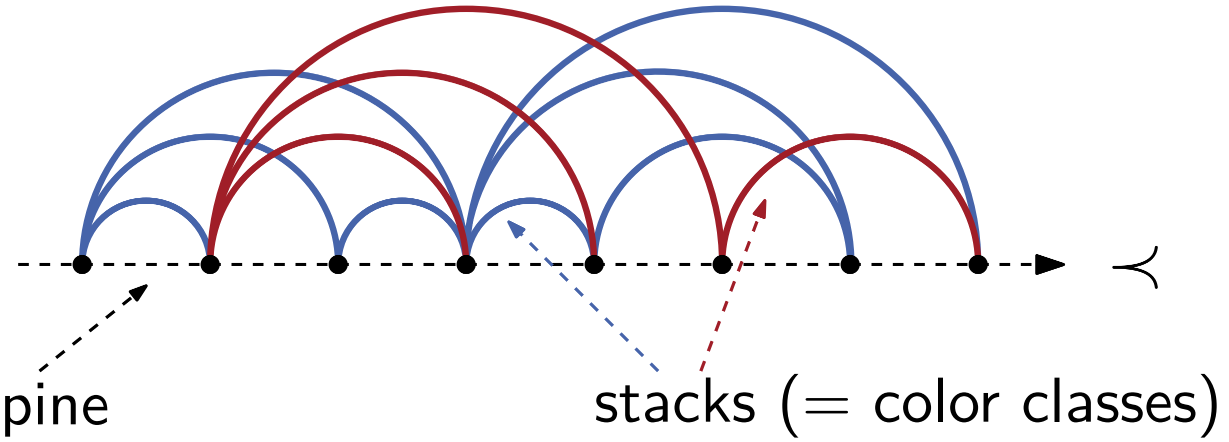


Stack Number: $\mathrm{sn}(G)$

Input: Graph G

Want:
- linear ordering $\prec$ of vertices
- k-coloring of edges, s.t:
  same color $\rightsquigarrow$ crossing-free

Stack Number: $\mathrm{sn}(G) := \min_{\prec} k$

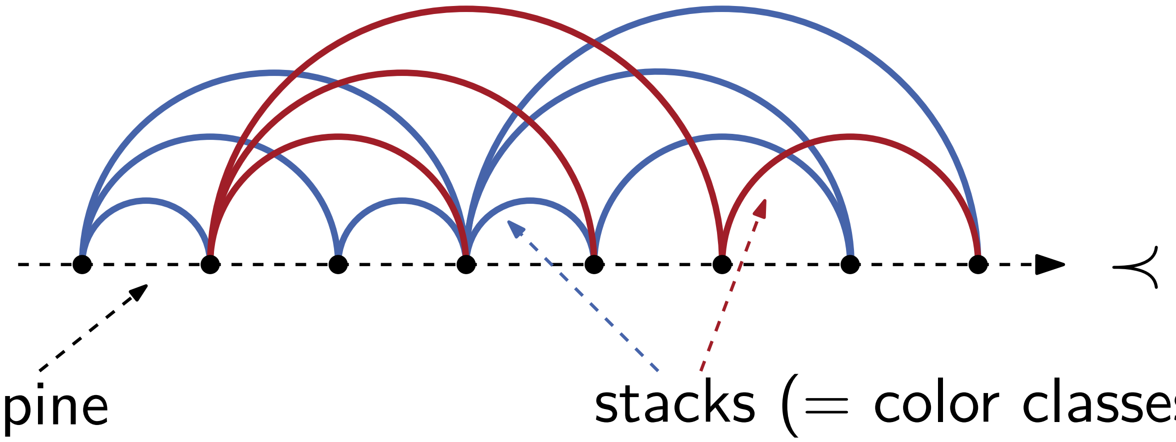


Stack Number: $\mathrm{sn}(G)$

Input: Graph G

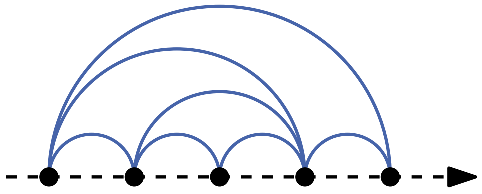
Want:
- linear ordering $\prec$ of vertices
- k-coloring of edges, s.t: same color $\rightsquigarrow$ crossing-free

Stack Number: $\mathrm{sn}(G) := \min_{\prec} k$

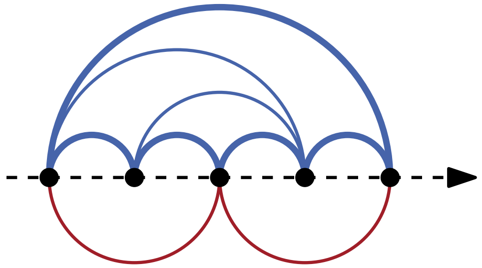


Examples:

Outerplanar: $\mathrm{sn}(G) = 1$



Hamiltonian $\mathrm{sn}(G) \leq 2$



Planar $\mathrm{sn}(G) \leq 4$ [Yannakakis 1989]

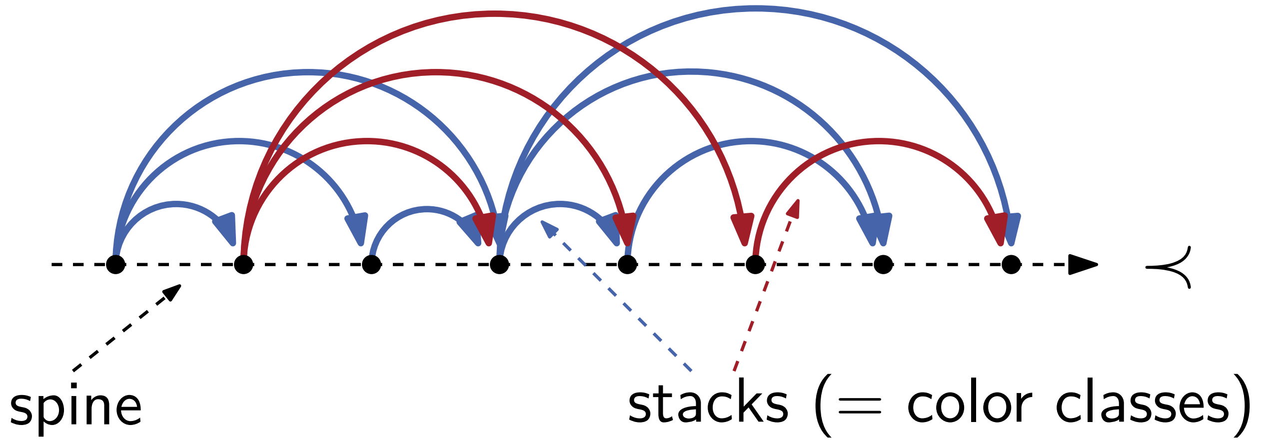
Paul Jungeblut, Laura Merker, Torsten Ueckerdt

Directed Stack Number: $\mathrm{sn}(G)$

Input: DAG G (directed acyclic graph)

Want:
- topological ordering $\prec$ of vertices
- k-coloring of edges, s.t: same color $\rightsquigarrow$ crossing-free

Stack Number: $\mathrm{sn}(G) := \min_{\prec} k$

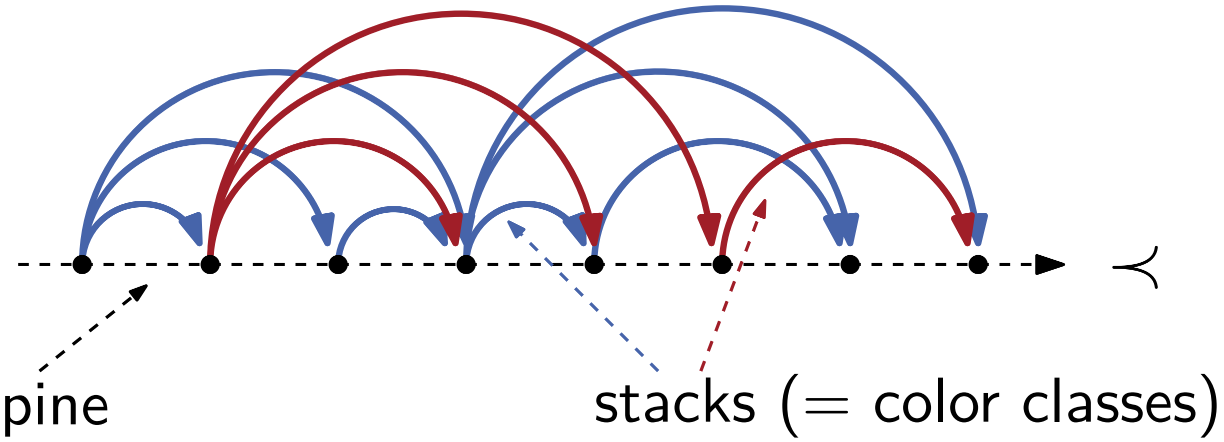


Directed Stack Number: $\mathrm{sn}(G)$

Input: DAG G (directed acyclic graph)

Want:
- topological ordering $\prec$ of vertices
- k-coloring of edges, s.t:
  same color $\rightsquigarrow$ crossing-free

Stack Number: $\mathrm{sn}(G) := \min_{\prec} k$



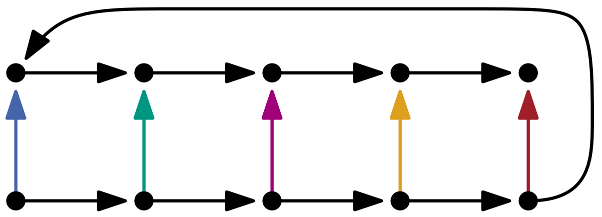
Examples:

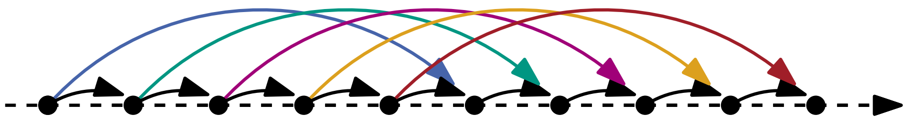
Trees:

$\mathrm{sn}(G) = 1$

Planar:

$\mathrm{sn}(G) = \infty$





Paul Jungeblut, Laura Merker, Torsten Ueckerdt

Our Contribution

Conjecture: (Heath, Pemmaraju, Trenk 1999)
Outerplanar DAGs have constant stack number.

- ✓ cacti
- ✓ single-source outerplanar
- ✓ monotone outerplanar
- ✓ outerpaths

Our Contribution

Conjecture: (Heath, Pemmaraju, Trenk 1999)
Outerplanar DAGs have constant stack number.

✓ cacti
✓ single-source outerplanar
✓ monotone outerplanar
✓ outerpaths

Theorem: (JMU 2023)
G is outerplanar: $\mathrm{sn}(G) \leq 24776$

new decomposition for outerplanar DAGs

Our Contribution

Conjecture: (Heath, Pemmaraju, Trenk 1999)
Outerplanar DAGs have constant stack number.

- ✓ cacti
- ✓ single-source outerplanar
- ✓ monotone outerplanar
- ✓ outerpaths

Theorem: (JMU 2023)
G is outerplanar: $\mathrm{sn}(G) \leq 24776$

new decomposition for outerplanar DAGs

Best possible:

Theorem: (JMU 2023)
G is a "very simple" 2-tree: $\mathrm{sn}(G) = \infty$

2-trees are a slightly larger graph class than outerplanar DAGs

Outerplanar Graphs and 2-Trees

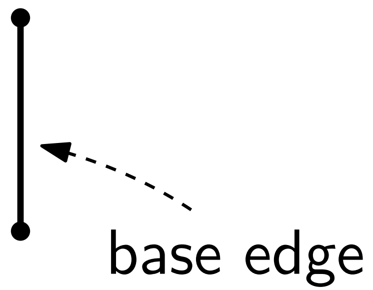
2-Tree:

- Base edge: 
- G is a 2-tree, $uv \in E(G)$.
  Then G' is a 2-tree:
  - $V(G') = V(G) \cup \{x\}$
  - $E(G') = E(G) \cup \{ux, vx\}$

Outerplanar Graphs and 2-Trees

2-Tree:

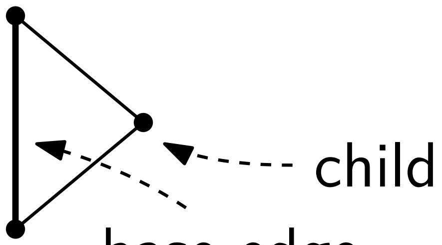
- Base edge: 
- G is a 2-tree, $uv \in E(G)$.
  Then G' is a 2-tree:
  - $V(G') = V(G) \cup \{x\}$
  - $E(G') = E(G) \cup \{ux, vx\}$



Outerplanar Graphs and 2-Trees

2-Tree:

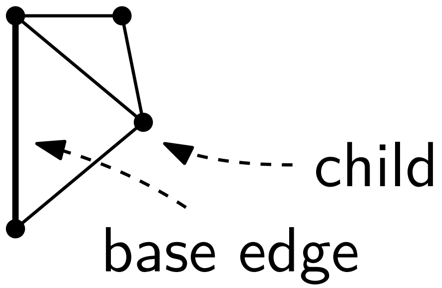
- Base edge: 
- G is a 2-tree, $uv \in E(G)$.
  Then G' is a 2-tree:
  - $V(G') = V(G) \cup \{x\}$
  - $E(G') = E(G) \cup \{ux, vx\}$



Outerplanar Graphs and 2-Trees

2-Tree:

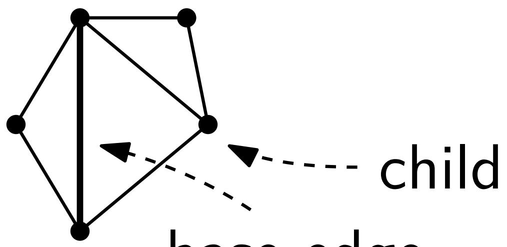
- Base edge: 
- G is a 2-tree, $uv \in E(G)$.
  Then G' is a 2-tree:
  - $V(G') = V(G) \cup \{x\}$
  - $E(G') = E(G) \cup \{ux, vx\}$



Outerplanar Graphs and 2-Trees

2-Tree:

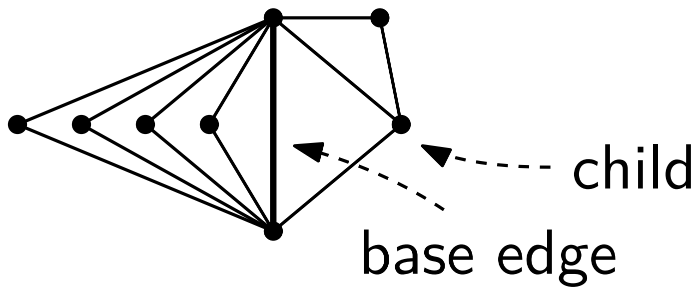
- Base edge: 
- G is a 2-tree, $uv \in E(G)$.
  Then G' is a 2-tree:
  - $V(G') = V(G) \cup \{x\}$
  - $E(G') = E(G) \cup \{ux, vx\}$



Outerplanar Graphs and 2-Trees

2-Tree:

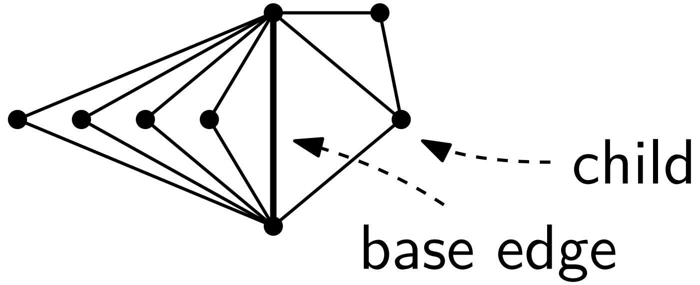
- Base edge: 
- G is a 2-tree, $uv \in E(G)$.
  Then G' is a 2-tree:
  - $V(G') = V(G) \cup \{x\}$
  - $E(G') = E(G) \cup \{ux, vx\}$



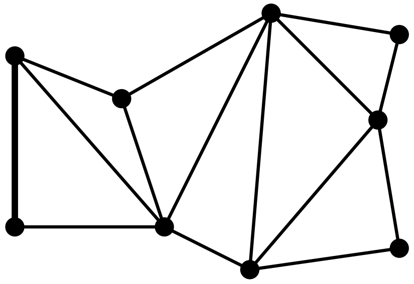
Outerplanar Graphs and 2-Trees

2-Tree:

- Base edge: 
- G is a 2-tree, $uv \in E(G)$. Then G' is a 2-tree:
  - $V(G') = V(G) \cup \{x\}$
  - $E(G') = E(G) \cup \{ux, vx\}$



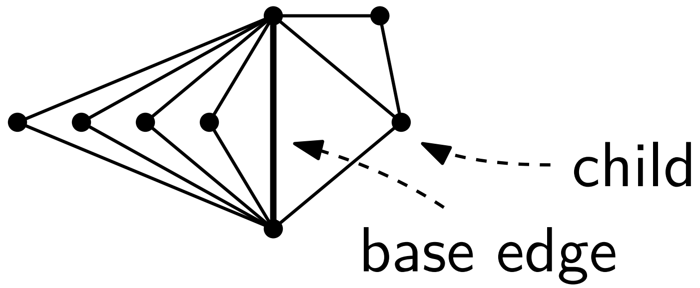
Outerplanar Graph:



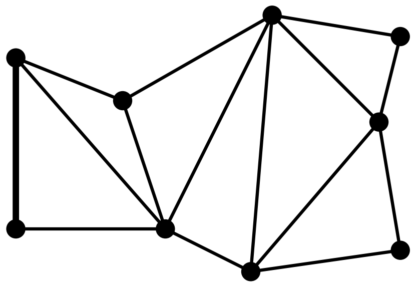
Outerplanar Graphs and 2-Trees

2-Tree:

- Base edge: 
- G is a 2-tree, $uv \in E(G)$.
  Then G' is a 2-tree:
  - $V(G') = V(G) \cup \{x\}$
  - $E(G') = E(G) \cup \{ux, vx\}$



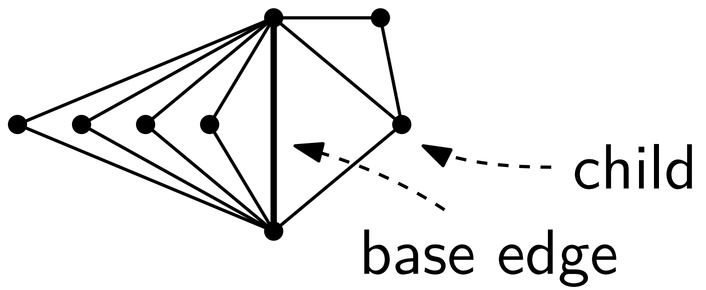
Outerplanar Graph:
(Subgraph of a) 2-tree in which each edge has at most one child.



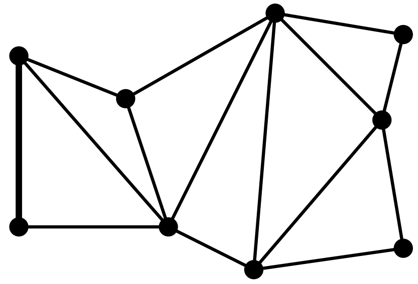
Outerplanar Graphs and 2-Trees

2-Tree:

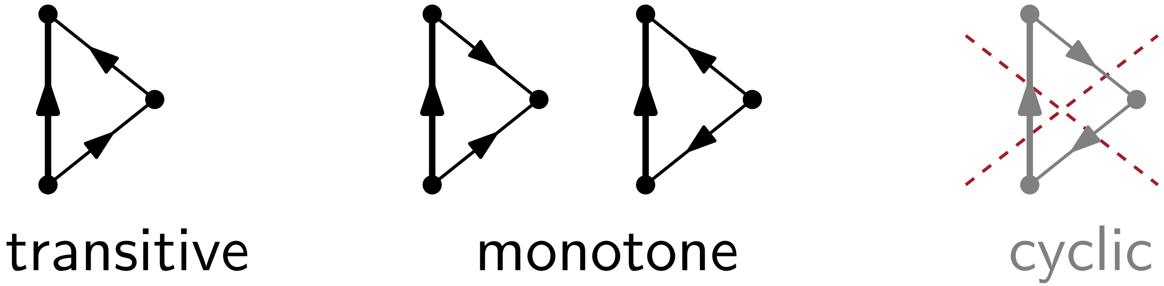
- Base edge: 
- G is a 2-tree, $uv \in E(G)$.
  Then G' is a 2-tree:
  - $V(G') = V(G) \cup \{x\}$
  - $E(G') = E(G) \cup \{ux, vx\}$



Outerplanar Graph:
(Subgraph of a) 2-tree in which each edge has at most one child.

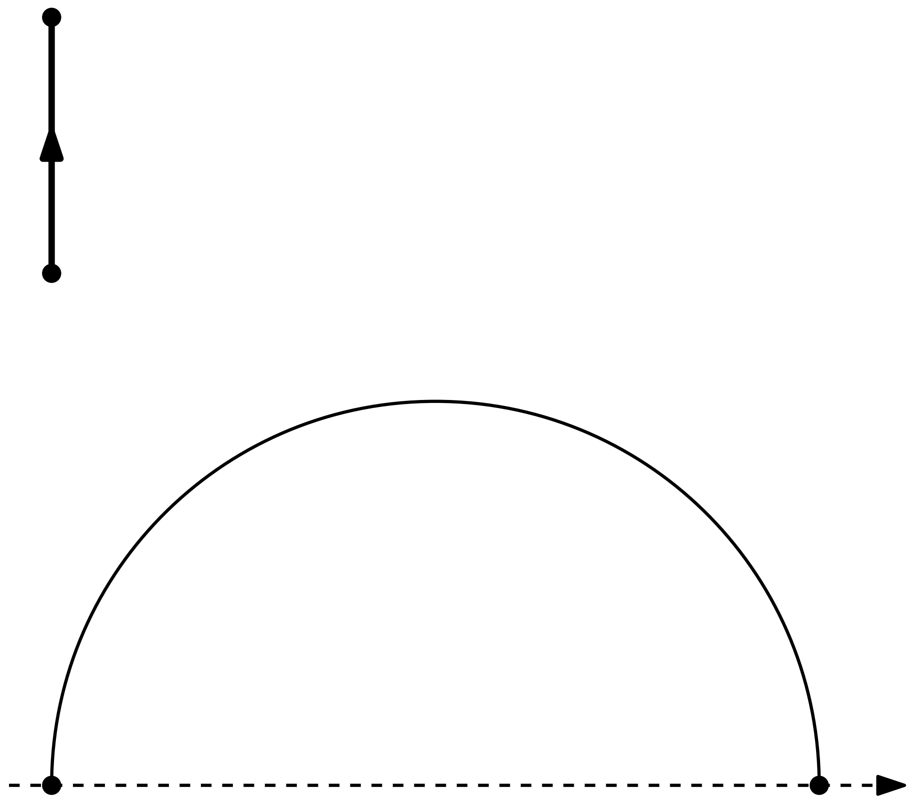


Directed:



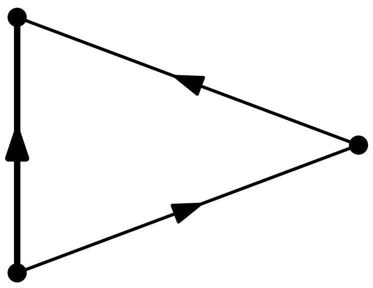
Special Cases

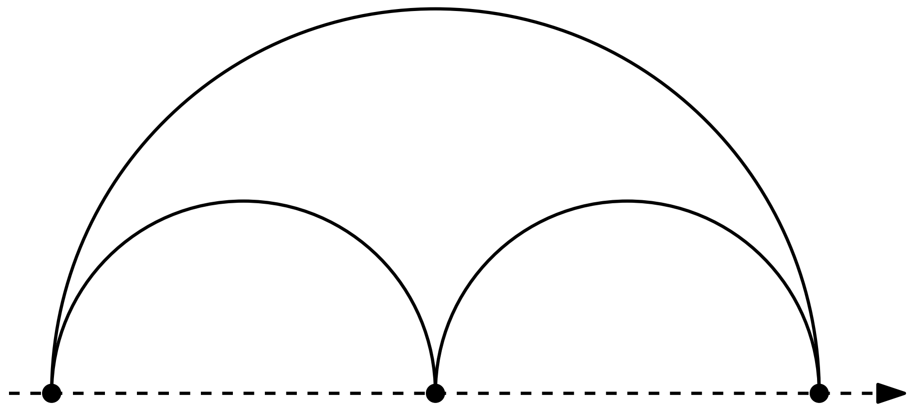
Transitive Outerplanar DAGs:



Special Cases

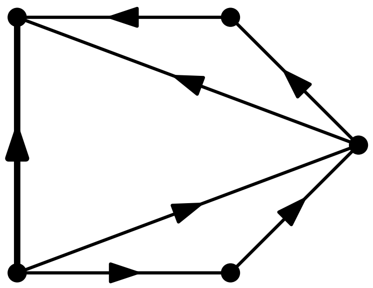
Transitive Outerplanar DAGs:

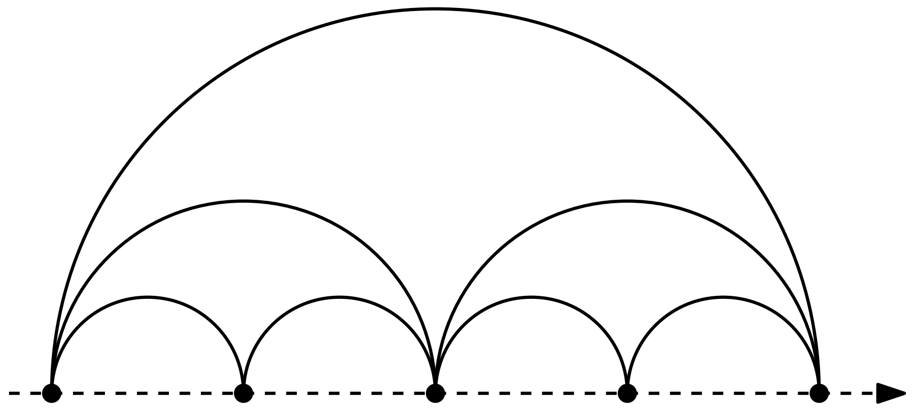




Special Cases

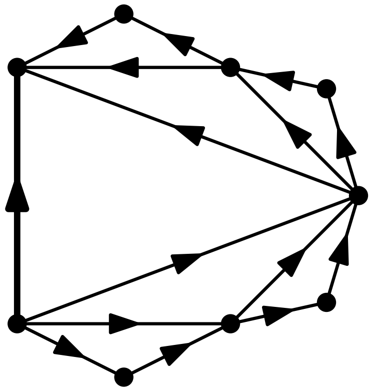
Transitive Outerplanar DAGs:



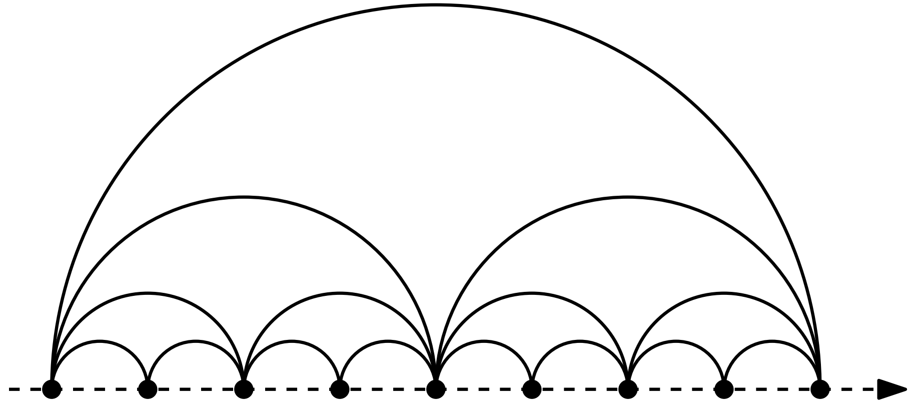


Special Cases

Transitive Outerplanar DAGs:

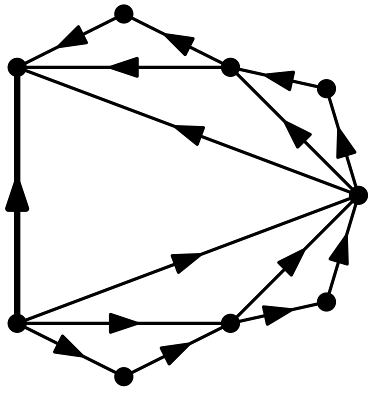


Observation:
Transitive G:
$\mathrm{sn}(G) = 1$

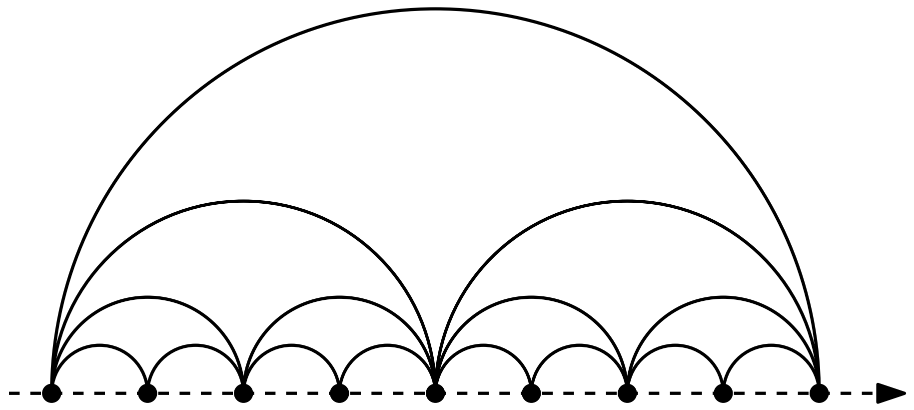


Special Cases

Transitive Outerplanar DAGs:



Observation:
Transitive G:
$\mathrm{sn}(G) = 1$

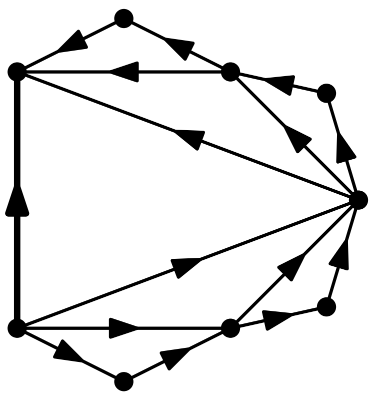


Montone Outerplanar DAGs:

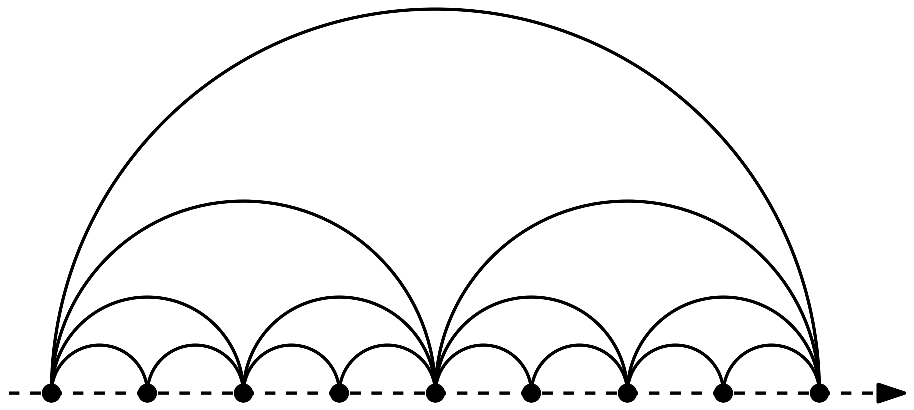
Theorem: (Nöllenburg, Pupyrev 2023)
Monotone outerplanar DAG G:
$\mathrm{sn}(G) \leq 128$

Special Cases

Transitive Outerplanar DAGs:



Observation:
Transitive G:
$\mathrm{sn}(G) = 1$



Montone Outerplanar DAGs:

Theorem: (Nöllenburg, Pupyrev 2023)
Monotone outerplanar DAG G:
$\mathrm{sn}(G) \leq 128$

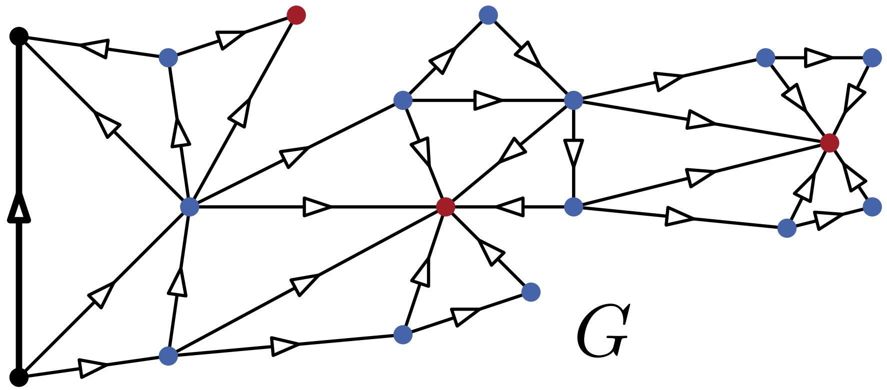
Block-Monotone:
Every biconnected component is monotone.

Lemma: (JMU 2023)
Block-monotone outerplanar DAG G:
$\mathrm{sn}(G) \leq 258$

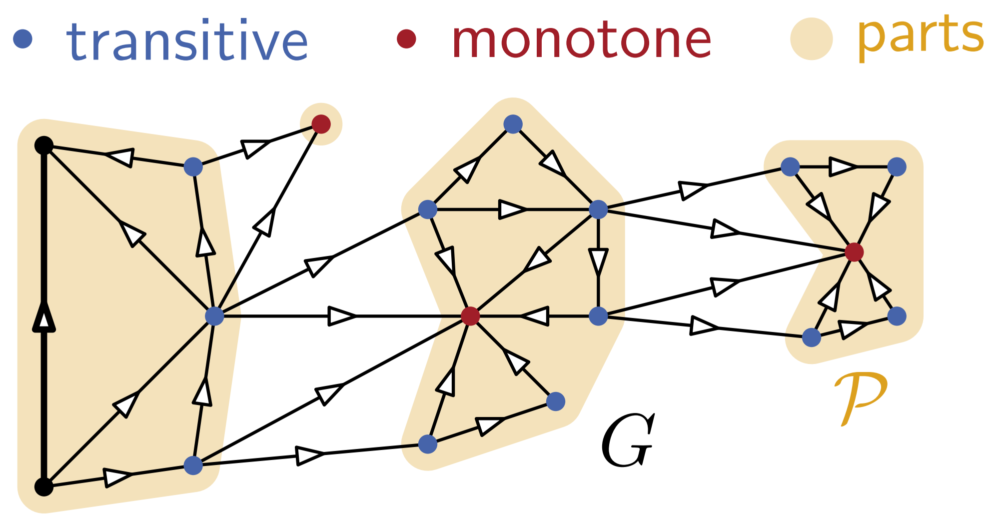
Paul Jungeblut, Laura Merker, Torsten Ueckerdt

Directed H-Partitions

- transitive
- monotone



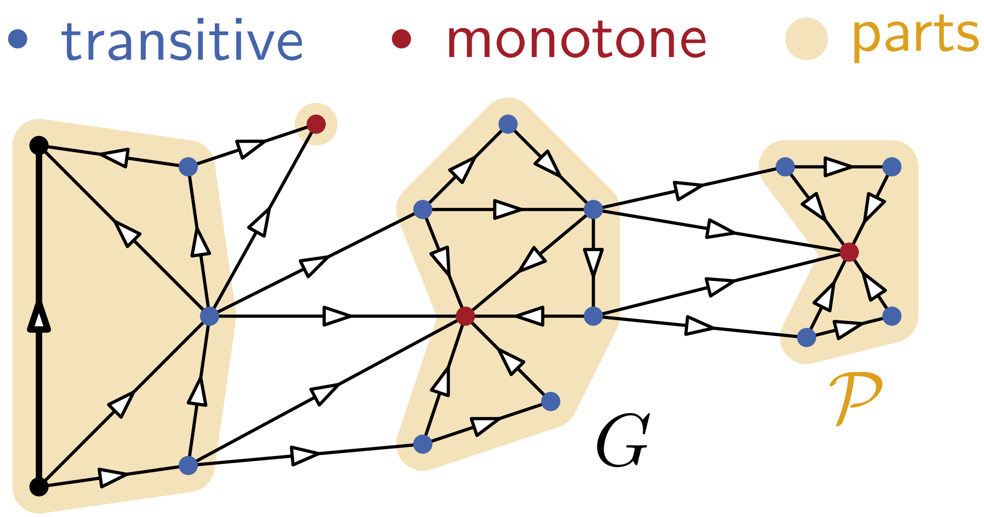
Directed H-Partitions



Partition vertices:

- start with base edge
- transitive $\rightsquigarrow$ add to current part
- monotone $\rightsquigarrow$ new part

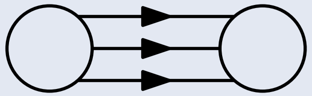
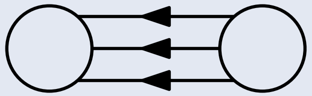
Directed H-Partitions



Partition vertices:

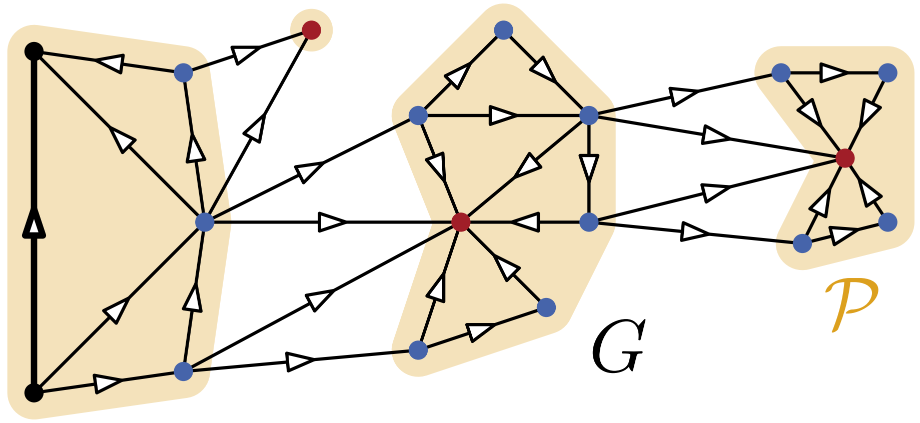
- start with base edge
- transitive $\rightsquigarrow$ add to current part
- monotone $\rightsquigarrow$ new part

Definition: (Directed H-Partition)

- partition $\mathcal{P}$ of $V(G)$
- quotient $G/\mathcal{P} \cong H$ (contract each part into single vertex)
- between each two parts:  or 

Constructing Stack Layouts

• transitive • monotone ● parts

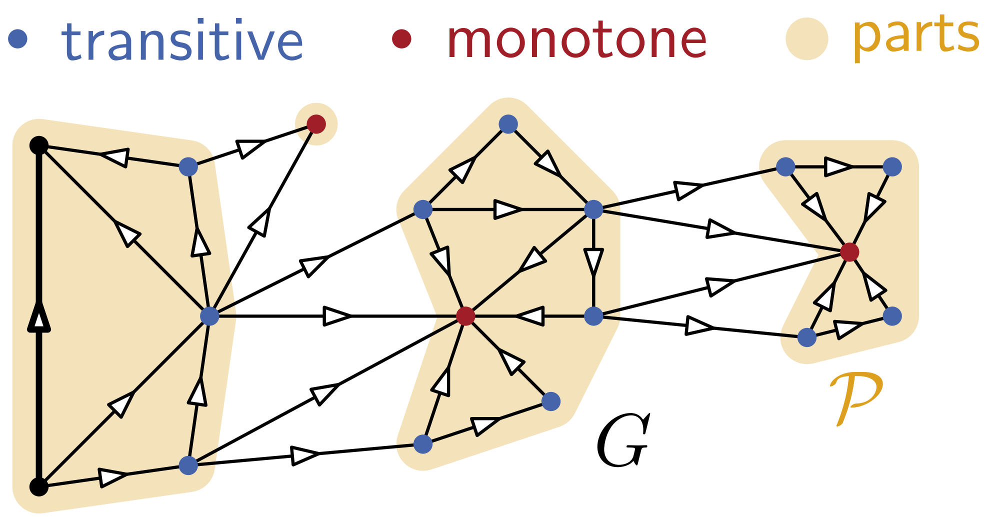


Lemma: (JMU 2023)
An outerplanar DAG G has a directed H-partition $\mathcal{P}$, such that:

- $H = G/\mathcal{P}$ is block-monotone
- each part is "transitive"

(+ other useful properties)

Constructing Stack Layouts

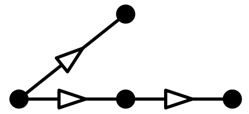


Lemma: (JMU 2023)
An outerplanar DAG G has a directed H-partition $\mathcal{P}$, such that:

- $H = G/\mathcal{P}$ is block-monotone
- each part is "transitive"

(+ other useful properties)

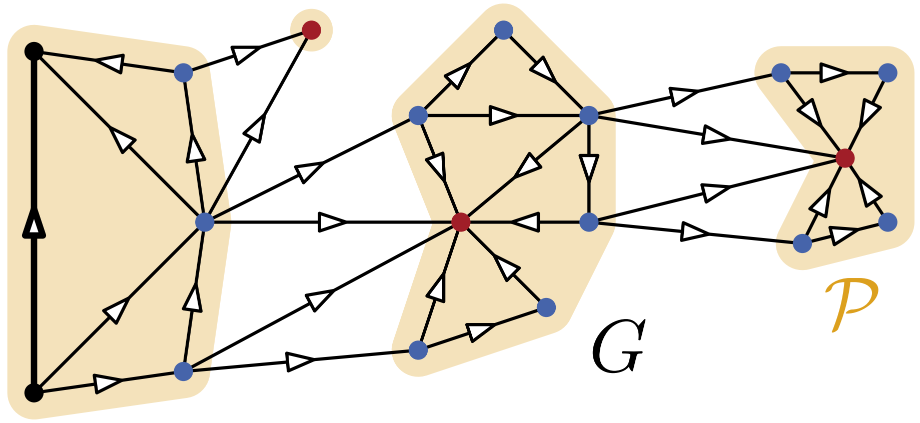
Proof Strategy:



$$H = G/\mathcal{P}$$

Constructing Stack Layouts

• transitive • monotone ● parts



Lemma: (JMU 2023)
An outerplanar DAG G has a directed H-partition $\mathcal{P}$, such that:

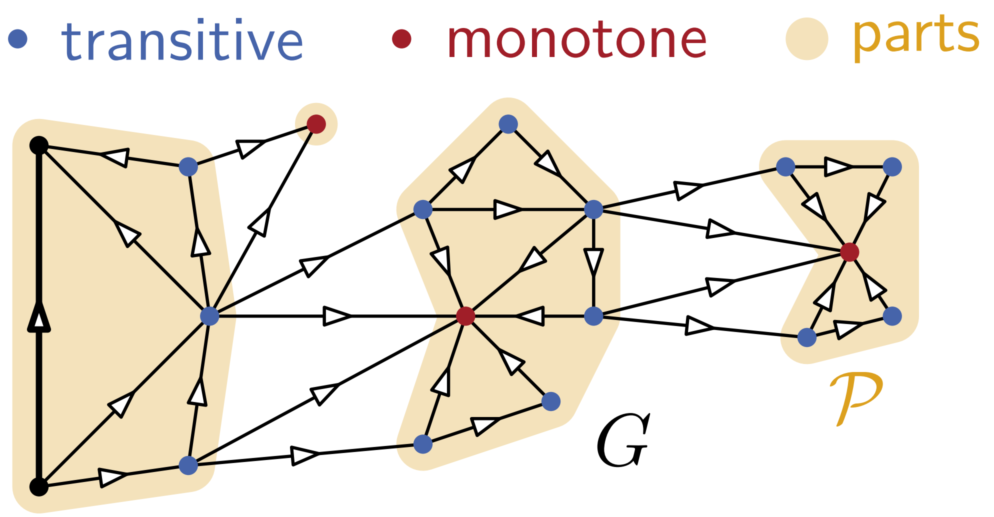
- $H = G/\mathcal{P}$ is block-monotone
- each part is "transitive"

(+ other useful properties)

Proof Strategy:

$H = G/\mathcal{P}$ $\quad\rightsquigarrow\quad$ H

Constructing Stack Layouts

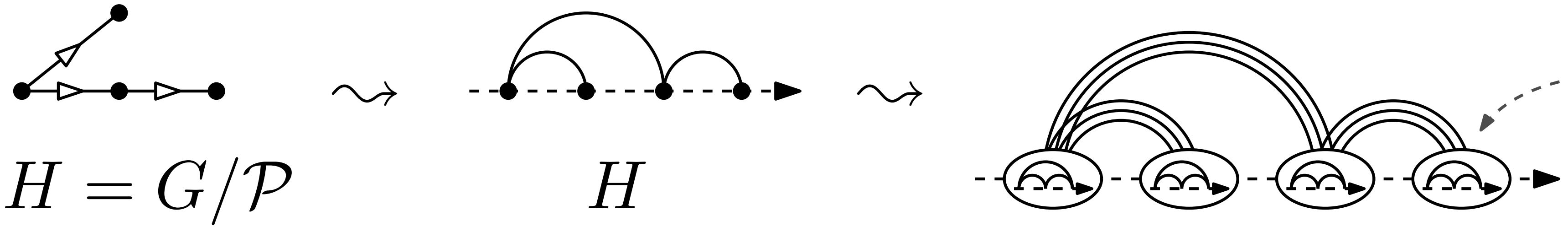


Lemma: (JMU 2023)
An outerplanar DAG G has a directed H-partition $\mathcal{P}$, such that:

- $H = G/\mathcal{P}$ is block-monotone
- each part is “transitive”

(+ other useful properties)

Proof Strategy:



hiding many, many details to make these compatible

Best Possible

Recall:

- an outerplanar DAG G is a 2-tree with at most one child per edge
- $\mathrm{sn}(G) \leq 24776$

Questions:

- Stack number of general directed 2-trees?
- Where is the boundary between constant and unbounded stack number?

Best Possible

Recall:

- an outerplanar DAG G is a 2-tree with at most one child per edge
- $\mathrm{sn}(G) \leq 24776$

Theorem: (JMU 2023)
For every k there exists a 2-tree G with stack number $\mathrm{sn}(G) = k$.

Questions:

- Stack number of general directed 2-trees?
- Where is the boundary between constant and unbounded stack number?

Best Possible

Recall:

- an outerplanar DAG G is a 2-tree with at most one child per edge
- $\mathrm{sn}(G) \leq 24776$

Theorem: (JMU 2023)
For every k there exists a 2-tree G with stack number $\mathrm{sn}(G) = k$.

Additionally:

- G is monotone
- at most two children per edge

$\rightsquigarrow$ outerplanar DAGs are right at that boundary

Questions:

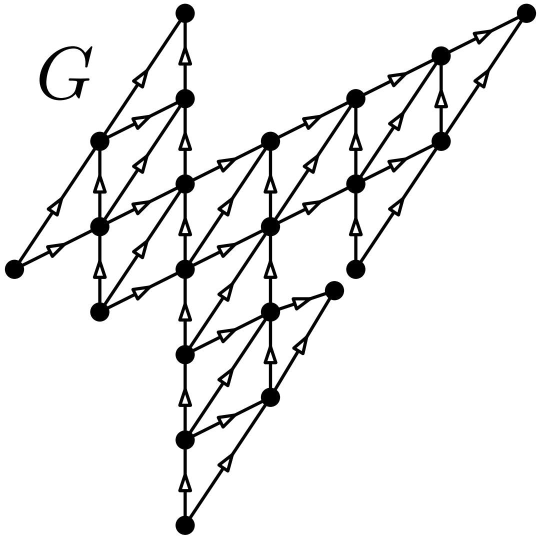
- Stack number of general directed 2-trees?
- Where is the boundary between constant and unbounded stack number?

Open Problems

Problem 1:
Precise bound for stack number of outerplanar DAGs?

JMU 2023:
$\mathrm{sn}(G) \leq 24776$

Nöllenburg + Pupyrev 2023:
$\mathrm{sn}(G) \geq 4$



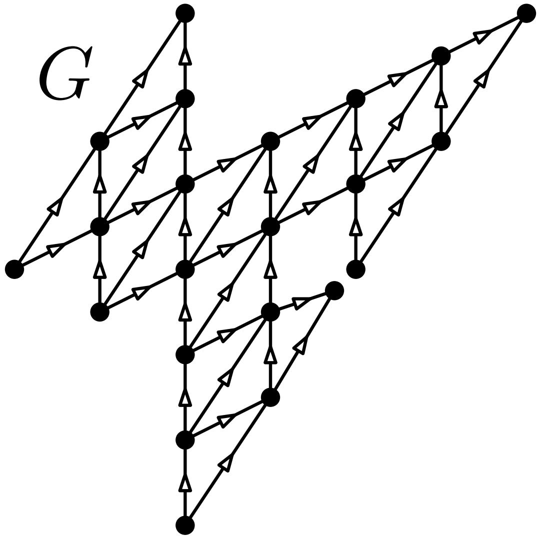
$\mathrm{sn}(G) = 4$

Open Problems

Problem 1:
Precise bound for stack number of outerplanar DAGs?

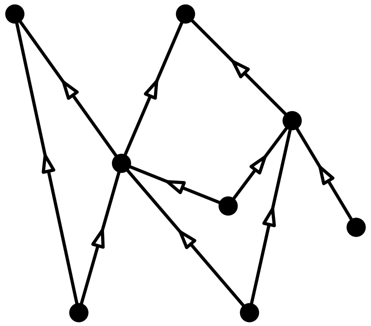
JMU 2023:
$\text{sn}(G) \leq 24776$

Nöllenburg + Pupyrev 2023:
$\text{sn}(G) \geq 4$



$\text{sn}(G) = 4$

Problem 2:
What is the stack number of upward planar graphs?



planar +
all edges upward

JMU 2022:
$5 \leq \text{sn}(G) \in O\left((n \log n)^{\frac{2}{3}}\right)$