DIRECTED TREE-WIDTH

Robin Thomas

School of Mathematics
Georgia Institute of Technology
www.math.gatech.edu/~thomas

joint work with

T. Johnson, N. Robertson, P. D. Seymour
OUTLINE

- Tree-width and havens for undirected graphs
- Even directed circuits
- Packing directed circuits
- Path-width of directed graphs
- Tree-width of directed graphs
- Havens in directed graphs
- Algorithms
A **tree-decomposition** of a graph $G$ is $(T, W)$, where $T$ is a tree and $W = (W_t : t \in V(T))$ satisfies

(T1) $\bigcup_{t \in V(T)} W_t = V(G)$,

(T2) if $t' \in T[t, t'']$, then $W_t \cap W_{t''} \subseteq W_{t'}$,

(T3) $\forall uv \in E(G) \exists t \in V(T)$ s.t. $u, v \in W_t$.

The **width** is $\max(|W_t| - 1 : t \in V(T))$.

The **tree-width** of $G$ is the minimum width of a tree-decomposition of $G$. 
• $tw(G) \leq 1 \iff G$ is a forest
• $tw(G) \leq 2 \iff G$ is series-parallel
• $tw(G) \leq 3 \iff$ no minor isomorphic to: $K_5$, 5-prism, octahedron, $V_8$
• $tw(K_n) = n - 1$
• tree-width is minor-monotone
• The $k \times k$ grid has tree-width $k$
Consider all functions $\phi$ mapping graphs into integers such that

1. $\phi(K_n) = n - 1$,

2. $G$ minor of $H \Rightarrow \phi(G) \leq \phi(H)$,

3. If $G \cap H$ is a clique, then $\phi(G \cup H) = \max\{\phi(G), \phi(H)\}$.

Order such functions by $\phi \leq \psi$ if $\phi(G) \leq \psi(G)$ for all $G$.

**THEOREM (Halin)** Tree-width is the maximum element in the above poset.
A haven $\beta$ of order $k$ in $G$ assigns to every $X \in [V(G)]^{<k}$ the vertex-set of a component of $G \setminus X$ such that

$$(H) \quad X \subseteq Y \in [V(G)]^{<k} \Rightarrow \beta(Y) \subseteq \beta(X).$$
A haven $\beta$ of order $k$ in $G$ assigns to every $X \in [V(G)]^{<k}$ the vertex-set of a component of $G \setminus X$ such that

$$(H) \ X \subseteq Y \in [V(G)]^{<k} \Rightarrow \beta(Y) \subseteq \beta(X).$$
A haven $\beta$ of order $k$ in $G$ assigns to every $X \in [V(G)]^{<k}$ the vertex-set of a component of $G \setminus X$ such that

\[(H) \quad X \subseteq Y \in [V(G)]^{<k} \Rightarrow \beta(Y) \subseteq \beta(X).\]
A haven $\beta$ of order $k$ in $G$ assigns to every $X \in [V(G)]^{<k}$ the vertex-set of a component of $G \setminus X$ such that

\[(H) \quad X \subseteq Y \in [V(G)]^{<k} \Rightarrow \beta(Y) \subseteq \beta(X).\]
A haven $\beta$ of order $k$ in $G$ assigns to every $X \in [V(G)]^{<k}$ the vertex-set of a component of $G\setminus X$ such that

$$(H) \quad X \subseteq Y \in [V(G)]^{<k} \Rightarrow \beta(Y) \subseteq \beta(X).$$
Cops and robbers. Fix a graph $G$ and an integer $k$. There are $k$ cops, they move slowly in helicopters. There is a robber, who moves infinitely fast along cop-free paths. He can see a helicopter landing, and can run to a safe place before the chopper lands.
Cops and robbers. Fix a graph $G$ and an integer $k$. There are $k$ cops, they move slowly in helicopters. There is a robber, who moves infinitely fast along cop-free paths. He can see a helicopter landing, and can run to a safe place before the chopper lands.
Cops and robbers. Fix a graph $G$ and an integer $k$. There are $k$ cops, they move slowly in helicopters. There is a robber, who moves infinitely fast along cop-free paths. He can see a helicopter landing, and can run to a safe place before the chopper lands.
Cops and robbers. Fix a graph $G$ and an integer $k$. There are $k$ cops, they move slowly in helicopters. There is a robber, who moves infinitely fast along cop-free paths. He can see a helicopter landing, and can run to a safe place before the chopper lands.
Cops and robbers. Fix a graph $G$ and an integer $k$. There are $k$ cops, they move slowly in helicopters. There is a robber, who moves infinitely fast along cop-free paths. He can see a helicopter landing, and can run to a safe place before the chopper lands.
**Cops and robbers.** Fix a graph $G$ and an integer $k$. There are $k$ cops, they move slowly in helicopters. There is a robber, who moves infinitely fast along cop-free paths. He can see a helicopter landing, and can run to a safe place before the chopper lands.
Cops and robbers. Fix a graph $G$ and an integer $k$. There are $k$ cops, they move slowly in helicopters. There is a robber, who moves infinitely fast along cop-free paths. He can see a helicopter landing, and can run to a safe place before the chopper lands.
Cops and robbers. Fix a graph $G$ and an integer $k$. There are $k$ cops, they move slowly in helicopters. There is a robber, who moves infinitely fast along cop-free paths. He can see a helicopter landing, and can run to a safe place before the chopper lands.
Cops and robbers. Fix a graph $G$ and an integer $k$. There are $k$ cops, they move slowly in helicopters. There is a robber, who moves infinitely fast along cop-free paths. He can see a helicopter landing, and can run to a safe place before the chopper lands.
Cops and robbers. Fix a graph $G$ and an integer $k$. There are $k$ cops, they move slowly in helicopters. There is a robber, who moves infinitely fast along cop-free paths. He can see a helicopter landing, and can run to a safe place before the chopper lands.
Cops and robbers. Fix a graph $G$ and an integer $k$. There are $k$ cops, they move slowly in helicopters. There is a robber, who moves infinitely fast along cop-free paths. He can see a helicopter landing, and can run to a safe place before the chopper lands.
Cops and robbers. Fix a graph $G$ and an integer $k$. There are $k$ cops, they move slowly in helicopters. There is a robber, who moves infinitely fast along cop-free paths. He can see a helicopter landing, and can run to a safe place before the chopper lands.
Cops and robbers. Fix a graph $G$ and an integer $k$. There are $k$ cops, they move slowly in helicopters. There is a robber, who moves infinitely fast along cop-free paths. He can see a helicopter landing, and can run to a safe place before the chopper lands.

Fact. A tree-decomposition of width $k - 1$ gives a search strategy for $k$ cops.
Cops and robbers. Fix a graph $G$ and an integer $k$. There are $k$ cops, they move slowly in helicopters. There is a robber, who moves infinitely fast along cop-free paths. He can see a helicopter landing, and can run to a safe place before the chopper lands.

**Fact.** A tree-decomposition of width $k - 1$ gives a search strategy for $k$ cops.

**Fact.** A haven gives an escape strategy for the robber.
Cops and robbers. Fix a graph $G$ and an integer $k$. There are $k$ cops, they move slowly in helicopters. There is a robber, who moves infinitely fast along cop-free paths. He can see a helicopter landing, and can run to a safe place before the chopper lands.

Fact. A tree-decomposition of width $k - 1$ gives a search strategy for $k$ cops.

Fact. A haven gives an escape strategy for the robber.

THEOREM (Seymour, RT) $G$ has a haven of order $k$ $\iff$ $G$ has tree-width at least $k - 1$
Cops and robbers. Fix a graph $G$ and an integer $k$. There are $k$ cops, they move slowly in helicopters. There is a robber, who moves infinitely fast along cop-free paths. He can see a helicopter landing, and can run to a safe place before the chopper lands.

Fact. A tree-decomposition of width $k - 1$ gives a search strategy for $k$ cops.

Fact. A haven gives an escape strategy for the robber.

**THEOREM** (Seymour, RT) $G$ has a haven of order $k \iff G$ has tree-with at least $k - 1$

**COR** Search strategy $\implies$ monotone search strategy.
THEOREM (Robertson, Seymour, RT) Every graph of tree-width $\geq 20^{2g^5}$ has a $g \times g$ grid minor.
THEOREM (Robertson, Seymour, RT) Every graph of tree-width $\geq 20^{2g^5}$ has a $g \times g$ grid minor.

THEOREM (Bodlaender) For every $k$ there is a linear-time algorithm to decide whether $\text{tw}(G) \leq k$. 
THEOREM (Robertson, Seymour, RT) Every graph of tree-width $\geq 20^2g^5$ has a $g \times g$ grid minor.

THEOREM (Bodlaender) For every $k$ there is a linear-time algorithm to decide whether $\text{tw}(G) \leq k$.

THEOREM (Arnborg, Proskurowski, ...) Many problems can be solved in linear time when restricted to graphs of bounded tree-width.
Tree-width is useful in

- theory
- design of theoretically fast algorithms
- practical computations
FEEDBACK VERTEX-SET FOR FIXED $k$

INSTANCE A graph $G$

QUESTION Is there a set $X \subseteq V(G)$ such that $|X| \leq k$ and $G \setminus X$ is acyclic?

ALGORITHM If $\text{tw}(G)$ is small use bounded tree-width methods. Otherwise answer “no”. That’s correct, because big tree-width $\Rightarrow$ big grid $\Rightarrow k + 1$ disjoint circuits $\Rightarrow X$ does not exist.
$k$ DISJOINT PATHS IN PLANAR GRAPHS

INSTANCE A planar graph $G$, vertices $s_1, s_2, \ldots, s_k, t_1, t_2, \ldots, t_k$ of $G$

QUESTION Are there disjoint paths $P_1, \ldots, P_k$ such that $P_i$ has ends $s_i$ and $t_i$?
$k$ DISJOINT PATHS IN PLANAR GRAPHS

INSTANCE A planar graph $G$, vertices $s_1, s_2, \ldots, s_k, t_1, t_2, \ldots, t_k$ of $G$

QUESTION Are there disjoint paths $P_1, \ldots, P_k$ such that $P_i$ has ends $s_i$ and $t_i$?
\( k \) DISJOINT PATHS IN PLANAR GRAPHS

INSTANCE A planar graph \( G \), vertices \( s_1, s_2, \ldots, s_k, t_1, t_2, \ldots, t_k \) of \( G \)

QUESTION Are there disjoint paths \( P_1, \ldots, P_k \) such that \( P_i \) has ends \( s_i \) and \( t_i \)?

ALGORITHM \( \text{tw}(G) \) small \( \Rightarrow \) bounded tree-width methods. Otherwise big grid minor \( \Rightarrow \) big grid minor with the terminals outside. The middle vertex of this grid minor can be deleted, without affecting the feasibility of the problem.
MINORS IN DIGRAPHS
An edge in a digraph is **contractible** if either it is the only edge leaving its tail, or it is the only edge entering its head.

A digraph $D$ is a **butterfly minor** of a digraph $D'$ if $D$ can be obtained from a subdigraph of $D'$ by contracting contractible edges.
An edge in a digraph is **contractible** if either it is the only edge leaving its tail, or it is the only edge entering its head.

A digraph $D$ is a **minor** of a digraph $D'$ if $D$ can be obtained from a subdigraph of $D'$ by contracting contractible edges.
A digraph is even if every subdivision has an even directed circuit. An odd double cycle, $O_{2k+1}$:
A digraph is even if every subdivision has an even directed circuit. An odd double cycle, $O_{2k+1}$:

THEOREM (Seymour, Thomassen) A digraph is not even $\Leftrightarrow$ it has no odd double cycle minor.
A digraph is even if every subdivision has an even directed circuit. An odd double cycle, $O_{2k+1}$:

THEOREM (Seymour, Thomassen) A digraph is not even $\iff$ it has no odd double cycle minor.

THEOREM (McCuaig; Robertson, Seymour, RT) $\iff$ it can be obtained from strongly planar digraphs and $F_7$ by means of 0-, 1-, 2-, 3-, and 4-sums.
\[ \tau(D) = \min \{|X| \subseteq V(D) : D \setminus X \text{ is acyclic} \} \]

\[ \nu(D) = \text{max number of disjoint cycles} \]
\[ \tau(D) = \min\{|X| \subseteq V(D) : D\setminus X \text{ is acyclic}\} \]
\[ \nu(D) = \max \text{ number of disjoint cycles} \]

**THEOREM** (Guenin, RT) \[ \tau(D') = \nu(D') \] for every subdigraph \( D' \) of \( D \) \( \iff \) \( D \) has no \( O_{2k+1} \) or \( F_7 \) minor.
\[ \tau(D) = \min \{|X| \subseteq V(D) : D \setminus X \text{ is acyclic}\} \]

\[ \nu(D) = \text{max number of disjoint cycles} \]

**THEOREM (Guenin, RT)** \( \tau(D') = \nu(D') \) for every subdigraph \( D' \) of \( D \) \( \iff \) \( D \) has no \( O_{2k+1} \) or \( F_7 \) minor.

**THEOREM (McCuaig)** \( \nu(D) \leq 1 \implies \tau(D) \leq 3 \)
\( \tau(D) = \min\{|X| \subseteq V(D) : D \setminus X \text{ is acyclic}\} \)

\( \nu(D) = \max \) number of disjoint cycles

**THEOREM (Guenin, RT)** \( \tau(D') = \nu(D') \) for every subdigraph \( D' \) of \( D \) \( \iff \) \( D \) has no \( O_{2k+1} \) or \( F_7 \) minor.

**THEOREM (McCuaig)** \( \nu(D) \leq 1 \Rightarrow \tau(D) \leq 3 \)

**THEOREM (Reed, Robertson, Seymour, RT)** There is a function \( f \) such that \( \tau(D) \leq f(\nu(D)) \) for every \( D \).
DIRECTED TREE-WIDTH
An arboreal decomposition of $D$ is $(R, X, W)$, where $R$ is an arborescence, and $X = (X_e : e \in E(R))$ and $W = (W_r : r \in V(R))$ satisfy
An **arboreal decomposition** of $D$ is $(R, X, W)$, where $R$ is an arborescence, and $X = (X_e : e \in E(R))$ and $W = (W_r : r \in V(R))$ satisfy

(D1) $(W_r : r \in V(R))$ partitions $V(D)$

(D2) $\bigcup_{r > e} W_r$ induces a strong component of $D \setminus X_e$ for every $e \in E(R)$
An arboreal decomposition of $D$ is $(R, X, W)$, where $R$ is an arborescence, and $X = (X_e : e \in E(R))$ and $W = (W_r : r \in V(R))$ satisfy

(D1) $(W_r : r \in V(R))$ partitions $V(D)$

(D2) $\bigcup_{r > e} W_r$ induces a strong component of $D \setminus X_e$ for every $e \in E(R)$

The width is the min, over all $r \in V(R)$, of $|W_r \cup \bigcup_{e \sim r} X_e| - 1$. 
An arboreal decomposition of $D$ is $(R, X, W)$, where $R$ is an arborescence, and $X = (X_e : e \in E(R))$ and $W = (W_r : r \in V(R))$ satisfy

(D1) $(W_r : r \in V(R))$ partitions $V(D)$

(D2) $\bigcup_{r \sim e} W_r$ induces a strong component of $D \setminus X_e$ for every $e \in E(R)$

The width is the min, over all $r \in V(R)$, of $|W_r \cup \bigcup_{e \sim r} X_e| - 1$.

The tree-width of $D$ is the minimum width of an arboreal decomposition of $D$. 
An arboreal decomposition of $D$ is $(R, X, W)$, where $R$ is an arborescence, and $X = (X_e : e \in E(R))$ and $W = (W_r : r \in V(R))$ satisfy

\[(D1) \ (W_r : r \in V(R)) \text{ partitions } V(D)\]
\[(D2) \bigcup_{r > e} W_r \text{ induces a strong component of } D \setminus X_e \text{ for every } e \in E(R)\]

The width is the min, over all $r \in V(R)$, of $|W_r \cup \bigcup_{e \sim r} X_e| - 1$.

The tree-width of $D$ is the minimum width of an arboreal decomposition of $D$.

FACT Tree-width is minor-monotone.
A haven $\beta$ of order $k$ in $D$ assigns to every $X \in [V(D)]^{<k}$ the vertex-set of a strong component of $D \setminus X$ such that

(H) $X \subseteq Y \in [V(D)]^{<k} \Rightarrow \beta(Y) \subseteq \beta(X)$. 
A haven $\beta$ of order $k$ in $D$ assigns to every $X \in [V(D)]^{<k}$ the vertex-set of a strong component of $D \setminus X$ such that

(H) $X \subseteq Y \in [V(D)]^{<k} \Rightarrow \beta(Y) \subseteq \beta(X)$.

FACT Haven of order $k \Rightarrow \text{tw}(D) \geq k - 1$. 
A haven $\beta$ of order $k$ in $D$ assigns to every $X \in [V(D)]^{<k}$ the vertex-set of a strong component of $D \setminus X$ such that

(\text{H}) \quad X \subseteq Y \in [V(D)]^{<k} \implies \beta(Y) \subseteq \beta(X).

\textbf{FACT} \quad \text{Haven of order } k \Rightarrow \text{tw}(D) \geq k - 1.

\textbf{QUESTION} \quad \text{Converse?}
A haven $\beta$ of order $k$ in $D$ assigns to every $X \in [V(D)]^{<k}$ the vertex-set of a strong component of $D \setminus X$ such that

(H) $X \subseteq Y \in [V(D)]^{<k} \Rightarrow \beta(Y) \subseteq \beta(X)$.

**FACT** Haven of order $k \Rightarrow tw(D) \geq k - 1$.

**QUESTION** Converse? Open.
A haven $\beta$ of order $k$ in $D$ assigns to every $X \in [V(D)]^{<k}$ the vertex-set of a strong component of $D \setminus X$ such that

(H) $X \subseteq Y \in [V(D)]^{<k} \Rightarrow \beta(Y) \subseteq \beta(X)$.

FACT Haven of order $k$ $\Rightarrow$ tw($D$) $\geq k - 1$.

QUESTION Converse? Open.

THEOREM (Johnson, Robertson, Seymour, RT)
Haven of order $k$ $\iff$ tw($D$) $\geq 3k - 1$. 
COPS-AND-ROBBER GAME Same as for undirected graphs, except that robber must stay within strongly connected components of the cop-free subdigraph.
COPS-AND-ROBBER GAME Same as for undirected graphs, except that robber must stay within strongly connected components of the cop-free subdigraph.

A haven of order $k$ gives an escape strategy for the robber against $k - 1$ cops, and an arboreal decomposition of width $k - 1$ gives a search strategy for $k$ cops.
COPS-AND-ROBBER GAME  Same as for undirected graphs, except that robber must stay within strongly connected components of the cop-free subdigraph.

A haven of order $k$ gives an escape strategy for the robber against $k - 1$ cops, and an arboreal decomposition of width $k - 1$ gives a search strategy for $k$ cops.

REMARK  The search strategy need not be monotone.
ALGORITHMS
Let $Z \subseteq V(D)$, and let $S_1, \ldots, S_t$ be the strong components of $D \setminus Z$ such that no edge goes from $S_j$ to $S_i$ for $j > i$. Then $S = S_i \cup S_{i+1} \cup \ldots \cup S_j$ is $Z$-normal. If $|Z| \leq k$, then $S$ is $k$-protected.
Let $Z \subseteq V(D)$, and let $S_1, \ldots, S_t$ be the strong components of $D \setminus Z$ such that no edge goes from $S_j$ to $S_i$ for $j > i$. Then $S = S_i \cup S_{i+1} \cup \ldots \cup S_j$ is $Z$-normal. If $|Z| \leq k$, then $S$ is $k$-protected.
For some $k$-protected sets $A$ we will compute an itinerary for $A$. 

For some $k$-protected sets $A$ we will compute an itinerary for $A$.

**AXIOM 1** $A, B \subseteq V(D)$ disjoint, no edge of $D$ has head in $A$ and tail in $B$. Then an itinerary for $A \cup B$ can be computed from itineraries of $A$ and $B$ in time $O((|A| + |B|)^\alpha)$. 
For some $k$-protected sets $A$ we will compute an itinerary for $A$.

**AXIOM 1** $A, B \subseteq V(D)$ disjoint, no edge of $D$ has head in $A$ and tail in $B$. Then an itinerary for $A \cup B$ can be computed from itineraries of $A$ and $B$ in time $O((|A| + |B|)^\alpha)$.

**AXIOM 2** $A, B \subseteq V(D)$ disjoint sets, $A$ is $k$-protected and $|B| \leq k$. Then an itinerary for $A \cup B$ can be computed from itineraries of $A$ and $B$ in time $O((|A| + 1)^\alpha)$. 
AXIOM 1

A → B → A

A

B
AXIOM 1

AXIOM 2

k−protected
<k
THEOREM (Johnson, Robertson, Seymour, RT) There is a polynomial-time algorithm for:

INPUT A digraph $D$ with an arboreal decomposition of bounded width.

OUTPUT An itinerary for $V(D)$
THEOREM (Johnson, Robertson, Seymour, RT) There is a polynomial-time algorithm for:

**INPUT** A digraph $D$ with an arboreal decomposition of bounded width.

**OUTPUT** An itinerary for $V(D)$

Thus HAMILTON PATH, HAMILTON CIRCUIT, $k$-DISJOINT PATHS ($k$ fixed) and other problems can be solved in polynomial time for digraphs of bounded tree-width.
CONJECTURE There is a function $f$ such that every digraph of tree-width at least $f(k)$ has a cylindrical $k \times k$ grid minor.
HOW TO USE A HAVEN?

REMINDER A haven $\beta$ of order $k$ in $D$ assigns to every $X \in [V(D)]^{<k}$ the vertex-set of a strong component of $D \setminus X$ such that

(H) $X \subseteq Y \in [V(D)]^{<k} \Rightarrow \beta(Y) \subseteq \beta(X)$. 
Let $\beta$ be a haven of order $k$ in $G$. Let $X \subseteq V(G)$ with $|X| \leq k/2$ and $\beta(X)$ minimum. Then $X$ is “externally linked”: 

![Diagram showing the concepts of $X$ and $\beta(X)$]
Let $\beta$ be a haven of order $k$ in $G$. Let $X \subseteq V(G)$ with $|X| \leq k/2$ and $\beta(X)$ minimum. Then $X$ is “externally linked”: 
Let $\beta$ be a haven of order $k$ in $G$. Let $X \subseteq V(G)$ with $|X| \leq k/2$ and $\beta(X)$ minimum. Then $X$ is “externally linked”: 
Let $\beta$ be a haven of order $k$ in $G$. Let $X \subseteq V(G)$ with $|X| \leq k/2$ and $\beta(X)$ minimum. Then $X$ is “externally linked”: 
Let $\beta$ be a haven of order $k$ in $G$. Let $X \subseteq V(G)$ with $|X| \leq k/2$ and $\beta(X)$ minimum. Then $X$ is “externally linked”:
Let $\beta$ be a haven of order $k$ in $G$. Let $X \subseteq V(G)$ with $|X| \leq k/2$ and $\beta(X)$ minimum. Then $X$ is “externally linked”:
Let $\beta$ be a haven of order $k$ in $G$. Let $X \subseteq V(G)$ with $|X| \leq k/2$ and $\beta(X)$ minimum. Then $X$ is “externally linked”: 

\[ \beta(X+Z) = \beta(Y) \]
A directed path decomposition of $D$ is a sequence $W_1, W_2, \ldots, W_n$ such that

(i) $\bigcup W_i = V(D)$,

(ii) if $i < i' < i''$ then $W_i \cap W_{i''} \subseteq W_{i'}$,

(iii) for every edge $uv \in E(D)$ there exist $i \leq j$ such that $u \in W_i$ and $v \in W_j$. 
A directed path decomposition of $D$ is a sequence $W_1, W_2, \ldots, W_n$ such that

(i) $\bigcup W_i = V(D)$,

(ii) if $i < i' < i''$ then $W_i \cap W_{i''} \subseteq W_{i'}$,

(iii) for every edge $uv \in E(D)$ there exist $i \leq j$ such that $u \in W_i$ and $v \in W_j$. 
A directed path decomposition of $D$ is a sequence $W_1, W_2, \ldots, W_n$ such that

(i) $\bigcup W_i = V(D)$,

(ii) if $i < i' < i''$ then $W_i \cap W_{i''} \subseteq W_i'$,

(iii) for every edge $uv \in E(D)$ there exist $i \leq j$ such that $u \in W_i$ and $v \in W_j$. 

A directed path decomposition of $D$ is a sequence $W_1, W_2, \ldots, W_n$ such that

(i) $\bigcup W_i = V(D)$,
(ii) if $i < i' < i''$ then $W_i \cap W_{i''} \subseteq W_{i'}$,
(iii) for every edge $uv \in E(D)$ there exist $i \leq j$ such that $u \in W_i$ and $v \in W_j$.

The width of $W_1, \ldots, W_n$ is $\max\{|W_i| - 1 : 1 \leq i \leq n\}$

The directed path-width of $D$ is the minimum width of a directed path-decomposition.
A directed path decomposition of $D$ is a sequence $W_1, W_2, \ldots, W_n$ such that

(i) $\bigcup W_i = V(D)$,
(ii) if $i < i' < i''$ then $W_i \cap W_{i''} \subseteq W_{i'}$,
(iii) for every edge $uv \in E(D)$ there exist $i \leq j$ such that $u \in W_i$ and $v \in W_j$.

The width of $W_1, \ldots, W_n$ is $\max\{|W_i| - 1 : 1 \leq i \leq n\}$

The directed path-width of $D$ is the minimum width of a directed path-decomposition.

**CONJECTURE** Big directed path-width $\Rightarrow$ big cylindrical grid minor or a big binary tree minor with each edge replaced by two antiparallel edges.