Well-Quasi-Ordered Classes of Bounded Clique-Width

Maël Dumas and Aliaume Lopez
Bordeaux INP, LaBRI, France

LICS 2026, FLOC 2026
Lisbon, 2026-07-21

HI!

Well-Quasi-Orders in $\ll 180$ seconds

Let $(Q, \leq)$ be a quasi-order, i.e. a reflexive and transitive relation on Q.

Well-Quasi-Orders in ≪ 180 seconds

Let $(Q, \leq)$ be a quasi-order, i.e. a reflexive and transitive relation on Q.

$$\forall \quad x_0 \quad x_1 \quad \cdots \quad x_i \quad \cdots \quad x_j \quad \cdots$$

Q is a Well-Quasi-Order if.
For all sequences

Well-Quasi-Orders in $\ll$ 180 seconds

Let $(Q, \leq)$ be a quasi-order, i.e. a reflexive and transitive relation on Q.

$$\forall \quad x_0 \quad x_1 \quad \cdots \quad x_i \quad \cdots \quad x_j \quad \cdots$$

$$\exists \quad x_i \leq x_j$$

Q is a Well-Quasi-Order if.
For all sequences, there exists an increasing pair of elements.

Well-Quasi-Orders in $\ll$ 180 seconds

Let $(Q, \leq)$ be a quasi-order, i.e. a reflexive and transitive relation on Q.

$$\forall \quad x_0 \quad x_1 \quad \dots \quad x_i \quad \dots \quad x_j \quad \dots$$

$$\exists \quad x_i \leq x_j$$

Q is a Well-Quasi-Order if.
For all sequences, there exists an increasing pair of elements.

Examples.

- $(\mathbb{N}, \leq)$
- $(\mathbb{N}^2, \leq)$ [Dickson]
- $(\Sigma^*, \leq^*)$ [Higman]
- $(\mathcal{G}, \leq_{\text{minor}})$ [Robertson-Seymour]
- $\dots$

Well-Quasi-Orders in $\ll$ 180 seconds

Let $(Q, \leq)$ be a quasi-order, i.e. a reflexive and transitive relation on Q.

$$\forall \quad x_0 \qquad x_1 \qquad \cdots \qquad x_i \qquad \cdots \qquad x_j \qquad \cdots$$

$$\exists \quad x_i \leq x_j$$

Q is a Well-Quasi-Order if.
For all sequences, there exists an increasing pair of elements.

Examples.

- $(\mathbb{N}, \leq)$
- $(\mathbb{N}^2, \leq)$ [Dickson]
- $(\Sigma^*, \leq^*)$ [Higman]
- $(\mathcal{G}, \leq_{\text{minor}})$ [Robertson-Seymour]
- $\ldots$

Non-Examples.

- $(\mathbb{Z}, \leq)$
- $(\Sigma^*, \leq_{\text{prefix}})$
- $(\mathcal{G}, \leq_{\text{subgraph}})$
- $\ldots$

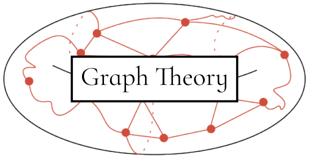
Well-Quasi-Orders in ≪ 180 seconds

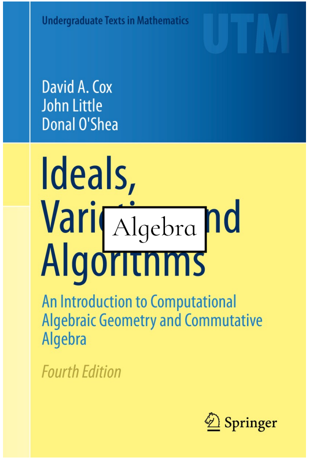
Let $(Q, \leq)$ be a quasi-order, i.e. a reflexive and transitive relation on Q.

$\forall \quad x_0 \quad x_1 \quad \dots \quad x_i \quad \dots \quad x_j$

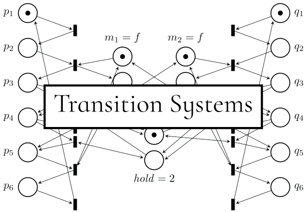
$\exists \quad x_i \leq x_j$

Q is a Well-Quasi-Order if:
For all sequences, there exists an increasing pai
of elements.





Examples.



— $(\mathbb{N}, \leq$
— $(\mathbb{N}^2,$
— $(\Sigma^*,$
— $(\mathcal{G}, \leq_{\text{minor}})$ [Robertson-Seymour]
— …

Non-Examples.

— $(\mathbb{Z}, \leq)$
— $(\Sigma^*, \leq_{\text{prefix}})$
— $(\mathcal{G}, \leq_{\text{subgraph}})$
— …

Which classes of graphs are WQO ?

The Meta Question.
Input : a class of graphs $\mathcal{C}$,
Output : is it well-quasi-ordered ?

Which classes of graphs are WQO ?

The Meta Question.
Input : a class of graphs $\mathcal{C}$,
Output : is it well-quasi-ordered ?

Classically, we ask different questions.
Graph Theory : find new well-quasi-orders on all graphs
Order Theory : use graphs to build new well-quasi-orders

Which classes of graphs are WQO ?

The Meta Question.
Input : a class of graphs $\mathcal{C}$,
Output : is it well-quasi-ordered ?

In this talk.
Order : induced subgraph relation,
Classes : hereditary (downwards closed) classes of graphs

Classically, we ask different questions.
Graph Theory : find new well-quasi-orders on all graphs
Order Theory : use graphs to build new well-quasi-orders

Which classes of graphs are WQO?

The Meta Question.
Input : a class of graphs $\mathcal{C}$,
Output : is it well-quasi-ordered?

In this talk.
Order : induced subgraph relation,
Classes : hereditary (downwards closed) classes of graphs

Classically, we ask different questions.
Graph Theory : find new well-quasi-orders on all graphs
Order Theory : use graphs to build new well-quasi-orders

Positive Results [Ding'92, folklore].
bounded tree-depth $\implies$ WQO,
bounded shrub-depth $\implies$ WQO,
m-partite cographs $\implies$ WQO

Which classes of graphs are WQO ?

The Meta Question.
Input : a class of graphs $\mathcal{C}$,
Output : is it well-quasi-ordered ?

In this talk.
Order : induced subgraph relation,
Classes : hereditary (downwards closed) classes of graphs

Classically, we ask different questions.
Graph Theory : find new well-quasi-orders on all graphs
Order Theory : use graphs to build new well-quasi-orders

Positive Results [Ding'92, folklore].
bounded tree-depth $\implies$ WQO,
bounded shrub-depth $\implies$ WQO,
m-partite cographs $\implies$ WQO

Negative Result [folklore].
"every" quasi-order is representable as a class of graphs under the induced subgraph relation

Which classes of graphs are WQO ?

The Meta Question.
Input : a class of graphs $\mathcal{C}$,
Output : is it well-quasi-or

In this talk.
Order : induced subgraph relation,
losed) classes of graphs

Classically, we ask differen
Graph Theory : find new w
Order Theory : use graphs

"Easier" Question.
Which classes of graphs can be used to build new well-quasi-orders ?

Higman's lemma $\rightsquigarrow$ labelling total orderings
Kruskal's theorem $\rightsquigarrow$ labelling trees
????????????????????? $\rightsquigarrow$ labelling graphs

Positive Results [Ding'92, fo
bounded tree-depth $\implies$ W
bounded shrub-depth $\implies$ WQO,
m-partite cographs $\implies$ WQO

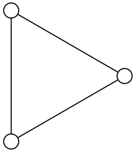
as a class of graphs under
the induced subgraph relation

Freely Labeled Graphs

$\mathsf{Label}_X(\mathcal{C})$ is the collection of all possible X-labellings of graphs in $\mathcal{C}$

Freely Labeled Graphs

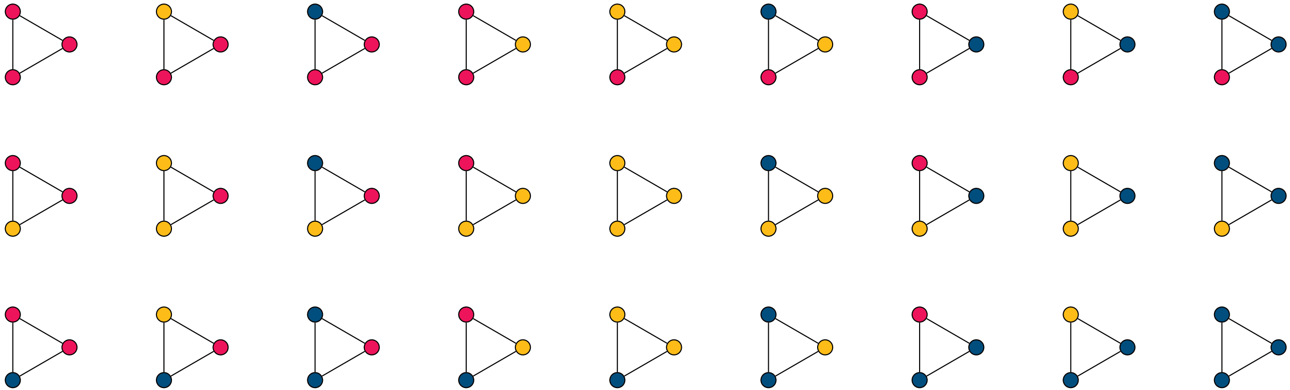
$\mathsf{Label}_X(\mathcal{C})$ is the collection of all possible X-labellings of graphs in $\mathcal{C}$



Freely Labeled Graphs

$\mathsf{Label}_X(\mathcal{C})$ is the collection of all possible X-labellings of graphs in $\mathcal{C}$

$\mathcal{C} = \{C_3\}$ and $X = \{\cdot, \cdot, \cdot\}$



Graph Theory

Is

$\mathcal{C}$

WQO ?

Order Theory

Is

$\mathcal{C}$

+ WQO set of labels

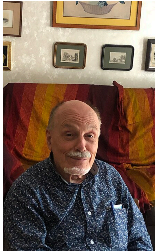
WQO ?

Graph Theory	Myself	Model Theory	Order Theory
Is $\mathcal{C}$ WQO?	Is $\mathcal{C}$ + 2 labels WQO?	Is $\mathcal{C}$ + finite set of labels WQO?	Is $\mathcal{C}$ + WQO set of labels WQO?

Graph Theory	Myself	Model Theory	Order Theory
Is	Is	Is	Is
$\mathcal{C}$	$\mathcal{C}$	$\mathcal{C}$	$\mathcal{C}$
	+ 2 labels	+ finite set of labels	+ WQO set of labels
WQO ?	WQO ?	WQO ?	WQO ?

Graph Theory	Myself	First Conjecture	Model Theory	Second Conjecture	Order Theory
Is $\mathcal{C}$ WQO ?	Is $\mathcal{C}$ + 2 labels WQO ?		Is $\mathcal{C}$ + finite set of labels WQO ?		Is $\mathcal{C}$ + WQO set of labels WQO ?

Graph Theory	Myself	First Conjecture	Model Theory	Second Conjecture	Order Theory
Is $\mathcal{C}$	Is $\mathcal{C}$ + 2 labels		Is $\mathcal{C}$ + finite set of labels		Is $\mathcal{C}$ + WQO set of labels
WQO ?	WQO ?		WQO ?		WQO ?



First Pouzet Conjecture : 2-WQO = ω-WQO

Second Pouzet Conjecture : ω-WQO = $\forall$-WQO

Schmitz's Conjecture : *finite paths* are the only obstruction to 2-WQO

Previously, in (hereditary) WQOs-on-Graphs...

Previously, in (hereditary) WQOs-on-Graphs...

For Monotone **Classes** [Ding'92].
2-WQO = ω-WQO = $\forall$-WQO
= no long paths
= bounded tree-depth

Previously, in (hereditary) WQOs-on-Graphs…

For Monotone **Classes** [Ding'92].
2-WQO = ω-WQO = $\forall$-WQO
= no long paths
= bounded tree-depth

For $\mathrm{NLC}^k_{\mathcal{F}}$ **Classes** [Daligault, Rao, Thomassé'10].
WQO = 2-WQO = ω-WQO = $\forall$-WQO
= no long paths
and is decidable!

Previously, in (hereditary) WQOs-on-Graphs...

For Monotone Classes [Ding'92].
2-WQO = ω-WQO = $\forall$-WQO
= no long paths
= bounded tree-depth

For $\mathrm{NLC}^k_{\mathcal{F}}$ Classes [Daligault, Rao, Thomassé'10].
WQO = 2-WQO = ω-WQO = $\forall$-WQO
= no long paths
and is decidable!

For bounded linear clique-width Classes [L.'25].
ω-WQO = $\forall$-WQO
and is decidable!

Previously, in (hereditary) WQOs-on-Graphs...

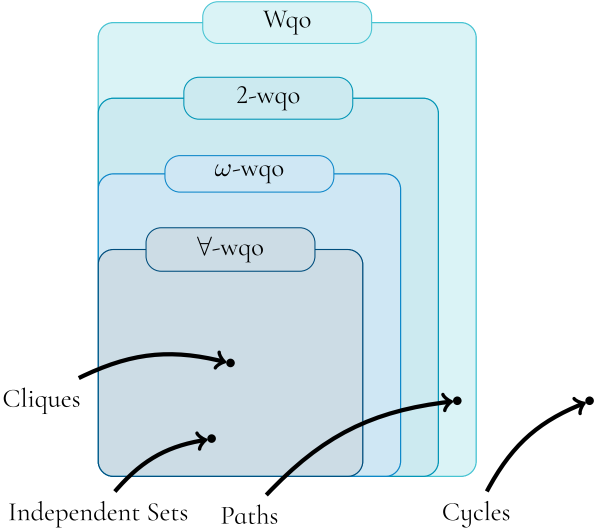
For Monotone **Classes** [Ding'92].
2-WQO = ω-WQO = $\forall$-WQO
= no long paths
= bounded tree-depth

For NLC$^k_{\mathcal{F}}$ **Classes** [Daligault, Rao, Thomassé'10].
WQO = 2-WQO = ω-WQO = $\forall$-WQO

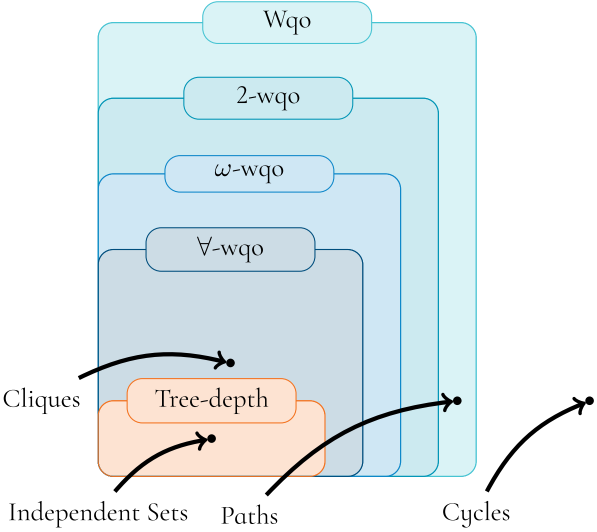
For bounded clique-width **Classes** [This Talk].
2-WQO = ω-WQO = $\forall$-WQO
= no $\exists$FO transduce paths and is decidable!

For bounded linear clique-width **Classes** [L.'25].
ω-WQO = $\forall$-WQO
and is decidable!

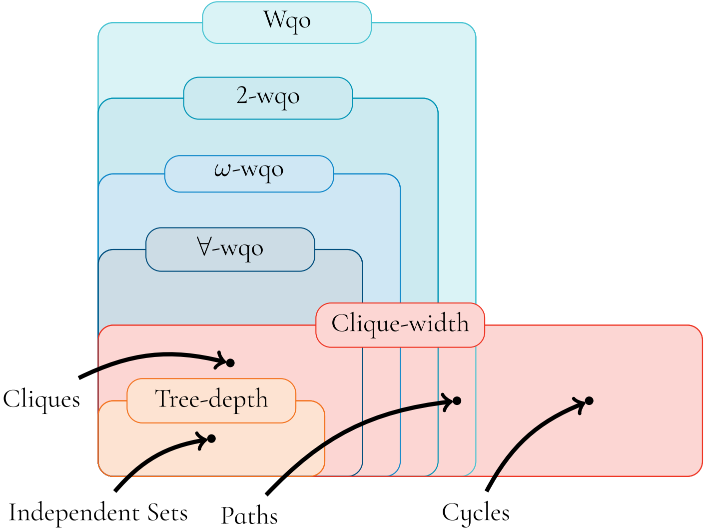
Examples



Examples


Wqo
2-wqo
ω-wqo
$\forall$-wqo
Tree-depth
Cliques
Independent Sets
Paths
Cycles

Examples


Wqo
2-wqo
ω-wqo
∀-wqo
Clique-width
Tree-depth
Cliques
Independent Sets
Paths
Cycles

I still did not define bounded clique-width...

I still did not define bounded clique-width...

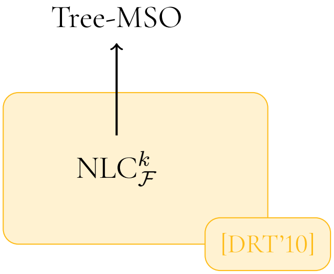
$\mathrm{NLC}^k_{\mathcal{F}}$ Classes.

$$
\begin{aligned}
G = {} & \bullet_i && (i \in [k]) \\
\mid {} & G \oplus_f G && (f \subseteq [k] \times [k]) \\
\mid {} & \rho(G) && (\rho : [k] \to [k], \rho \in \mathcal{F})
\end{aligned}
$$

$\mathrm{NLC}^k_{\mathcal{F}}$

[DRT'10]

I still did not define bounded clique-width...

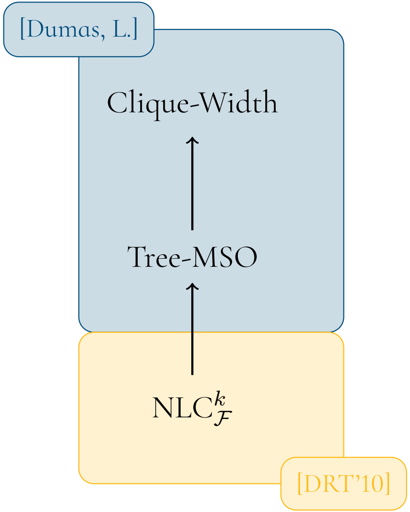


$\mathrm{NLC}^k_{\mathcal{F}}$ Classes.

$$
\begin{aligned}
G = {} & \bullet_i && (i \in [k]) \\
\mid {} & G \oplus_f G && (f \subseteq [k] \times [k]) \\
\mid {} & \rho(G) && (\rho : [k] \to [k], \rho \in \mathcal{F})
\end{aligned}
$$

Tree-MSO Classes.
$\mathcal{C} = \mathcal{I}(\mathsf{Tree}_\Sigma)$,
where $\mathcal{I}$ given by formulas $\phi_{\text{vertex}}(x), \phi_{\text{edge}}(x, y), \phi_{\text{domain}}$

I still did not define bounded clique-width...



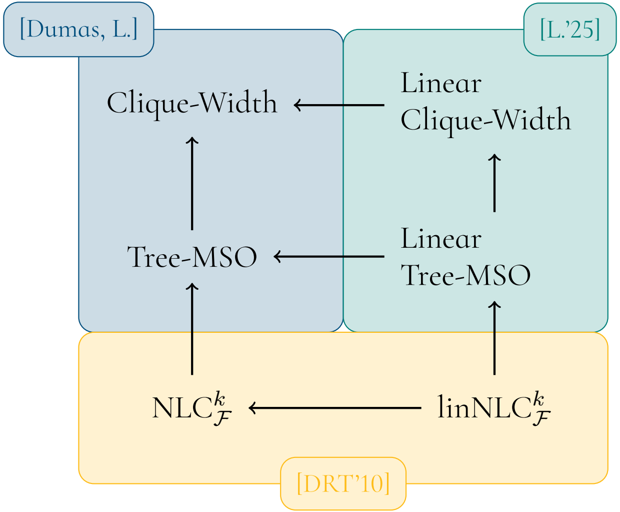
$\mathrm{NLC}^k_{\mathcal{F}}$ Classes.

$$\begin{aligned} G = {} & \bullet_i && (i \in [k]) \\ \mid {} & G \oplus_f G && (f \subseteq [k] \times [k]) \\ \mid {} & \rho(G) && (\rho : [k] \to [k], \rho \in \mathcal{F}) \end{aligned}$$

Tree-MSO Classes.
$\mathcal{C} = \mathcal{I}(\mathsf{Tree}_\Sigma)$,
where $\mathcal{I}$ given by formulas $\phi_{\text{vertex}}(x), \phi_{\text{edge}}(x, y), \phi_{\text{domain}}$

Clique-Width Classes.
$\mathcal{C} \subseteq \mathcal{D}$ for some Tree-MSO class $\mathcal{D}$

I still did not define bounded clique-width…



$\mathrm{NLC}^k_{\mathcal{F}}$ Classes.

$$\begin{aligned} G &= \bullet_i && (i \in [k]) \\ &\mid G \oplus_f G && (f \subseteq [k] \times [k]) \\ &\mid \rho(G) && (\rho : [k] \to [k], \rho \in \mathcal{F}) \end{aligned}$$

Tree-MSO Classes.
$\mathcal{C} = \mathcal{I}(\mathsf{Tree}_\Sigma)$,
where $\mathcal{I}$ given by formulas $\phi_{\text{vertex}}(x), \phi_{\text{edge}}(x, y), \phi_{\text{domain}}$

Clique-Width Classes.
$\mathcal{C} \subseteq \mathcal{D}$ for some Tree-MSO class $\mathcal{D}$

How to Prove such a Theorem?

Theorem 1. Assume hereditary and bounded clique-width.

$2\text{-WQO} = \omega\text{-WQO} = \forall\text{-WQO}$

$\oplus$ and with a total ordering

How to Prove such a Theorem ?

Theorem 1. Assume hereditary and bounded clique-width.

2-WQO = ω-WQO = $\forall$-WQO
$\oplus$ and with a total ordering

Theorem 2. Assume tree-mso.

2-WQO = ω-WQO = $\forall$-WQO
$\oplus$ and with a total ordering
$\oplus$ and it is decidable !

How to Prove such a Theorem ?

Theorem 1. Assume hereditary and bounded clique-width.

2-WQO = ω-WQO = $\forall$-WQO
$\oplus$ and with a total ordering

Theorem 2. Assume tree-mso.

2-WQO = ω-WQO = $\forall$-WQO
$\oplus$ and with a total ordering
$\oplus$ and it is decidable !

$$\text{Theorem 1} \Longleftarrow \text{Theorem 2}$$

(as opposed to [DRT'10])

How to Prove such a Theorem ?

$$\mathcal{I} \quad : \quad \mathsf{Tree}_\Sigma \xrightarrow{\text{surjective}} \mathcal{C}$$

Theorem 1. Assume hereditary and bounded clique-width.

2-WQO = ω-WQO = $\forall$-WQO

$\oplus$ and with a total ordering

Theorem 2. Assume tree-mso.

2-WQO = ω-WQO = $\forall$-WQO

$\oplus$ and with a total ordering

$\oplus$ and it is decidable !

Theorem 1 $\Longleftarrow$ Theorem 2

(as opposed to [DRT'10])

Proof Scheme of Theorem 2

How to Prove such a Theorem ?

$$\mathcal{I} \quad : \quad \mathsf{Tree}_\Sigma \xrightarrow{\text{surjective}} \mathcal{C}$$

Theorem 1. Assume hereditary and bounded clique-width.

2-WQO = ω-WQO = $\forall$-WQO

$\oplus$ and with a total ordering

Theorem 2. Assume tree-mso.

2-WQO = ω-WQO = $\forall$-WQO

$\oplus$ and with a total ordering

$\oplus$ and it is decidable!

Build an ordering on trees such that

(A) $\mathcal{I}$ is *order-preserving*

Theorem 1 $\Longleftarrow$ Theorem 2

(as opposed to [DRT'10])

Proof Scheme of Theorem 2

How to Prove such a Theorem ?

$$\mathcal{I} \quad : \quad \mathsf{Tree}_\Sigma \xrightarrow{\text{surjective}} \mathcal{C}$$

Theorem 1. Assume hereditary and bounded clique-width.

2-WQO = ω-WQO = $\forall$-WQO

$\oplus$ and with a total ordering

Theorem 2. Assume tree-mso.

2-WQO = ω-WQO = $\forall$-WQO

$\oplus$ and with a total ordering

$\oplus$ and it is decidable!

Build an ordering on trees such that

(A) $\mathcal{I}$ is *order-preserving*

(B) Tree_Σ is $\forall$-WQO

Theorem 1 $\Longleftarrow$ Theorem 2

(as opposed to [DRT'10])

Proof Scheme of Theorem 2

How to Prove such a Theorem ?

$$\mathcal{I} \quad : \quad \mathsf{Tree}_\Sigma \xrightarrow{\text{surjective}} \mathcal{C}$$

Theorem 1. Assume hereditary and bounded clique-width.

2-WQO = ω-WQO = $\forall$-WQO

$\oplus$ and with a total ordering

Theorem 2. Assume tree-mso.

2-WQO = ω-WQO = $\forall$-WQO

$\oplus$ and with a total ordering

$\oplus$ and it is decidable !

Build an ordering on trees such that

(A) $\mathcal{I}$ is *order-preserving*
(not too many relations)

(B) Tree_Σ is $\forall$-WQO
(a lot of relations)

Theorem 1 $\Longleftarrow$ Theorem 2

(as opposed to [DRT'10])

Proof Scheme of Theorem 2

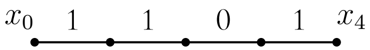
Automata Theory Strikes Back

Towards Monoids.
Using *automata magic*, we can reduce the problem to talk about *finite monoids* $(M, \cdot)$ instead of *formulas* φ. E.g. $(\mathbb{Z}/2\mathbb{Z}, +)$.

Automata Theory Strikes Back

Towards Monoids.
Using *automata magic*, we can reduce the problem to talk about *finite monoids* $(M, \cdot)$ instead of *formulas* φ. E.g. $(\mathbb{Z}/2\mathbb{Z}, +)$.

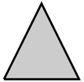
Monoid Labelled Trees.
Trees now have **edges** labelled by M, and we can consider $[\![x : y]\!]$ the *product* of the labels along the path from x to y.



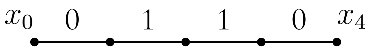
$$[\![x_0 : x_4]\!] = 1$$

Automata Theory Strikes Back

Towards Monoids.
Using *automata magic*, we can reduce the problem to talk about *finite monoids* $(M, \cdot)$ instead of *formulas* φ. E.g. $(\mathbb{Z}/2\mathbb{Z}, +)$.

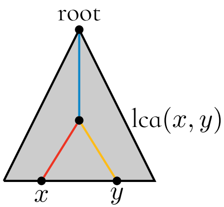
$\mathcal{I}:$  $\longmapsto \quad (\mathrm{leaves}(T), E)$

Monoid Labelled Trees.
Trees now have **edges** labelled by M, and we can consider $[\![x : y]\!]$ the *product* of the labels along the path from x to y.



$$[\![x_0 : x_4]\!] = 0$$

How to Build Edges.
$E(x, y)$ is determined using the *values* of paths in the tree starting from x and y



Forward Ramseyan Splits

Recall.

We search an ordering on trees s.t.

(A) $\mathcal{I}$ is *order-preserving* ($\ominus$ relations)

(B) Tree_Σ is $\forall$-WQO ($\oplus$ relations)

Forward Ramseyan Splits

Recall.
We search an ordering on trees s.t.
(A) $\mathcal{I}$ is *order-preserving* ($\ominus$ relations)
(B) Tree_Σ is $\forall$-WQO ($\oplus$ relations)

Attempt 0.
$T \leq_{\text{comp}} T'$ if there is a map $h \colon T \to T'$ preserving
- the root and leaves
- least common ancestors
- the *values* of paths in the tree.

Forward Ramseyan Splits

Recall.
We search an ordering on trees s.t.
(A) $\mathcal{I}$ is *order-preserving* ($\ominus$ relations)
(B) Tree_Σ is $\forall$-WQO ($\oplus$ relations)

Attempt 0.
$T \leq_{\text{comp}} T'$ if there is a map $h \colon T \to T'$ preserving
- the root and leaves
- least common ancestors
- the *values* of paths in the tree.

(A) ✓ (B) ✗

Forward Ramseyan Splits

Recall.
We search an ordering on trees s.t.
(A) $\mathcal{I}$ is *order-preserving* ($\ominus$ relations)
(B) Tree_Σ is $\forall$-WQO ($\oplus$ relations)

Attempt 0.
$T \leq_{\text{comp}} T'$ if there is a map $h \colon T \to T'$ preserving
- the root and leaves
- least common ancestors
- the *values* of paths in the tree.

(A) ✓ (B) ✗

"Fun" remark.
Requiring (B) to hold is *exactly* [DRT'10]...

Forward Ramseyan Splits

Recall.
We search an ordering on trees s.t.
(A) $\mathcal{I}$ is *order-preserving* ($\ominus$ relations)
(B) Tree_Σ is $\forall$-WQO ($\oplus$ relations)

Attempt 0.
$T \leq_{\text{comp}} T'$ if there is a map $h \colon T \to T'$ preserving
- the root and leaves
- least common ancestors
- the *values* of paths in the tree.

(A) ✓ (B) ✗

"Fun" remark.
Requiring (B) to hold is *exactly* [DRT'10]...

More Automata Magic [Colcombet'07].
We can enrich the trees T with a *forward Ramseyan split* $\mathfrak{s} \colon T \to [n]$ allowing us to **short circuit** computations.

Forward Ramseyan Splits

Recall.
We search an ordering on trees s.t.
(A) $\mathcal{I}$ is *order-preserving* ($\ominus$ relations)
(B) Tree_Σ is $\forall$-WQO ($\oplus$ relations)

More Automata Magic [Colcombet'07].
We can enrich the trees T with a *forward Ramseyan split* $\mathfrak{s}\colon T \to [n]$ allowing us to **short circuit** computations.

Attempt 0.
$T \leq_{\text{comp}} T'$ if there is a map $h\colon T \to T'$ preserving
- the root and leaves
- least common ancestors
- the *values* of paths in the tree.

(A) ✓ (B) ✗

$$[\![x : y]\!] = [\![x : z_1]\!] \cdot [\![z_1 : z_3]\!] \cdot [\![z_3 : y]\!]$$

$$x \longrightarrow z_1 \longrightarrow z_2 \longrightarrow \cdots \longrightarrow z_3 \longrightarrow y$$

"Fun" remark.
Requiring (B) to hold is *exactly* [DRT'10]...

Forward Ramseyan Splits

Recall.
We search an ordering on trees s.t.
(A) $\mathcal{I}$ is *order-preserving* ($\ominus$ relations)
(B) Tree_Σ is $\forall$-WQO ($\oplus$ relations)

More Automata Magic [Colcombet'07].
We can enrich the trees T with a *forward Ramseyan split* $\mathfrak{s}\colon T \to [n]$ allowing us to **short circuit** computations.

Attempt 0.
$T \leq_{\text{comp}} T'$ if there is a map $h\colon T \to T'$ preserving
- the root and leaves
- least common ancestors
- the *values* of paths in the tree.

(A) ✓ (B) ✗

$$[\![x : y]\!] = [\![x : z_1]\!] \cdot [\![z_1 : z_3]\!] \cdot [\![z_3 : y]\!]$$

$$x \longrightarrow z_1 \longrightarrow z_2 \longrightarrow \cdots \longrightarrow z_3 \longrightarrow y$$

$\mathfrak{s}:$	$> k$	k	$> k$	k	$\geq k$	k	$> k$

"Fun" remark.
Requiring (B) to hold is *exactly* [DRT'10]...

Forward Ramseyan Splits

Recall.
We search an ordering on trees s.t.
(A) $\mathcal{I}$ is *order-preserving* ($\ominus$ relations)
(B) Tree_Σ is $\forall$-WQO ($\oplus$ relations)

More Automata Magic [Colcombet'07].
We can enrich the trees T with a *forward Ramseyan split* $\mathfrak{s} \colon T \to [n]$ allowing us to **short circuit** computations.

Attempt 0.
$T \leq_{\text{comp}} T'$ if there is a map $h \colon T \to T'$ preserving
- the root and leaves
- least common ancestors
- the *values* of paths in the tree.

(A) ✓ (B) ✗

$[\![z_1 : z_3]\!] = [\![z_1 : z_2]\!]$ (short-circuiting)

$$[\![x : y]\!] = [\![x : z_1]\!] \cdot [\![z_1 : z_3]\!] \cdot [\![z_3 : y]\!]$$

$$x \longrightarrow z_1 \longrightarrow z_2 \longrightarrow \cdots \longrightarrow z_3 \longrightarrow y$$

$\mathfrak{s}:$	$> k$	k	$> k$	k	$\geq k$	k	$> k$

"Fun" remark.
Requiring (B) to hold is *exactly* [DRT'10]...

The Gap Embedding

Recall.
We search an ordering on trees s.t.
(A) $\mathcal{I}$ is *order-preserving* ($\ominus$ relations)
(B) Tree_Σ is $\forall$-WQO ($\oplus$ relations)

$$\ominus \text{ relations} \longleftrightarrow \oplus \text{ relations}$$

$\leq_{\mathrm{comp}}$

(A) ✓
(B) ✗

The Gap Embedding

Recall.
We search an ordering on trees s.t.
(A) $\mathcal{I}$ is *order-preserving* ($\ominus$ relations)
(B) Tree_Σ is $\forall$-WQO ($\oplus$ relations)

Theorem [Dershowitz, Tzameret'03].
Trees with the *gap embedding* (respecting splits) are $\forall$-WQO!

$$\ominus \text{ relations} \xleftrightarrow{\quad \leq_{\text{gap}} \qquad\qquad\qquad \leq_{\text{comp}} \quad} \oplus \text{ relations}$$

$\leq_{\text{gap}}$: (A) ✗ (B) ✓

$\leq_{\text{comp}}$: (A) ✓ (B) ✗

The Gap Embedding

Recall.
We search an ordering on trees s.t.
(A) $\mathcal{I}$ is *order-preserving* ($\ominus$ relations)
(B) Tree_{Σ} is $\forall$-WQO ($\oplus$ relations)

Theorem [Dershowitz, Tzameret'03].
Trees with the *gap embedding* (respecting splits) are $\forall$-WQO!

$\ominus$ relations $\longleftrightarrow$ $\oplus$ relations

$\leq_{\text{gap}}$	$\leq_{\text{super-nice}}$	$\leq_{\text{comp}}$
(A) ✗	(A) ✓?	(A) ✓
(B) ✓	(B) ✓?	(B) ✗

The Good, The Bad, and the Ugly Boughs

$$\llbracket x : y \rrbracket = \llbracket x : z_1 \rrbracket \cdot \llbracket z_1 : z_3 \rrbracket \cdot \llbracket z_3 : y \rrbracket$$

$$x \longrightarrow z_1 \longrightarrow z_2 \longrightarrow \cdots \longrightarrow z_3 \longrightarrow y$$

The Good, The Bad, and the Ugly Boughs

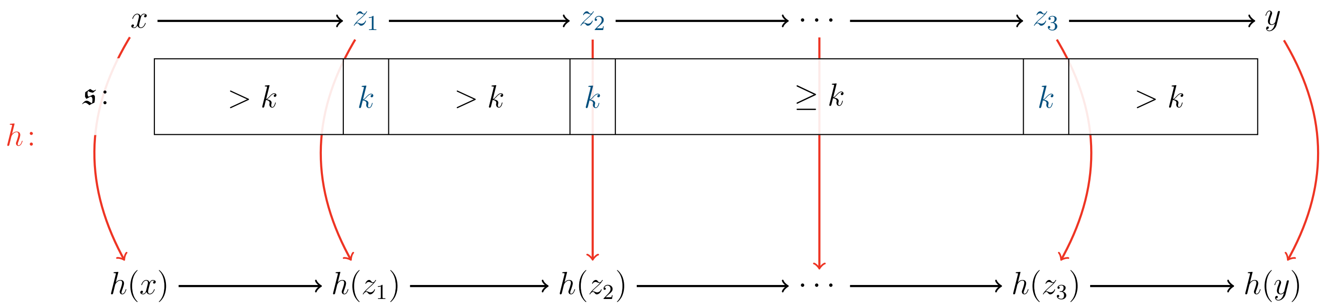
$$[\![x : y]\!] = [\![x : z_1]\!] \cdot [\![z_1 : z_3]\!] \cdot [\![z_3 : y]\!]$$

$$h : \begin{tikzcd} x \arrow[r] \arrow[d, bend right] & z_1 \arrow[r] \arrow[d, bend right] & z_2 \arrow[r] \arrow[d] & \cdots \arrow[r] \arrow[d] & z_3 \arrow[r] \arrow[d, bend left] & y \arrow[d, bend left] \\ h(x) \arrow[r] & h(z_1) \arrow[r] & h(z_2) \arrow[r] & \cdots \arrow[r] & h(z_3) \arrow[r] & h(y) \end{tikzcd}$$

GOAL : $[\![x : y]\!] = [\![h(x) : h(y)]\!]$

The Good, The Bad, and the Ugly Boughs

$$[\![x : y]\!] = [\![x : z_1]\!] \cdot [\![z_1 : z_3]\!] \cdot [\![z_3 : y]\!]$$

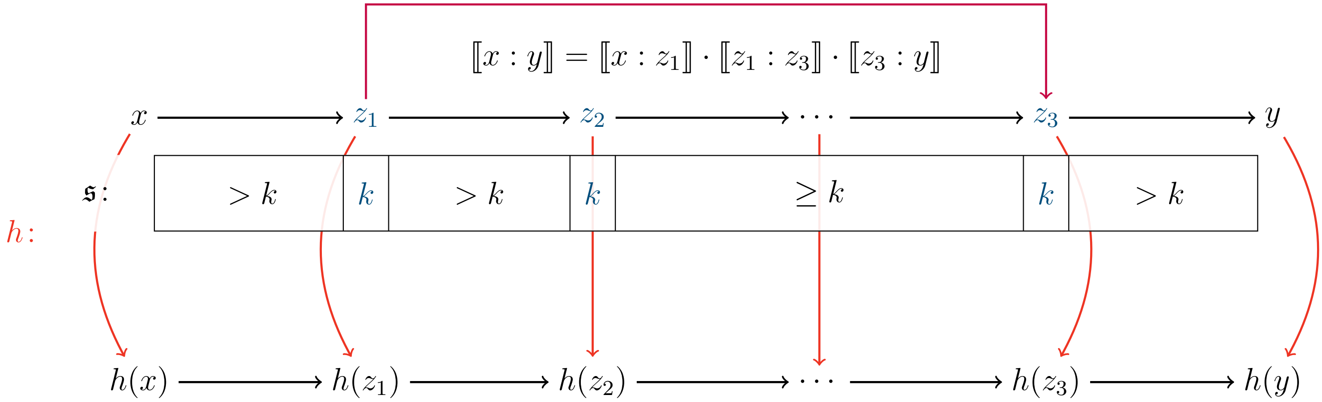


GOAL : $[\![x : y]\!] = [\![h(x) : h(y)]\!]$

The Good, The Bad, and the Ugly Boughs

$[\![z_1 : z_3]\!] = [\![z_1 : z_2]\!]$ (short-circuiting)

$$[\![x : y]\!] = [\![x : z_1]\!] \cdot [\![z_1 : z_3]\!] \cdot [\![z_3 : y]\!]$$

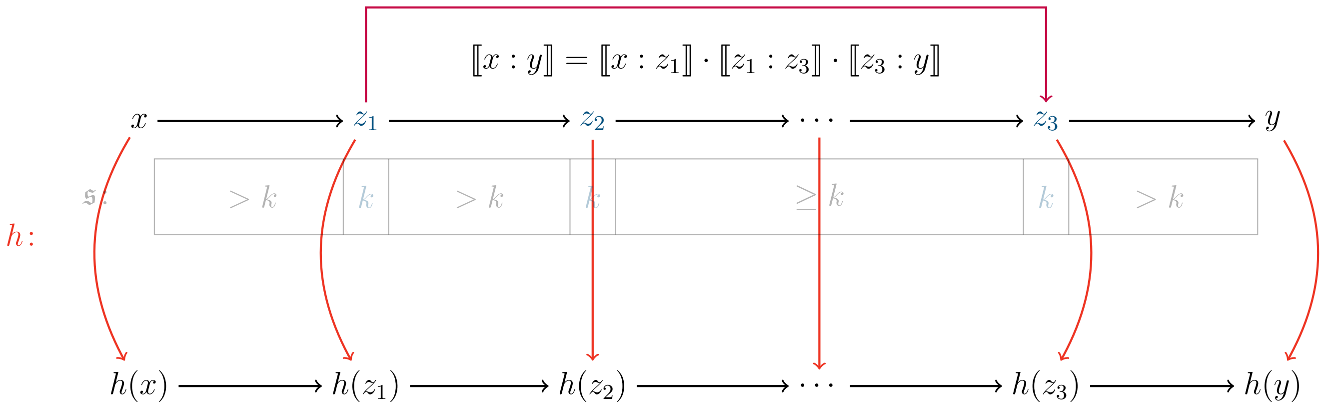


GOAL : $[\![x : y]\!] = [\![h(x) : h(y)]\!]$

The Good, The Bad, and the Ugly Boughs

$[\![z_1 : z_3]\!] = [\![z_1 : z_2]\!]$ (short-circuiting)

$$[\![x : y]\!] = [\![x : z_1]\!] \cdot [\![z_1 : z_3]\!] \cdot [\![z_3 : y]\!]$$

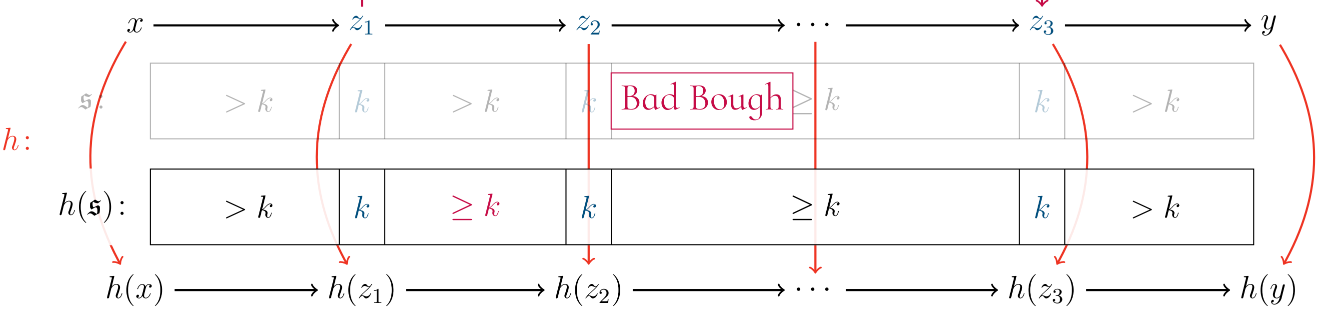


GOAL : $[\![x : y]\!] = [\![h(x) : h(y)]\!]$

The Good, The Bad, and the Ugly Boughs

$[\![z_1 : z_3]\!] = [\![z_1 : z_2]\!]$ (short-circuiting)

$$[\![x : y]\!] = [\![x : z_1]\!] \cdot [\![z_1 : z_3]\!] \cdot [\![z_3 : y]\!]$$

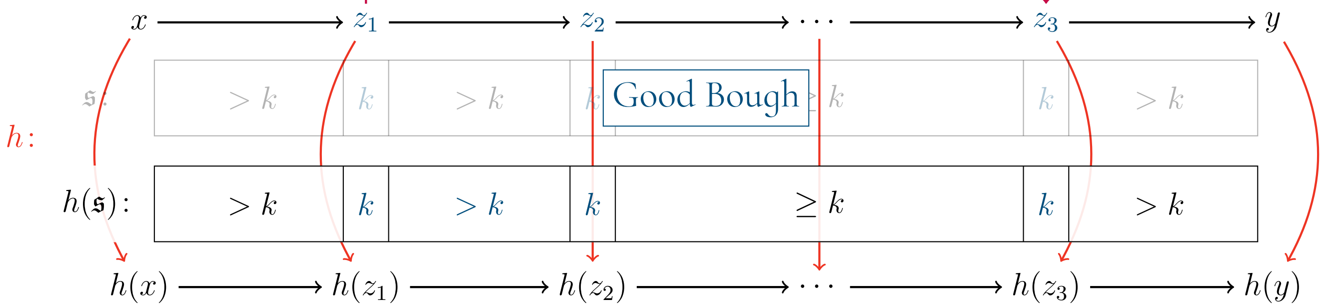


GOAL : $[\![x : y]\!] = [\![h(x) : h(y)]\!]$

The Good, The Bad, and the Ugly Boughs

$$[\![z_1 : z_3]\!] = [\![z_1 : z_2]\!] \text{ (short-circuiting)}$$

$$[\![x : y]\!] = [\![x : z_1]\!] \cdot [\![z_1 : z_3]\!] \cdot [\![z_3 : y]\!]$$



$$\text{GOAL} : [\![x : y]\!] = [\![h(x) : h(y)]\!]$$

The Good, The Bad, and the Ugly Boughs

$[\![z_1 : z_3]\!] = [\![z_1 : z_2]\!]$ (short-circuiting)

$x \longrightarrow y$

$\mathfrak{s} :$ k

$h :$

$h(\mathfrak{s}) :$ k

Key Technical Argument.
∞ bad boughs $\implies$ not 2-WQO
finitely bad boughs $\implies$ *order-preserving* ($\implies$ $\forall$-WQO)
(and it is decidable whether there are finitely many bad boughs)

$$h(x) \longrightarrow h(z_1) \longrightarrow h(z_2) \longrightarrow \cdots \longrightarrow h(z_3) \longrightarrow h(y)$$

GOAL : $[\![x : y]\!] = [\![h(x) : h(y)]\!]$

Conclusion

Theorem 1.
$\mathcal{C}$ *hereditary* + *bounded clique-width* :
1. $\mathcal{C}$ 2-WQO
2. $\mathcal{C}$ $\forall$-WQO

+ total ordering!

Theorem 2.
$\mathcal{C}$ Tree-MSO :
1. $\mathcal{C}$ is 2-WQO
2. $\mathcal{C}$ is $\forall$-WQO

+ total ordering! **decidable**!

Theorem 3.
$\mathcal{C}$ *hereditary* + *bounded clique-width* :
1. $\mathcal{C}$ **not** 2-WQO
2. $\exists \mathcal{D} \subseteq \mathcal{C}$ *bounded* ***linear*** *clique-width* + not 2-WQO
3. Periodic Antichain
4. $\mathcal{C}$ existentially *transduces* paths

Conclusion

Theorem 1.
$\mathcal{C}$ *hereditary* + *bounded clique-width* :
1. $\mathcal{C}$ 2-WQO
2. $\mathcal{C}$ $\forall$-WQO

+ total ordering!

Theorem 3.
$\mathcal{C}$ *hereditary* + *bounded clique-width* :
1. $\mathcal{C}$ **not** 2-WQO
2. $\exists\mathcal{D} \subseteq \mathcal{C}$ *bounded* ***linear*** *clique-width* + not 2-WQO
3. Periodic Antichain
4. $\mathcal{C}$ existentially *transduces* paths

Theorem 2.
$\mathcal{C}$ Tree-MSO :
1. $\mathcal{C}$ is 2-WQO
2. $\mathcal{C}$ is $\forall$-WQO

+ total ordering! **decidable**!

Future Work.
1. All *hereditary* classes?
2. *interpret* paths?
3. Structural characterisation?
4. Beyond graphs?
5. WQOs?

Conclusion

Theorem 1.
$\mathcal{C}$ *hereditary* + *bounded clique-width* :
1. $\mathcal{C}$ 2-WQO
2. $\mathcal{C}$ $\forall$-WQO

+ total ordering!

Theorem 2.
$\mathcal{C}$ Tree-MSO :
1. $\mathcal{C}$ is 2-WQO
2. $\mathcal{C}$ is $\forall$-WQO

+ total ordering! **decidable**!

Theorem 3.
$\mathcal{C}$ *hereditary* + *bounded clique-width* :
1. $\mathcal{C}$ **not** 2-WQO
2. $\exists \mathcal{D} \subseteq \mathcal{C}$ *bounded* ***linear*** *clique-width* + not 2-WQO
3. Periodic Antichain
4. $\mathcal{C}$ existentially *transduces* paths

Future Work.
1. ~~All *hereditary* classes?~~

2-WQO $\Longrightarrow$ bounded clique width
[Duron, Mählmann, Toruńczyk]

2. *interpret* paths?
3. Structural characterisation?
4. Beyond graphs? ...