Graph decompositions and well-quasi-ordering

Jean-Florent Raymond

LIRMM, University of Montpellier, France, and
MIMUW, University of Warsaw, Poland

Orléans, November 2015

Well-quasi-ordering

A WQO is an order where:

- every decreasing sequence is finite;
- every sequence of non-comparable elements is finite.

Well-quasi-ordering

A WQO is an order where:

- every decreasing sequence is finite;
- every sequence of non-comparable elements is finite.

Examples:

- $1, \frac{1}{2}, \frac{1}{3}, \dots$ is an infinite decreasing sequence wrt. $\leqslant$
  $\rightarrow (\mathbb{Q}, \leqslant)$ is not a WQO;

Well-quasi-ordering

A WQO is an order where:

- every decreasing sequence is finite;
- every sequence of non-comparable elements is finite.

Examples:

- $1, \frac{1}{2}, \frac{1}{3}, \ldots$ is an infinite decreasing sequence wrt. $\leqslant$
  $\rightarrow (\mathbb{Q}, \leqslant)$ is not a WQO;
- $\{0\}, \{1\}, \{2\}, \ldots$ is an infinite antichain wrt. $\subseteq$
  $\rightarrow (\mathcal{P}(\mathbb{N}), \subseteq)$ is not a WQO;

Well-quasi-ordering

A WQO is an order where:

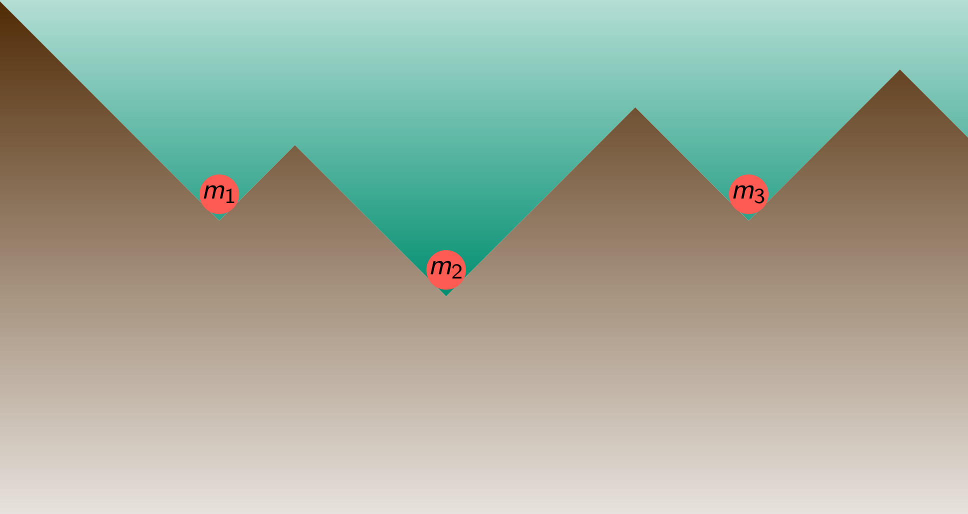
- every decreasing sequence is finite;
- every sequence of non-comparable elements is finite.

Examples:

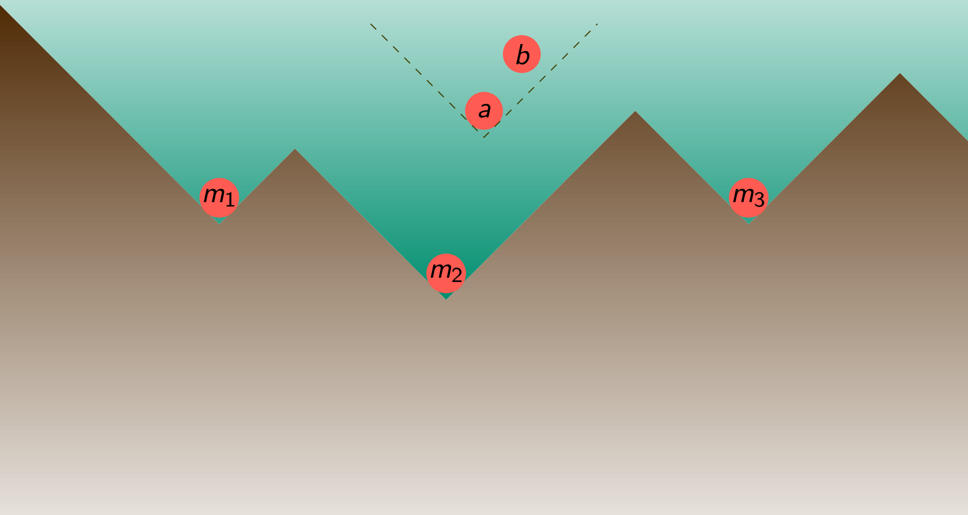
- $1, \frac{1}{2}, \frac{1}{3}, \dots$ is an infinite decreasing sequence wrt. $\leqslant$
  $\to (\mathbb{Q}, \leqslant)$ is not a WQO;
- $\{0\}, \{1\}, \{2\}, \dots$ is an infinite antichain wrt. $\subseteq$
  $\to (\mathcal{P}(\mathbb{N}), \subseteq)$ is not a WQO;
- $(A^\star, \leqslant_{\mathrm{subseq}})$ with A finite: WQO;
- $(\text{graphs}, \leqslant_{\mathrm{minor}})$: WQO.

Why do we like well-quasi-orders?

Upwards closed classes have a finite number of minimal elements.



Upwards closed classes have a finite number of minimal elements.

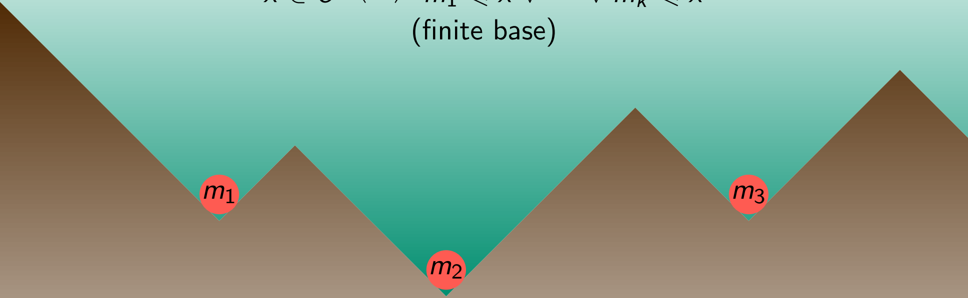


Why do we like well-quasi-orders?

Upwards closed classes have a finite number of minimal elements.

$$x \in U \iff m_1 \leqslant x \vee \cdots \vee m_k \leqslant x$$

(finite base)



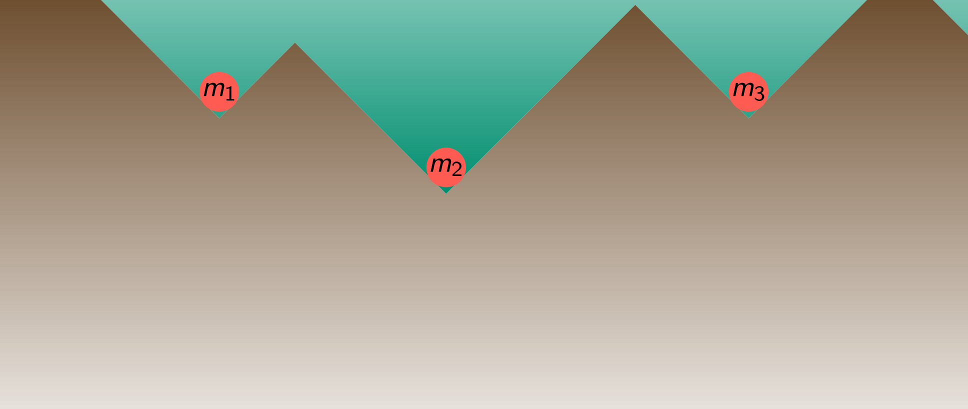
Why do we like well-quasi-orders?

Upwards closed classes have a finite number of minimal elements.

$$x \in U \iff m_1 \leqslant x \vee \cdots \vee m_k \leqslant x$$

(finite base)

Membership testing can be done in a finite number of checks.



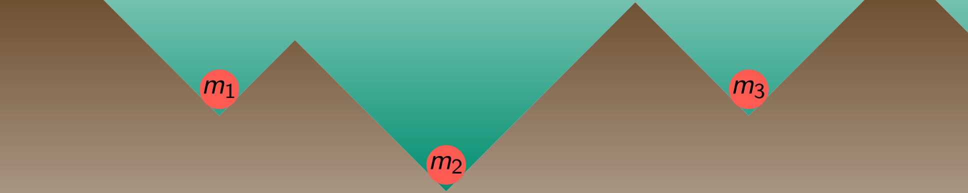
Why do we like well-quasi-orders?

Upwards closed classes have a finite number of minimal elements.

$$x \in U \iff m_1 \leqslant x \vee \cdots \vee m_k \leqslant x$$

(finite base)

Membership testing can be done in a finite number of checks.



Downwards closed classes have a finite number of obstructions.

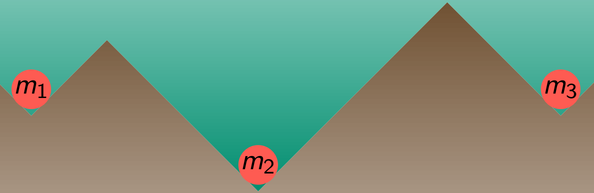
Why do we like well-quasi-orders?

Upwards closed classes have a finite number of minimal elements.

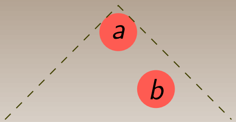
$$x \in U \iff m_1 \leqslant x \vee \cdots \vee m_k \leqslant x$$

(finite base)

Membership testing can be done in a finite number of checks.



Downwards closed classes have a finite number of obstructions.



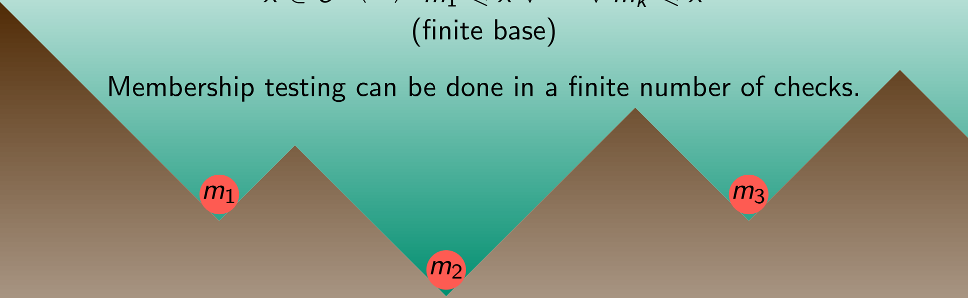
Why do we like well-quasi-orders?

Upwards closed classes have a finite number of minimal elements.

$$x \in U \iff m_1 \leqslant x \vee \cdots \vee m_k \leqslant x$$

(finite base)

Membership testing can be done in a finite number of checks.



Downwards closed classes have a finite number of obstructions.

$$x \in D \iff m_1 \nleqslant x \wedge \cdots \wedge m_k \nleqslant x$$

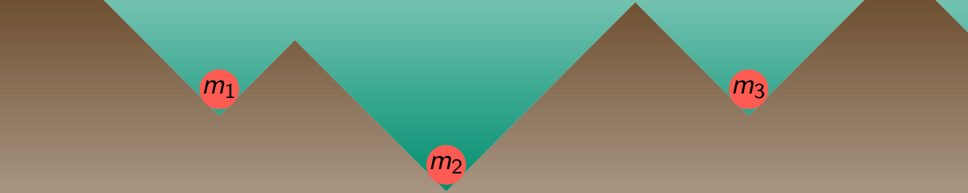
Why do we like well-quasi-orders?

Upwards closed classes have a finite number of minimal elements.

$$x \in U \iff m_1 \leqslant x \vee \cdots \vee m_k \leqslant x$$

(finite base)

Membership testing can be done in a finite number of checks.



Downwards closed classes have a finite number of obstructions.

$$x \in D \iff m_1 \not\leqslant x \wedge \cdots \wedge m_k \not\leqslant x$$

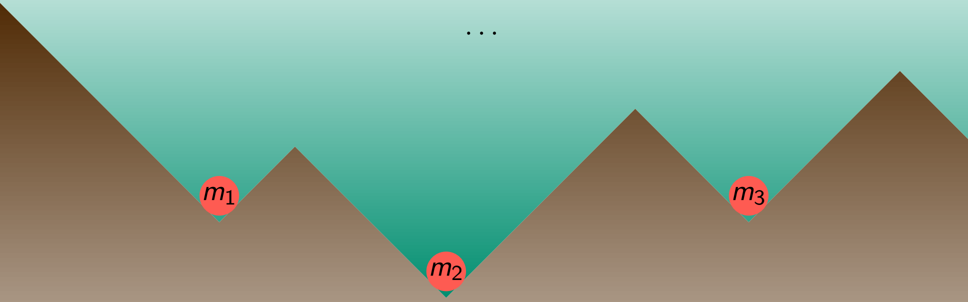
Membership testing can be done in a finite number of checks.

Why do we like well-quasi-orders?

graphs of genus $\geqslant g$ + minor relation

$(\geqslant k)$-colorable graphs + induced subgraph relation

...



graphs of treewidth $\leqslant k$ + minor relation

trees + contraction relation

...

In this talk...

$(S, \leqslant) = (\text{graph class}, \text{graph containment relation})$

In this talk...

$(S, \leqslant) =$ (graph class, graph containment relation)

- contraction relation $\leqslant_{\mathrm{ctr}}$: E contraction;
- induced minor relation $\leqslant_{\mathrm{im}}$: V deletion and E contraction;
- minor relation $\leqslant_{\mathrm{m}}$: V and E deletion, E contraction.

Containment relations on graphs

- Some are WQO:
  - minors: [GMXX] (Wagner's conjecture)
  - immersions: [GMXXIII] (Nash-Williams' Conjecture)

Containment relations on graphs

- Some are WQO:
  - minors: [GMXX] (Wagner's conjecture)
  - immersions: [GMXXIII] (Nash-Williams' Conjecture)
- ...some are not:
  - (induced) subgraphs
  - contractions
  - induced minors
  - (induced) topological minor

Containment relations on graphs

- Some are WQO:
  - minors: [GMXX] (Wagner's conjecture)
  - immersions: [GMXXIII] (Nash-Williams' Conjecture)
- ...some are not:
  - (induced) subgraphs
  - contractions
  - induced minors
  - (induced) topological minor
- ...and some are still open:
  - induced immersions
  - strong immersions

Containment relations on graphs

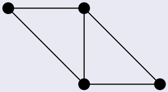
- Some are WQO:
  - minors: [GMXX] (Wagner's conjecture)
  - immersions: [GMXXIII] (Nash-Williams' Conjecture)
- ...some are not:
  - (induced) subgraphs
  - contractions
  - induced minors
  - (induced) topological minor
- ...and some are still open:
  - induced immersions
  - strong immersions

Main message: decomposition results (sometimes) imply WQO.

From structure to WQO

Toy example

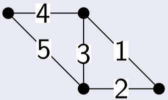
Subdivisions of H are WQO by contraction.

$H =$ 

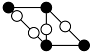
From structure to WQO

Toy example

Subdivisions of H are WQO by contraction.

$H =$ 

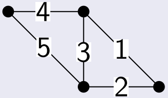
1. choose an encoding of graphs as tuples (for instance)
   e.g. # of subdivisions for each edge, in some chosen order;

$\text{enc}($  $) = (1, 0, 1, 0, 2)$

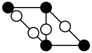
From structure to WQO

Toy example

Subdivisions of H are WQO by contraction.

$H =$ 

1. choose an encoding of graphs as tuples (for instance)
   e.g. # of subdivisions for each edge, in some chosen order;

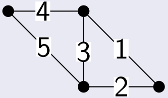
   enc() $= (1, 0, 1, 0, 2)$

2. choose an order on tuples s.t. $\mathrm{enc}(G) \leqslant \mathrm{enc}(G') \Rightarrow G \leqslant_{\mathrm{ctr}} G'$
   e.g. the product order, $(2, 1, 0, 3, 1) \leqslant (5, 1, 2, 4, 1)$

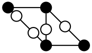
From structure to WQO

Toy example

Subdivisions of H are WQO by contraction.

$H =$ 

1. choose an encoding of graphs as tuples (for instance)
   e.g. # of subdivisions for each edge, in some chosen order;

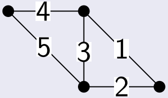
   enc() $= (1, 0, 1, 0, 2)$

2. choose an order on tuples s.t. $\mathrm{enc}(G) \leqslant \mathrm{enc}(G') \Rightarrow G \leqslant_{\mathrm{ctr}} G'$
   e.g. the product order, $(2, 1, 0, 3, 1) \leqslant (5, 1, 2, 4, 1)$
3. show that encodings are WQO by this order;

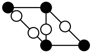
From structure to WQO

Toy example

Subdivisions of H are WQO by contraction.

$H =$ 

1. choose an encoding of graphs as tuples (for instance)
   e.g. # of subdivisions for each edge, in some chosen order;

   enc() $= (1, 0, 1, 0, 2)$

2. choose an order on tuples s.t. $\mathrm{enc}(G) \leqslant \mathrm{enc}(G') \Rightarrow G \leqslant_{\mathrm{ctr}} G'$
   e.g. the product order, $(2, 1, 0, 3, 1) \leqslant (5, 1, 2, 4, 1)$
3. show that encodings are WQO by this order;
4. that's it!
   antichain $\{G_1, G_2, \dots\} \Rightarrow$ antichain $\{\mathrm{enc}(G_1), \mathrm{enc}(G_2), \dots\}$

High level view

Goal: show that $(\mathcal{G}, \leqslant)$ is a WQO.

High level view

Goal: show that $(\mathcal{G}, \leqslant)$ is a WQO.

1. choose a mapping $enc\colon \mathcal{G} \to \mathcal{S}$ where $\mathcal{S}$ is simple;

High level view

Goal: show that $(\mathcal{G}, \leqslant)$ is a WQO.

1. choose a mapping $enc \colon \mathcal{G} \to \mathcal{S}$ where $\mathcal{S}$ is simple;
2. choose an order $\preceq_{\mathcal{S}}$ on $\mathcal{S}$ s.t. $\mathrm{enc}(G) \preceq_{\mathcal{S}} \mathrm{enc}(G') \Rightarrow G \preceq G'$;

High level view

Goal: show that $(\mathcal{G}, \leqslant)$ is a WQO.

1. choose a mapping $enc\colon \mathcal{G} \to \mathcal{S}$ where $\mathcal{S}$ is simple;
2. choose an order $\preceq_{\mathcal{S}}$ on $\mathcal{S}$ s.t. $\mathrm{enc}(G) \preceq_{\mathcal{S}} \mathrm{enc}(G') \Rightarrow G \preceq G'$;
3. show that $(\mathcal{S}, \preceq_{\mathcal{S}})$ is a WQO.

High level view

Goal: show that $(\mathcal{G}, \leqslant)$ is a WQO.

1. choose a mapping $enc\colon \mathcal{G} \to \mathcal{S}$ where $\mathcal{S}$ is simple;
2. choose an order $\preceq_{\mathcal{S}}$ on $\mathcal{S}$ s.t. $\mathrm{enc}(G) \preceq_{\mathcal{S}} \mathrm{enc}(G') \Rightarrow G \preceq G'$;
3. show that $(\mathcal{S}, \preceq_{\mathcal{S}})$ is a WQO.

In other words: order-preserving functions send WQOs on WQOs.

High level view

Goal: show that $(\mathcal{G}, \leqslant)$ is a WQO.

1. choose a mapping $enc \colon \mathcal{G} \to \mathcal{S}$ where $\mathcal{S}$ is simple;
2. choose an order $\preceq_{\mathcal{S}}$ on $\mathcal{S}$ s.t. $\mathrm{enc}(G) \preceq_{\mathcal{S}} \mathrm{enc}(G') \Rightarrow G \preceq G'$;
3. show that $(\mathcal{S}, \preceq_{\mathcal{S}})$ is a WQO.

In other words: order-preserving functions send WQOs on WQOs.

Choice of $\mathcal{S}$:

- given by a decomposition;

High level view

Goal: show that $(\mathcal{G}, \leqslant)$ is a WQO.

1. choose a mapping $enc\colon \mathcal{G} \to \mathcal{S}$ where $\mathcal{S}$ is simple;
2. choose an order $\preceq_{\mathcal{S}}$ on $\mathcal{S}$ s.t. $\mathrm{enc}(G) \preceq_{\mathcal{S}} \mathrm{enc}(G') \Rightarrow G \preceq G'$;
3. show that $(\mathcal{S}, \preceq_{\mathcal{S}})$ is a WQO.

In other words: order-preserving functions send WQOs on WQOs.

Choice of $\mathcal{S}$:

- given by a decomposition;
- $\mathcal{S} =$ (labeled) 2-c graphs of $\mathcal{G}$;

High level view

Goal: show that $(\mathcal{G}, \leqslant)$ is a WQO.

1. choose a mapping $enc\colon \mathcal{G} \to \mathcal{S}$ where $\mathcal{S}$ is simple;
2. choose an order $\preceq_{\mathcal{S}}$ on $\mathcal{S}$ s.t. $\mathrm{enc}(G) \preceq_{\mathcal{S}} \mathrm{enc}(G') \Rightarrow G \preceq G'$;
3. show that $(\mathcal{S}, \preceq_{\mathcal{S}})$ is a WQO.

In other words: order-preserving functions send WQOs on WQOs.

Choice of $\mathcal{S}$:

- given by a decomposition;
- $\mathcal{S} =$ (labeled) 2-c graphs of $\mathcal{G}$;
- $\mathcal{S} \subseteq \mathcal{G}^{\star}$ (Nash-Williams' minimum bad sequence).

First example

Theorem (Błasiok, Kamiński, R., Trunck '15)

H-induced minor-free graphs are WQO by $\leqslant_{\mathrm{im}}$ iff H is induced minor of  *or* *.*

First example

Theorem (Błasiok, Kamiński, R., Trunck '15)

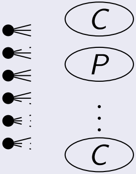
H-induced minor-free graphs are WQO by $\leqslant_{\mathrm{im}}$ iff H is induced minor of  *or* *.*

Lemma

*-induced minor-free graphs are WQO by induced minors.*

Lemma (Błasiok, Kamiński, R., Trunck '15)

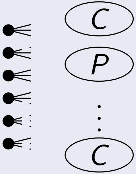
Every 2-connected  *-induced minor-free graph is like this:*



C: *cograph*
P: *path*

Lemma (Błasiok, Kamiński, R., Trunck '15)

Every 2-connected  *-induced minor-free graph is like this:*

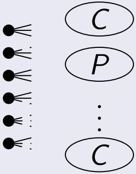


C: *cograph*
P: *path*

$$\mathrm{enc}(G) = (G_1, \lambda_1), \ldots, (G_k, \lambda_k)$$

Lemma (Błasiok, Kamiński, R., Trunck '15)

Every 2-connected  *-induced minor-free graph is like this:*



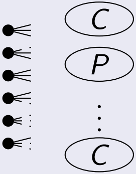
C: cograph
P: path

$\mathrm{enc}(G) = (G_1, \lambda_1), \ldots, (G_k, \lambda_k)$

- G_i is a path or a cograph;
- $\lambda_i \colon V(G_i) \to \{0,1\}^6$ (labeling)

Lemma (Błasiok, Kamiński, R., Trunck '15)

Every 2-connected  *-induced minor-free graph is like this:*



C: cograph
P: path

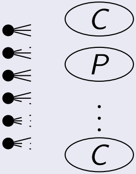
$\mathrm{enc}(G) = (G_1, \lambda_1), \ldots, (G_k, \lambda_k)$

- G_i is a path or a cograph;
- $\lambda_i \colon V(G_i) \to \{0,1\}^6$ (labeling)
- we choose an order on encodings.

Following the recipe

Lemma (Błasiok, Kamiński, R., Trunck '15)

Every 2-connected  *-induced minor-free graph is like this:*



C: cograph
P: path

$\mathrm{enc}(G) = (G_1, \lambda_1), \dots, (G_k, \lambda_k)$

- G_i is a path or a cograph;
- $\lambda_i \colon V(G_i) \to \{0,1\}^6$ (labeling)
- we choose an order on encodings.

(labeled) cographs and paths are easy to order.

Second example

Theorem (Błasiok, Kamiński, R., Trunck '15+)

H-contraction-free graphs are WQO by contractions iff H is a contraction of .

Second example

Theorem (Błasiok, Kamiński, R., Trunck '15+)

H-contraction-free graphs are WQO by contractions iff H is a contraction of .

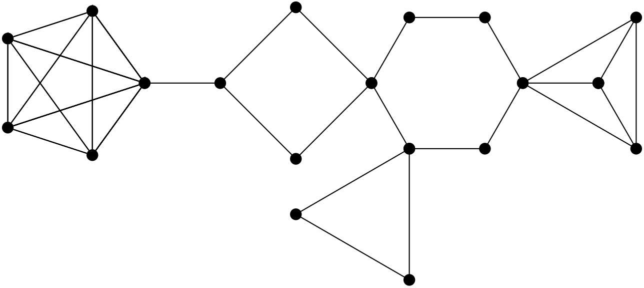
Lemma

*-contraction-free graphs are wqo by contractions.*

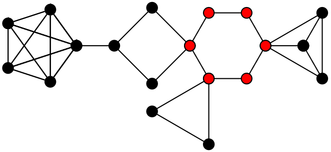
The decomposition

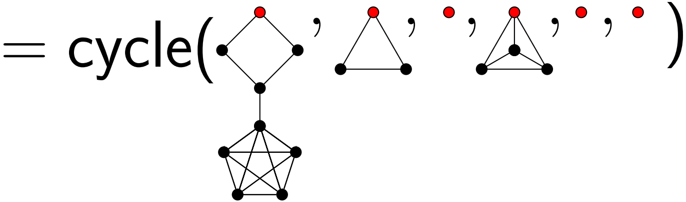
Lemma

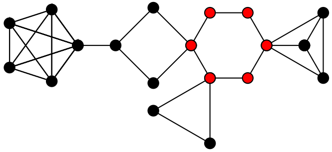
If G is  *-contraction-free then every block of G is either a clique or an induced cycle.*

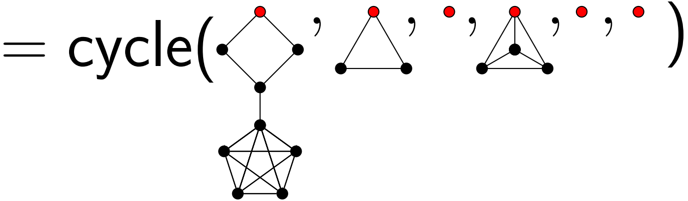


The encoding









If $(G_1, \ldots, G_p) \leqslant_{\mathrm{ctr}}{}^{\star} (H_1, \ldots, H_q)$

then $\mathrm{cycle}(G_1, \ldots, G_p) \leqslant_{\mathrm{ctr}} \mathrm{cycle}(H_1, \ldots, H_q)$

and $\mathrm{clique}(G_1, \ldots, G_p) \leqslant_{\mathrm{ctr}} \mathrm{clique}(H_1, \ldots, H_q)$

Well-quasi-ordering encodings

1. Consider a minimal infinite antichain:

$$A_1, A_2, \ldots$$

Well-quasi-ordering encodings

1. Consider a minimal infinite antichain:

$$A_1, A_2, \dots$$

2. $\mathcal{B} =$ "parts" of A_i's;

Well-quasi-ordering encodings

1. Consider a minimal infinite antichain:

$$A_1, A_2, \ldots$$

2. $\mathcal{B} =$ "parts" of A_i's;
3. $(\mathcal{B}, \leqslant_{\mathrm{ctr}})$ is a wqo, by minimality of $\{A_i\}_i$;

Well-quasi-ordering encodings

1. Consider a minimal infinite antichain:

$$A_1, A_2, \ldots$$

2. $\mathcal{B} =$ "parts" of A_i's;
3. $(\mathcal{B}, \leqslant_{\mathrm{ctr}})$ is a wqo, by minimality of $\{A_i\}_i$;
4. $(\mathcal{B}^\star, \leqslant_{\mathrm{ctr}})$ is a wqo;

Well-quasi-ordering encodings

1. Consider a minimal infinite antichain:

$$A_1, A_2, \ldots$$

2. $\mathcal{B} =$ "parts" of A_i's;
3. $(\mathcal{B}, \leqslant_{\mathrm{ctr}})$ is a wqo, by minimality of $\{A_i\}_i$;
4. $(\mathcal{B}^\star, \leqslant_{\mathrm{ctr}})$ is a wqo;
5. $(\mathrm{cycle}(\mathcal{B}^\star) \cup \mathrm{clique}(\mathcal{B}^\star), \leqslant_{\mathrm{ctr}})$ is a wqo;

Well-quasi-ordering encodings

1. Consider a minimal infinite antichain:

$$A_1, A_2, \ldots$$

2. $\mathcal{B} =$ "parts" of A_i's;
3. $(\mathcal{B}, \leqslant_{\mathrm{ctr}})$ is a wqo, by minimality of $\{A_i\}_i$;
4. $(\mathcal{B}^\star, \leqslant_{\mathrm{ctr}})$ is a wqo;
5. $(\mathrm{cycle}(\mathcal{B}^\star) \cup \mathrm{clique}(\mathcal{B}^\star), \leqslant_{\mathrm{ctr}})$ is a wqo;
6. $\{A_i\}_i$ is WQO, contradiction.

Well-quasi-ordering encodings

1. Consider a minimal infinite antichain:

$$A_1, A_2, \ldots$$

2. $\mathcal{B} =$ "parts" of A_i's;
3. $(\mathcal{B}, \leqslant_{\mathrm{ctr}})$ is a wqo, by minimality of $\{A_i\}_i$;
4. $(\mathcal{B}^\star, \leqslant_{\mathrm{ctr}})$ is a wqo;
5. $(\mathrm{cycle}(\mathcal{B}^\star) \cup \mathrm{clique}(\mathcal{B}^\star), \leqslant_{\mathrm{ctr}})$ is a wqo;
6. $\{A_i\}_i$ is WQO, contradiction.

-contraction-free graphs are WQO by contractions.

Conclusion

- general recipe to go from a decomposition to a wqo;

Conclusion

- general recipe to go from a decomposition to a wqo;
- needs a good choice of the (encoding, order)
  (not always possible);

Conclusion

- general recipe to go from a decomposition to a wqo;
- needs a good choice of the (encoding, order) (not always possible);
- well-quasi-orders are "simple" orders.

Conclusion

- general recipe to go from a decomposition to a wqo;
- needs a good choice of the (encoding, order) (not always possible);
- well-quasi-orders are "simple" orders.

Further work:

- study the limit cases for parameterized classes in non-wqos (H-$\leqslant$-free, sparse classes, bounded parameter, etc.);
- which classes are wqo by strong immersions?

Conclusion

- general recipe to go from a decomposition to a wqo;
- needs a good choice of the (encoding, order)
  (not always possible);
- well-quasi-orders are “simple” orders.

Further work:

- study the limit cases for parameterized classes in non-wqos
  (H-$\leqslant$-free, sparse classes, bounded parameter, etc.);
- which classes are wqo by strong immersions?

Thank you!