

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. \leq
 $\rightarrow (\mathbb{Q}, \leq)$ 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. \leq
 $\rightarrow (\mathbb{Q}, \leq)$ is not a WQO;
- $\{0\}, \{1\}, \{2\}, \dots$ 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. \leq
 $\rightarrow (\mathbb{Q}, \leq)$ is not a WQO;
- $\{0\}, \{1\}, \{2\}, \dots$ is an infinite antichain wrt. \subseteq
 $\rightarrow (\mathcal{P}(\mathbb{N}), \subseteq)$ is not a WQO;
- $(A^*, \leq_{\text{subseq}})$ with A finite: WQO;
- (graphs, \leq_{minor}): WQO.

Why do we like well-quasi-orders?

Upwards closed classes have a **finite** number of minimal elements.



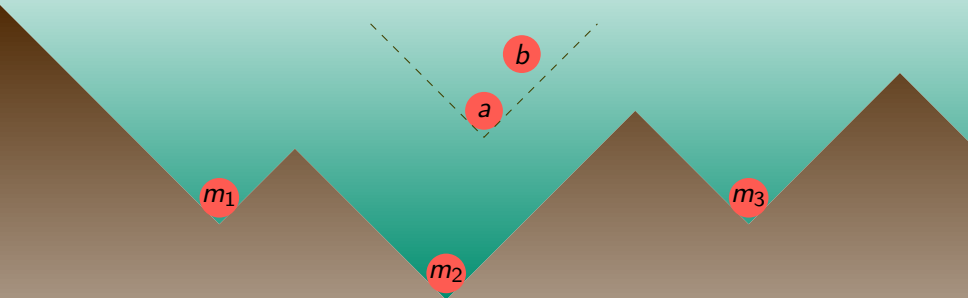
m_1

m_2

m_3

Why do we like well-quasi-orders?

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.

$$x \in U \iff m_1 \leq x \vee \dots \vee m_k \leq x$$

(finite base)

A diagram illustrating a well-quasi-order. The background is a light teal color. A brown, jagged shape represents the complement of an upwards closed class. The shape has three valleys, each containing a red circle with a white label: m_1 in the left valley, m_2 in the center valley, and m_3 in the right valley. The top of the brown shape is a series of connected line segments forming a sawtooth pattern.

m_1

m_2

m_3

Why do we like well-quasi-orders?

Upwards closed classes have a **finite** number of minimal elements.

$$x \in U \iff m_1 \leq x \vee \dots \vee m_k \leq x$$

(finite base)

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

A diagram illustrating a well-quasi-order. The background is a light teal color with a brown, jagged, mountain-like shape at the bottom. Three red circles, each containing a label, are positioned at the valleys of the brown shape. The labels are m_1 , m_2 , and m_3 , representing minimal elements. m_1 is on the left, m_2 is in the center, and m_3 is on the right. The teal area above the brown shape represents the set of elements greater than or equal to these minimal elements.

m_1

m_2

m_3

Why do we like well-quasi-orders?

Upwards closed classes have a **finite** number of minimal elements.

$$x \in U \iff m_1 \leq x \vee \dots \vee m_k \leq x$$

(finite base)

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



The diagram shows a brown background with a teal-colored region. The teal region is bounded by a jagged, downward-pointing path. Three red circles, each containing a black label, are placed at the bottom-most points of the teal region. From left to right, these labels are m1, m2, and m3. m1 is at the leftmost valley, m2 is at the central valley, and m3 is at the rightmost valley. The teal region is above these points, representing the set of elements greater than or equal to at least one of the minimal elements.

m_1

m_2

m_3

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.

$$x \in U \iff m_1 \leq x \vee \dots \vee m_k \leq x$$

(finite base)

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



m_1

m_2

m_3

Downwards closed classes have a **finite** number of obstructions.



a

b

Why do we like well-quasi-orders?

Upwards closed classes have a **finite** number of minimal elements.

$$x \in U \iff m_1 \leq x \vee \dots \vee m_k \leq x$$

(finite base)

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



m_1

m_2

m_3

Downwards closed classes have a **finite** number of obstructions.

$$x \in D \iff m_1 \not\leq x \wedge \dots \wedge m_k \not\leq x$$

Why do we like well-quasi-orders?

Upwards closed classes have a **finite** number of minimal elements.

$$x \in U \iff m_1 \leq x \vee \dots \vee m_k \leq x$$

(finite base)

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



m_1

m_2

m_3

Downwards closed classes have a **finite** number of obstructions.

$$x \in D \iff m_1 \not\leq x \wedge \dots \wedge m_k \not\leq x$$

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

Why do we like well-quasi-orders?

graphs of genus $\geq g$ + minor relation

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

...

m_1

m_2

m_3

graphs of treewidth $\leq k$ + minor relation

trees + contraction relation

...

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

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

- contraction relation \leq_{ctr} : E contraction;
- induced minor relation \leq_{im} : V deletion and E contraction;
- minor relation \leq_{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

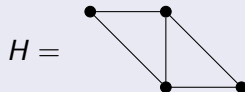
Containment relations on graphs

- 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.

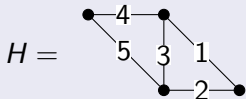
Toy example

Subdivisions of H are WQO by contraction.




Toy example

Subdivisions of H are WQO by contraction.

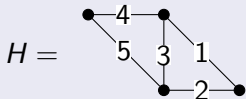


- 1 choose an encoding of graphs as tuples (for instance)
e.g. # of subdivisions for each edge, in some chosen order;

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

Toy example

Subdivisions of H are WQO by contraction.



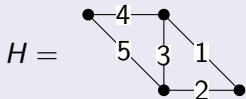
- 1 choose an encoding of graphs as tuples (for instance)
e.g. # of subdivisions for each edge, in some chosen order;

$$\text{enc}\left(\begin{array}{c} \bullet \quad \bullet \\ \circ \quad \circ \\ \circ \quad \bullet \end{array}\right) = (1, 0, 1, 0, 2)$$

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

Toy example

Subdivisions of H are WQO by contraction.



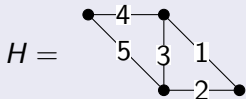
- 1 choose an encoding of graphs as tuples (for instance)
e.g. # of subdivisions for each edge, in some chosen order;

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

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

Toy example

Subdivisions of H are WQO by contraction.



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

$$\text{enc}\left(\begin{array}{c} \bullet \text{---} \bullet \\ \circ \text{---} \circ \\ \bullet \text{---} \bullet \end{array} \right) = (1, 0, 1, 0, 2)$$

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

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

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

- 1 choose a mapping $enc: \mathcal{G} \rightarrow \mathcal{S}$ where \mathcal{S} is **simple**;

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

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

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

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

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

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

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

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

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

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

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

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



- 1 choose a mapping $enc: \mathcal{G} \rightarrow \mathcal{S}$ where \mathcal{S} is **simple**;
- 2 choose an order $\preceq_{\mathcal{S}}$ on \mathcal{S} s.t. $enc(G) \preceq_{\mathcal{S}} enc(G') \Rightarrow G \preceq G'$;
- 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} ;
- $\mathcal{S} \subseteq \mathcal{G}^*$ (Nash-Williams' minimum bad sequence).


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

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


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

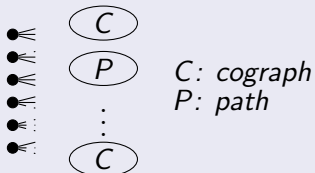
H-induced minor-free graphs are WQO by \leq_{im} iff *H* is induced minor of  or .

Lemma


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

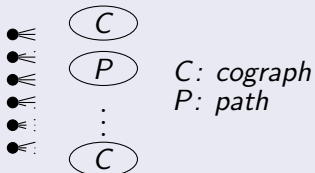
Lemma (Błasiok, Kamiński, R., Trunck '15)

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



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

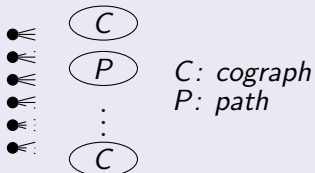
Every 2-connected -induced minor-free graph is like this:



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

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

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

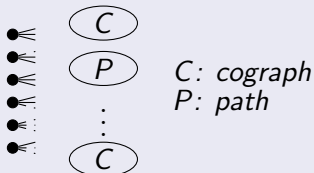


$\text{enc}(G) = (G_1, \lambda_1), \dots, (G_k, \lambda_k)$

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

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

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

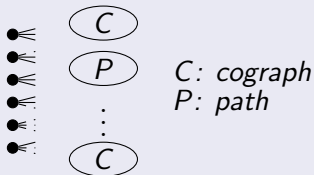


$\text{enc}(G) = (G_1, \lambda_1), \dots, (G_k, \lambda_k)$

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

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

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




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

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

(labeled) cographs and paths are easy to order.

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 \diamond .

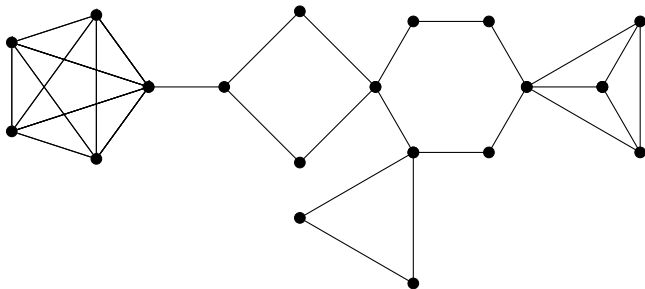
Lemma

\diamond -contraction-free graphs are wqo by contractions.

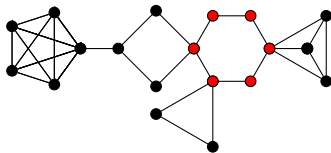
The decomposition

Lemma

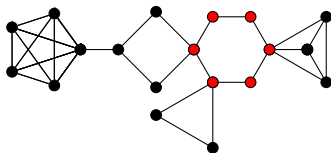
If G is \diamond -contraction-free then every block of G is either a *clique* or an *induced cycle*.



The encoding



$$= \text{cycle}(\text{graph}_1, \text{graph}_2, \text{graph}_3, \text{graph}_4, \text{graph}_5, \text{graph}_6)$$



$$= \text{cycle}(\text{diamond}, \text{triangle}, \text{square}, \text{triangle}, \dots)$$

If $(G_1, \dots, G_p) \leq_{\text{ctr}^*} (H_1, \dots, H_q)$

then $\text{cycle}(G_1, \dots, G_p) \leq_{\text{ctr}} \text{cycle}(H_1, \dots, H_q)$

and $\text{clique}(G_1, \dots, G_p) \leq_{\text{ctr}} \text{clique}(H_1, \dots, H_q)$

- 1 Consider a **minimal** infinite antichain:

$$A_1, A_2, \dots$$

- 1 Consider a **minimal** infinite antichain:

$$A_1, A_2, \dots$$

- 2 $\mathcal{B} =$ “parts” of A_i 's;

- 1 Consider a **minimal** infinite antichain:

$$A_1, A_2, \dots$$

- 2 \mathcal{B} = “parts” of A_i 's;
- 3 $(\mathcal{B}, \leq_{\text{ctr}})$ is a wqo, by minimality of $\{A_i\}_i$;

- 1 Consider a **minimal** infinite antichain:

$$A_1, A_2, \dots$$

- 2 \mathcal{B} = “parts” of A_i 's;
- 3 $(\mathcal{B}, \leq_{\text{ctr}})$ is a wqo, by minimality of $\{A_i\}_i$;
- 4 $(\mathcal{B}^*, \leq_{\text{ctr}})$ is a wqo;

- 1 Consider a **minimal** infinite antichain:

$$A_1, A_2, \dots$$

- 2 \mathcal{B} = “parts” of A_i 's;
- 3 $(\mathcal{B}, \leq_{\text{ctr}})$ is a wqo, by minimality of $\{A_i\}_i$;
- 4 $(\mathcal{B}^*, \leq_{\text{ctr}})$ is a wqo;
- 5 $(\text{cycle}(\mathcal{B}^*) \cup \text{clique}(\mathcal{B}^*), \leq_{\text{ctr}})$ is a wqo;

- 1 Consider a **minimal** infinite antichain:


$$A_1, A_2, \dots$$

- 2 \mathcal{B} = “parts” of A_i 's;
- 3 $(\mathcal{B}, \leq_{\text{ctr}})$ is a wqo, by minimality of $\{A_i\}_i$;
- 4 $(\mathcal{B}^*, \leq_{\text{ctr}})$ is a wqo;
- 5 $(\text{cycle}(\mathcal{B}^*) \cup \text{clique}(\mathcal{B}^*), \leq_{\text{ctr}})$ is a wqo;
- 6 $\{A_i\}_i$ is WQO, contradiction.

- 1 Consider a **minimal** infinite antichain:

$$A_1, A_2, \dots$$

- 2 \mathcal{B} = “parts” of A_i 's;
- 3 $(\mathcal{B}, \leq_{\text{ctr}})$ is a wqo, by minimality of $\{A_i\}_i$;
- 4 $(\mathcal{B}^*, \leq_{\text{ctr}})$ is a wqo;
- 5 $(\text{cycle}(\mathcal{B}^*) \cup \text{clique}(\mathcal{B}^*), \leq_{\text{ctr}})$ is a wqo;
- 6 $\{A_i\}_i$ is WQO, contradiction.

-contraction-free graphs are WQO by contractions.

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

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

- 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.

- 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 - \leq -free, sparse classes, bounded parameter, etc.);
- which classes are wqo by strong immersions?

- 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 - \leq -free, sparse classes, bounded parameter, etc.);
- which classes are wqo by strong immersions?

Thank you!