Local certification of/on sparse graph classes

Nicolas Bousquet

joint works with
Laurent Feuilloley   Théo Pierron

CIRM - December 2021

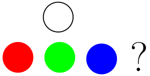
Disclaimer

All graphs are connected !

3-coloring and LOCAL model

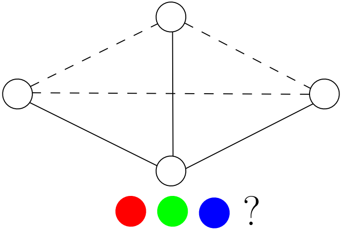
All the vertices have a unique ID.

3-coloring and LOCAL model



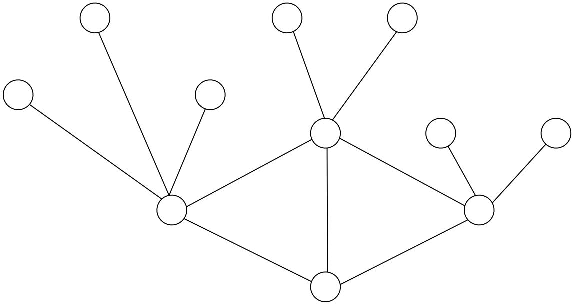
All the vertices have a unique ID.

3-coloring and LOCAL model



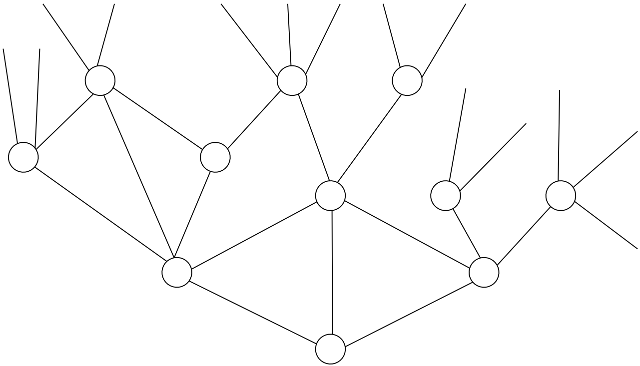
All the vertices have a unique ID.

3-coloring and LOCAL model



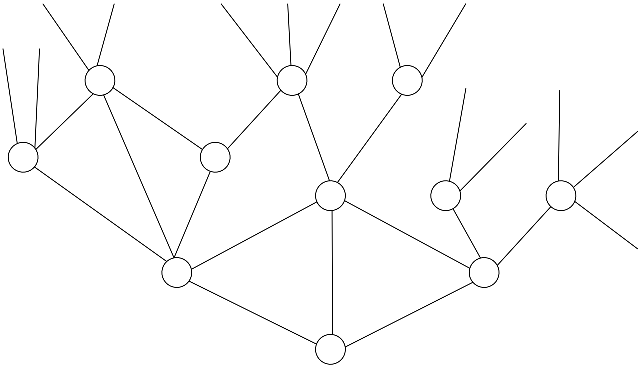
All the vertices have a unique ID.

3-coloring and LOCAL model



All the vertices have a unique ID.

3-coloring and LOCAL model

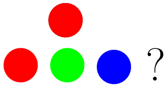


All the vertices have a unique ID.

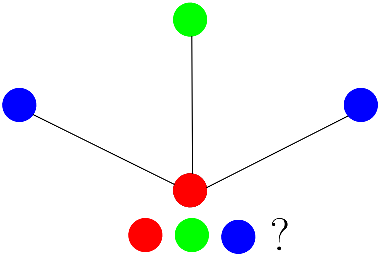
Question

Up to which distance do we have to look at to take a correct decision ?

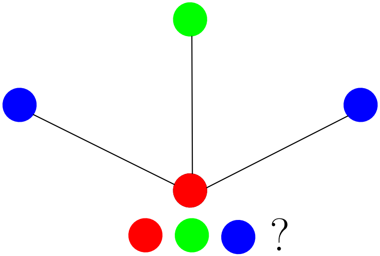
Certify the 3-coloring



Certify the 3-coloring



Certify the 3-coloring



If vertices receive as labels their colors, we can check the coloring in the future by looking at vertices at distance 1 !

Local certification

- Every node has a unique ID in $[1, n]$. (It can be $O(n^c)$ (almost) for free)
- Every node has a certificate.

Local certification

- Every node has a unique ID in $[1, n]$. (It can be $O(n^c)$ (almost) for free)
- Every node has a certificate.

Certification algorithm

- Every node reads its certificate and the certificate of its neighbors.
- Every node has an inifinite power of computation.

Local certification

- Every node has a unique ID in $[1, n]$. (It can be $O(n^c)$ (almost) for free)
- Every node has a certificate.

Certification algorithm

- Every node reads its certificate and the certificate of its neighbors.
- Every node has an inifinite power of computation.
- Every node finally accepts or reject.

Local certification

- Every node has a unique ID in $[1, n]$. (It can be $O(n^c)$ (almost) for free)
- Every node has a certificate.

Certification algorithm

- Every node reads its certificate and the certificate of its neighbors.
- Every node has an inifinite power of computation.
- Every node finally accepts or reject.

A property Π can be certified with $f(n)$ bits when :

- If Π is positive, there exists a certificate assignment to the nodes, each of size at most $f(n)$, such that all the nodes accept.
- If Π is negative, at least one node rejects for any possible certificate assignment.

What can be certified locally ?

What can be certified locally ?

Everything !

What can be certified locally ?

Everything !

(with certificate of size $O(n^2)$)

What can be certified locally ?

Everything !

(with certificate of size $O(n^2)$)

Idea :
Give the entire graph with IDs to each vertex.

What can be certified locally ?

Everything !

(with certificate of size $O(n^2)$)

Idea :
Give the entire graph with IDs to each vertex.

Questions :

- Can we improve $\Omega(n^2)$ in general ?

What can be certified locally ?

Everything !

(with certificate of size $O(n^2)$)

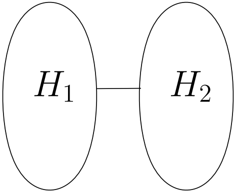
Idea :
Give the entire graph with IDs to each vertex.

Questions :

- Can we improve $\Omega(n^2)$ in general ?
- What is a decent lower bound ?
  $\rightarrow \Omega(\log n)$ (size of labels).

What can't be certified with small certificate ?

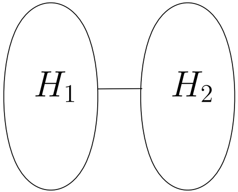
[Göös, Suomela '16] **Non trivial automorphism**



$\rightarrow \Omega(n^2)$ bits are needed.

What can't be certified with small certificate ?

[Göös, Suomela '16] **Non trivial automorphism**



→ $\Omega(n^2)$ bits are needed.

[Censor-Hillel et al. '20] **Diameter 2.**
→ $\Omega(n)$ bits.

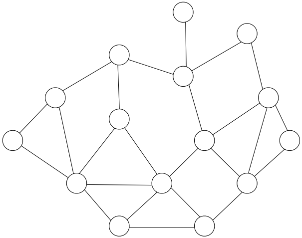
Certifying spanning trees

[Korman et al. '10] Spanning trees can be certified with $O(\log n)$ rounds.

Certifying spanning trees

[Korman et al. '10] Spanning trees can be certified with $O(\log n)$ rounds.

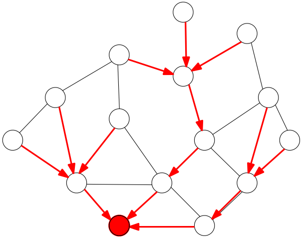
1. Every node receives its parent ID (and the root its own ID).



Certifying spanning trees

[Korman et al. '10] Spanning trees can be certified with $O(\log n)$ rounds.

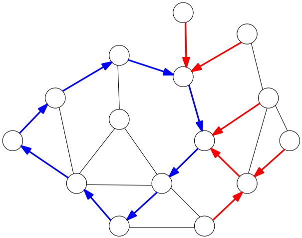
1. Every node receives its parent ID (and the root its own ID).



Certifying spanning trees

[Korman et al. '10] Spanning trees can be certified with $O(\log n)$ rounds.

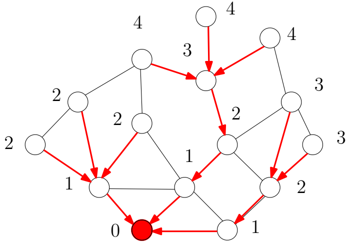
1. Every node receives its parent ID (and the root its own ID).



Certifying spanning trees

[Korman et al. '10] Spanning trees can be certified with $O(\log n)$ rounds.

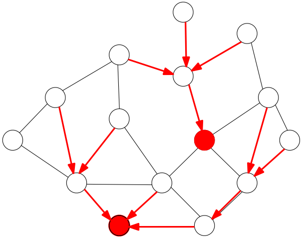
1. Every node receives its parent ID (and the root its own ID).
2. Every node receives its distance (in the tree) to the root.



Certifying spanning trees

[Korman et al. '10] Spanning trees can be certified with $O(\log n)$ rounds.

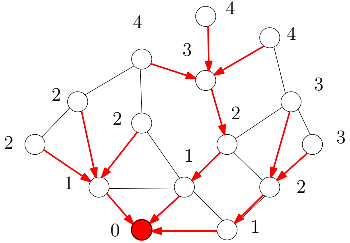
1. Every node receives its parent ID (and the root its own ID).
2. Every node receives its distance (in the tree) to the root.



Certifying spanning trees

[Korman et al. '10] Spanning trees can be certified with $O(\log n)$ rounds.

1. Every node receives its parent ID (and the root its own ID).
2. Every node receives its distance (in the tree) to the root.
3. Every node receives the ID of the root.



Research directions (I)

Which properties can be certified locally with $O(\log n)$ bits ?

- k-colorable graphs.

Research directions (I)

Which properties can be certified locally with $O(\log n)$ bits ?

- k-colorable graphs.
- [Feuilloley et al. '20] Planar graphs.
- [Feuilloley et al., Esperet et al. '21] Bounded genus.

Research directions (I)

Which properties can be certified locally with $O(\log n)$ bits ?

- k-colorable graphs.
- [Feuilloley et al. '20] Planar graphs.
- [Feuilloley et al., Esperet et al. '21] Bounded genus.
- [Montealegre et al. '21+] Interval and chordal graphs.

Research directions (I)

Which properties can be certified locally with $O(\log n)$ bits ?

- k-colorable graphs.
- [Feuilloley et al. '20] Planar graphs.
- [Feuilloley et al., Esperet et al. '21] Bounded genus.
- [Montealegre et al. '21+] Interval and chordal graphs.

Question : (Feuilloley)
Is it possible to certify every H-minor free graph class with $O(\log n)$ bits ?

Research directions (I)

Which properties can be certified locally with $O(\log n)$ bits ?

- k-colorable graphs.
- [Feuilloley et al. '20] Planar graphs.
- [Feuilloley et al., Esperet et al. '21] Bounded genus.
- [Montealegre et al. '21+] Interval and chordal graphs.

Question : (Feuilloley)
Is it possible to certify every H-minor free graph class with $O(\log n)$ bits ?

- [B., Feuilloley, Pierron '21] True if $|H| \leq 4$.

Research directions (I)

Which properties can be certified locally with $O(\log n)$ bits ?

- k-colorable graphs.
- [Feuilloley et al. '20] Planar graphs.
- [Feuilloley et al., Esperet et al. '21] Bounded genus.
- [Montealegre et al. '21+] Interval and chordal graphs.

Question : (Feuilloley)
Is it possible to certify every H-minor free graph class with $O(\log n)$ bits ?

- [B., Feuilloley, Pierron '21] True if $|H| \leq 4$.
- [Fraigniaud et al. '21] True for planar H with $O(\log^2 n)$ certificate.

Research directions (I)

Which properties can be certified locally with $O(\log n)$ bits ?

- k-colorable graphs.
- [Feuilloley et al. '20] Planar graphs.
- [Feuilloley et al., Esperet et al. '21] Bounded genus.
- [Montealegre et al. '21+] Interval and chordal graphs.

Question : (Feuilloley)
Is it possible to certify every H-minor free graph class with $O(\log n)$ bits ?

- [B., Feuilloley, Pierron '21] True if $|H| \leq 4$.
- [Fraigniaud et al. '21] True for planar H with $O(\log^2 n)$ certificate.

On the way :
Certification of 2 and 3-connectivity, block cut trees, development of new tools...

Research directions (II)

On which (certifiable) graph classes can we certify "many things" locally ?
$\approx$ Obtain an analogue of Courcelle's theorem for certification.

Research directions (II)

On which (certifiable) graph classes can we certify "many things" locally ?
$\approx$ Obtain an analogue of Courcelle's theorem for certification.

Theorem (B, Feuilloley, Pierron '21+)

- $\mathrm{td}(G) \leq k$ can be certified with $O(k \log n)$ bits.

Research directions (II)

On which (certifiable) graph classes can we certify "many things" locally ?
$\approx$ Obtain an analogue of Courcelle's theorem for certification.

Theorem (B, Feuilloley, Pierron '21+)

- $\mathrm{td}(G) \leq k$ can be certified with $O(k \log n)$ bits.
- Every MSO formula can be certified with $O(\log n)$ bits on bounded treedepth graphs.

Certifying K_4-minor free graphs (I)

Lemma (B., Feuilloley, Pierron)

If H is 2-connected :
Certification of 2-connected H-minor-free graphs with $O(\log(n))$ bits
$\Rightarrow$ Certification of H-minor-free graphs with $O(\log n)$ bits.

Certifying K_4-minor free graphs (I)

Lemma (B., Feuilloley, Pierron)

If H is 2-connected :
Certification of 2-connected H-minor-free graphs with $O(\log(n))$ bits
$\Rightarrow$ Certification of H-minor-free graphs with $O(\log n)$ bits.

(Idea of the) Proof :

- A model of H should belong to a 2-connected component.

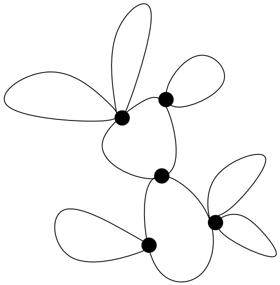
Certifying K_4-minor free graphs (I)

Lemma (B., Feuilloley, Pierron)

If H is 2-connected :
Certification of 2-connected H-minor-free graphs with $O(\log(n))$ bits
$\Rightarrow$ Certification of H-minor-free graphs with $O(\log n)$ bits.

(Idea of the) Proof :

- A model of H should belong to a 2-connected component.
- Certify the block cut tree.



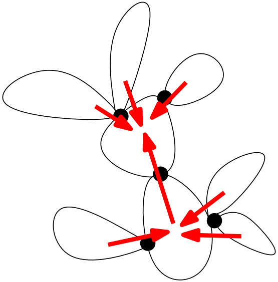
Certifying K_4-minor free graphs (I)

Lemma (B., Feuilloley, Pierron)

If H is 2-connected :
Certification of 2-connected H-minor-free graphs with $O(\log(n))$ bits
$\Rightarrow$ Certification of H-minor-free graphs with $O(\log n)$ bits.

(Idea of the) Proof :

- A model of H should belong to a 2-connected component.
- Certify the block cut tree.



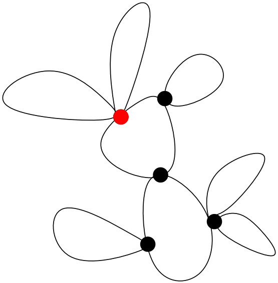
Certifying K_4-minor free graphs (I)

Lemma (B., Feuilloley, Pierron)

If H is 2-connected :
Certification of 2-connected H-minor-free graphs with $O(\log(n))$ bits
$\Rightarrow$ Certification of H-minor-free graphs with $O(\log n)$ bits.

(Idea of the) Proof :

- A model of H should belong to a 2-connected component.
- Certify the block cut tree.



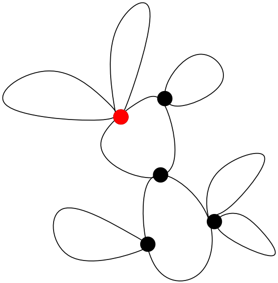
Certifying K_4-minor free graphs (I)

Lemma (B., Feuilloley, Pierron)

If H is 2-connected :
Certification of 2-connected H-minor-free graphs with $O(\log(n))$ bits
$\Rightarrow$ Certification of H-minor-free graphs with $O(\log n)$ bits.

(Idea of the) Proof :

- A model of H should belong to a 2-connected component.
- Certify the block cut tree.
- Certify that 2-connected components are 2-connected.



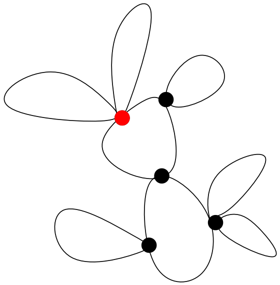
Certifying K_4-minor free graphs (I)

Lemma (B., Feuilloley, Pierron)

If H is 2-connected :
Certification of 2-connected H-minor-free graphs with $O(\log(n))$ bits
$\Rightarrow$ Certification of H-minor-free graphs with $O(\log n)$ bits.

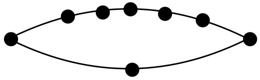
(Idea of the) Proof :

- A model of H should belong to a 2-connected component.
- Certify the block cut tree.
- Certify that 2-connected components are 2-connected.
- Give the H-certificate to each 2-connected component.



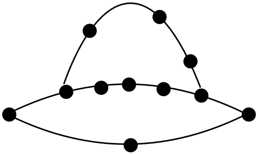
Ear decompositions $\Leftrightarrow$ 2-connected.

- Start from a cycle



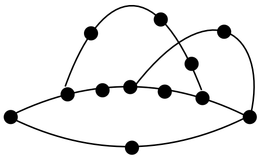
Ear decompositions ⇔ 2-connected.

- Start from a cycle



Ear decompositions ⇔ 2-connected.

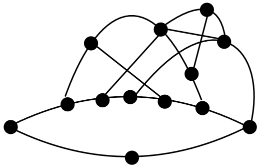
- Start from a cycle
- Add new paths called ears



K_4-free graphs (II)

Ear decompositions $\Leftrightarrow$ 2-**connected.**

- Start from a cycle
- Add new paths called ears



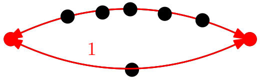
K_4-free graphs (II)

Ear decompositions $\Leftrightarrow$ 2-connected.

- Start from a cycle
- Add new paths called ears

Certify ear decompositions.

- Certify the initial cycle and give label 1 to these vertices.



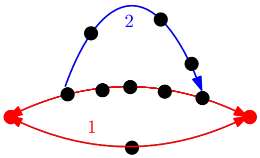
K_4-free graphs (II)

Ear decompositions $\Leftrightarrow$ 2-**connected.**

- Start from a cycle
- Add new paths called ears

Certify ear decompositions.

- Certify the initial cycle and give label 1 to these vertices.
- Certify the new path and give label i to the vertices of the i-th ear.

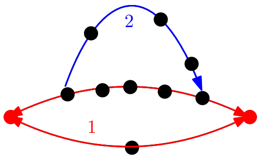


Ear decompositions $\Leftrightarrow$ 2-connected.

- Start from a cycle
- Add new paths called ears

Certify ear decompositions.

- Certify the initial cycle and give label 1 to these vertices.
- Certify the new path and give label i to the vertices of the i-th ear.



Theorem (Eppstein)

The following are equivalent :

- G is a 2-connected K_4-minor-free graph,
- G is a 2-connected series-parallel graphs,
- G has a nested ear decomposition.

Question 1 - Next step ?

Question

Certification of H-minor free graphs with $O(\log n)$ bits
$\Rightarrow$ Certification of $(H + K_1)$-minor free graphs with $O(\log n)$ bits ?

Question 1 - Next step ?

Question

Certification of H-minor free graphs with $O(\log n)$ bits
$\Rightarrow$ Certification of $(H + K_1)$-minor free graphs with $O(\log n)$ bits ?

Example :
Can we certify (vertex) minimally non planar graphs ?

Question 1 - Next step ?

Question

Certification of H-minor free graphs with $O(\log n)$ bits
$\Rightarrow$ Certification of $(H + K_1)$-minor free graphs with $O(\log n)$ bits ?

Example :

Can we certify (vertex) minimally non planar graphs ?

Question

Certity F-minor free graphs for any forest F ?

Question 1 - Next step ?

Question

Certification of H-minor free graphs with $O(\log n)$ bits
$\Rightarrow$ Certification of $(H + K_1)$-minor free graphs with $O(\log n)$ bits ?

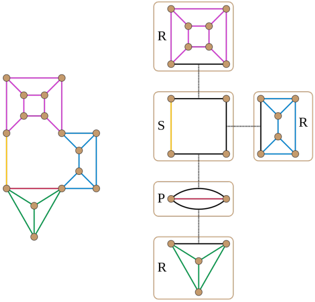
Example :
Can we certify (vertex) minimally non planar graphs ?

Question

Certity F-minor free graphs for any forest F ?

[Fraigniaud et al. '21] Certificate of size $O(\log^2 n)$.

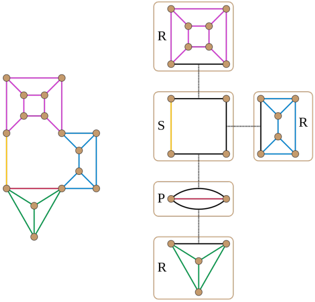
Question 2 - SPQR trees partition



Source : wikipedia

- S node - Cycle of length at least 3.
- P node - Multigraph with 2 vertices and ≥ 3 edges.
- Q node - Single real edge.
- R node - 3-connected graph that is not S or P.

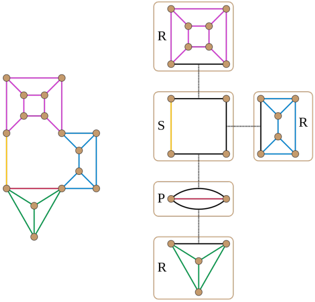
Question 2 - SPQR trees partition



Source : wikipedia

- S node - Cycle of length at least 3.
- P node - Multigraph with 2 vertices and ≥ 3 edges.
- Q node - Single real edge.
- R node - 3-connected graph that is not S or P.

Question 2 - SPQR trees partition

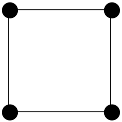


Source : wikipedia

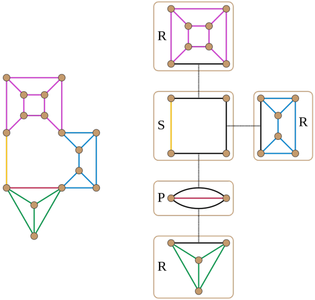
- S node - Cycle of length at least 3.
- P node - Multigraph with 2 vertices and ≥ 3 edges.
- Q node - Single real edge.
- R node - 3-connected graph that is not S or P.

Question : Certification of SPQR-trees

- Certifying S,P,Q,R components ✓



Question 2 - SPQR trees partition

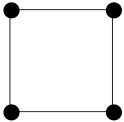


Source : wikipedia

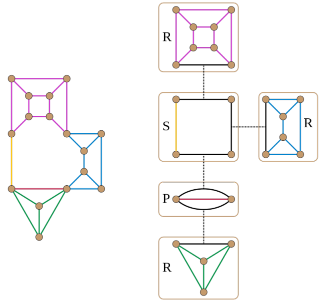
- S node - Cycle of length at least 3.
- P node - Multigraph with 2 vertices and ≥ 3 edges.
- Q node - Single real edge.
- R node - 3-connected graph that is not S or P.

Question : Certification of SPQR-trees

- Certifying S,P,Q,R components ✓
- Deal with "iterated" false edges ?



Question 2 - SPQR trees partition

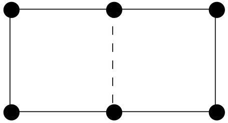


- S node - Cycle of length at least 3.
- P node - Multigraph with 2 vertices and ≥ 3 edges.
- Q node - Single real edge.
- R node - 3-connected graph that is not S or P.

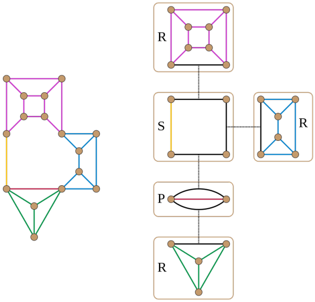
Source : wikipedia

Question : Certification of SPQR-trees

- Certifying S,P,Q,R components ✓
- Deal with "iterated" false edges ?



Question 2 - SPQR trees partition

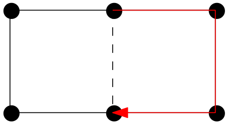


Source : wikipedia

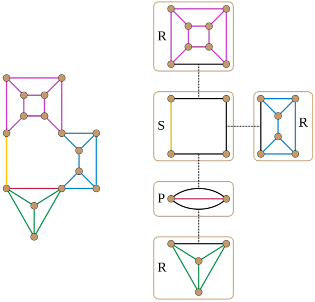
- S node - Cycle of length at least 3.
- P node - Multigraph with 2 vertices and ≥ 3 edges.
- Q node - Single real edge.
- R node - 3-connected graph that is not S or P.

Question : Certification of SPQR-trees

- Certifying S,P,Q,R components ✓
- Deal with "iterated" false edges ?



Question 2 - SPQR trees partition

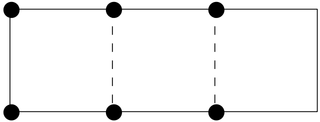


- S node - Cycle of length at least 3.
- P node - Multigraph with 2 vertices and ≥ 3 edges.
- Q node - Single real edge.
- R node - 3-connected graph that is not S or P.

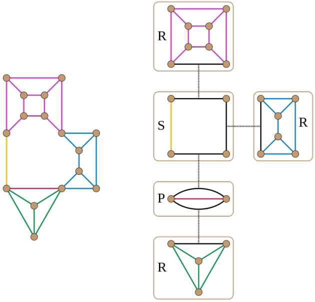
Source : wikipedia

Question : Certification of SPQR-trees

- Certifying S,P,Q,R components ✓
- Deal with "iterated" false edges ?



Question 2 - SPQR trees partition

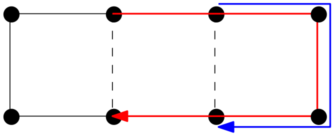


Source : wikipedia

- S node - Cycle of length at least 3.
- P node - Multigraph with 2 vertices and ≥ 3 edges.
- Q node - Single real edge.
- R node - 3-connected graph that is not S or P.

Question : Certification of SPQR-trees

- Certifying S,P,Q,R components ✓
- Deal with "iterated" false edges ?



Question 3 - K_5-minor free graphs

Theorem (Wagner ‘’37)

K_5-minor free graphs $\Leftrightarrow$ 3-sums of planar graphs and Wagner graph.

Question 3 - K_5-minor free graphs

Theorem (Wagner ''37)

K_5-minor free graphs $\Leftrightarrow$ 3-sums of planar graphs and Wagner graph.

- Certifying planar graphs and Wagner graphs ✓

Question 3 - K_5-minor free graphs

Theorem (Wagner ''37)

K_5-minor free graphs $\Leftrightarrow$ 3-sums of planar graphs and Wagner graph.

- Certifying planar graphs and Wagner graphs ✓
- Gluing on cliques in a tree like structure ✓

Question 3 - K_5-minor free graphs

Theorem (Wagner ''37)

K_5-minor free graphs $\Leftrightarrow$ 3-sums of planar graphs and Wagner graph.

- Certifying planar graphs and Wagner graphs ✓
- Gluing on cliques in a tree like structure ✓
- We can remove edges of the cliques afterwards ?

Question 3 - K_5-minor free graphs

Theorem (Wagner ''37)

K_5-minor free graphs $\Leftrightarrow$ 3-sums of planar graphs and Wagner graph.

- Certifying planar graphs and Wagner graphs ✓
- Gluing on cliques in a tree like structure ✓
- We can remove edges of the cliques afterwards ?

Remark : A similar result for $K_{3,3}$-graphs exist.

Monadic Second Order logic

First order (FO) :

- Quantifies on vertices
- Predicate $\rightarrow$ adjacency

$$\forall x \forall y, (x = y) \vee (x - y) \vee \exists z (x - z \wedge z - y)$$

Monadic Second Order logic

First order (FO) :

- Quantifies on vertices
- Predicate $\rightarrow$ adjacency

$$\forall x \forall y, (x = y) \vee (x - y) \vee \exists z (x - z \wedge z - y)$$

Monadic second order :

- Quantifies on sets of vertices (MSO_1) and edges (MSO_2)
- Predicates $\rightarrow$ adjacency + membership

$$\begin{aligned} &\exists V_1 \exists V_2, \\ &\quad \forall v, (v \in V_1) \Leftrightarrow \neg(v \in V_2) \\ &\wedge \forall v \forall w, v - w \Rightarrow (v \in V_1 \Leftrightarrow w \in V_2) \end{aligned}$$

Monadic Second Order logic

First order (FO) :

- Quantifies on vertices
- Predicate → adjacency

$$\forall x \forall y, (x = y) \vee (x - y) \vee \exists z (x - z \wedge z - y)$$

Monadic second order :

- Quantifies on sets of vertices (MSO_1) and edges (MSO_2)
- Predicates → adjacency + membership

$$\begin{aligned} &\exists V_1 \exists V_2, \\ &\quad \forall v, (v \in V_1) \Leftrightarrow \neg(v \in V_2) \\ &\wedge \forall v \forall w, v - w \Rightarrow (v \in V_1 \Leftrightarrow w \in V_2) \end{aligned}$$

- Excluding a minor is MSO_1
- Hamiltonian cycle is MSO_2

Certifying MSO formulas

Theorem (Courcelle, informal)

Every MSO formula can be decided in FPT time parameterized by the treewidth.

Certifying MSO formulas

Theorem (Courcelle, informal)

Every MSO formula can be decided in FPT time parameterized by the treewidth.

Ultimate goal :

Every MSO formula can be certified with $O(\log n)$ bits on bounded treewidth graphs.

Certifying MSO formulas

Theorem (Courcelle, informal)

Every MSO formula can be decided in FPT time parameterized by the treewidth.

Ultimate goal :

Every MSO formula can be certified with $O(\log n)$ bits on bounded treewidth graphs.

But...

- Certify bounded treewidth graphs is open...

Certifying MSO formulas

Theorem (Courcelle, informal)

Every MSO formula can be decided in FPT time parameterized by the treewidth.

Ultimate goal :

Every MSO formula can be certified with $O(\log n)$ bits on bounded treewidth graphs.

But...

- Certify bounded treewidth graphs is open...
  Issue 1 - Diameter of the bags is arbitrary.

Certifying MSO formulas

Theorem (Courcelle, informal)

Every MSO formula can be decided in FPT time parameterized by the treewidth.

Ultimate goal :

Every MSO formula can be certified with $O(\log n)$ bits on bounded treewidth graphs.

But...

- Certify bounded treewidth graphs is open...
  Issue 1 - Diameter of the bags is arbitrary.
  Issue 2 - A vertex can belong to a lot of bags of the tree decomposition.

Certifying MSO formulas

Theorem (Courcelle, informal)

Every MSO formula can be decided in FPT time parameterized by the treewidth.

Ultimate goal :

Every MSO formula can be certified with $O(\log n)$ bits on bounded treewidth graphs.

But...

- Certify bounded treewidth graphs is open...
  Issue 1 - Diameter of the bags is arbitrary.
  Issue 2 - A vertex can belong to a lot of bags of the tree decomposition.
- ... And even if we assume that we have bounded treewidth, not clear how to do it !

Certifying MSO formulas

Theorem (Courcelle, informal)

Every MSO formula can be decided in FPT time parameterized by the treewidth.

Ultimate goal :

Every MSO formula can be certified with $O(\log n)$ bits on bounded treewidth graphs.

But...

- Certify bounded treewidth graphs is open...
  Issue 1 - Diameter of the bags is arbitrary.
  Issue 2 - A vertex can belong to a lot of bags of the tree decomposition.
- ... And even if we assume that we have bounded treewidth, not clear how to do it !

Certifying MSO formulas

Theorem (Courcelle, informal)

Every MSO formula can be decided in FPT time parameterized by the treewidth.

Ultimate goal :

Every MSO formula can be certified with $O(\log n)$ bits on bounded treewidth graphs.

But...

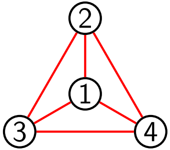
- Certify bounded treewidth graphs is open...
  Issue 1 - Diameter of the bags is arbitrary.
  Issue 2 - A vertex can belong to a lot of bags of the tree decomposition.
- ... And even if we assume that we have bounded treewidth, not clear how to do it !

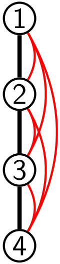
[Fraigniaud et al. '21]

- 3-approximation of tw in $O(\log^2 n)$ bits.
- MSO-certification on bounded treewidth graphs with $O(\log^2 n)$ bits.

One step back : Treedepth

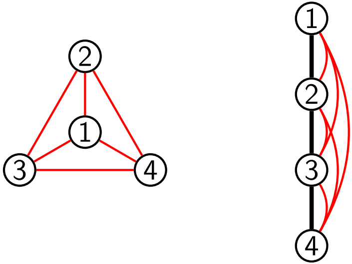
Find a (rooted) tree T on $|V(G)|$ vertices such that each edge of G links a vertex with an ancestor.
$\mathrm{td}(G)$ = minimum depth of such a T





One step back : Treedepth

Find a (rooted) tree T on $|V(G)|$ vertices such that each edge of G links a vertex with an ancestor.
$\mathrm{td}(G) =$ minimum depth of such a T



Bounded treedepth $\Rightarrow$ No long path

Zoology of depths

$$\mathrm{tw}(G) \leq \mathrm{pw}(G) \leq \mathrm{td}(G)$$

Advantages of treedepth :

- Diameter is bounded.
- [Gajarský, Hlinený '16] For every graph of bounded schrubdepth we can construct a kernel that satisfies the same FO formulas.

  And bounded schrubdepth $\Rightarrow$ Bounded treedepth.
- For bounded schrubdepth graphs, (informally) FO = MSO.

Certification of treedepth

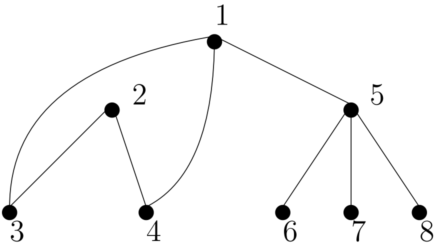
Theorem (B, Feuilloley, Pierron '21+)

$\mathrm{td}(G) \leq k$ can be certified with $O(k \log n)$ bits.

Certification of treedepth

Theorem (B, Feuilloley, Pierron '21+)

$\mathrm{td}(G) \leq k$ can be certified with $O(k \log n)$ bits.

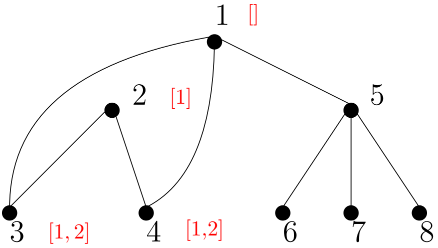


Certificate :

Certification of treedepth

Theorem (B, Feuilloley, Pierron '21+)

$\mathrm{td}(G) \leq k$ can be certified with $O(k \log n)$ bits.



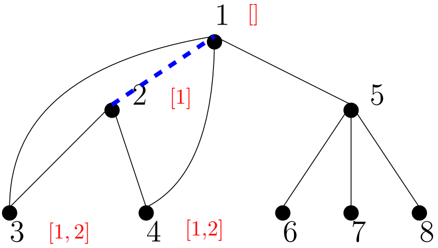
Certificate :

- List of ancestors

Certification of treedepth

Theorem (B, Feuilloley, Pierron '21+)

$\mathrm{td}(G) \leq k$ can be certified with $O(k \log n)$ bits.



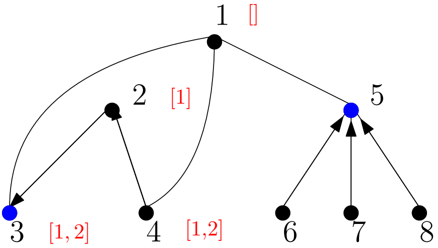
Certificate :

- List of ancestors

Certification of treedepth

Theorem (B, Feuilloley, Pierron '21+)

$\mathrm{td}(G) \leq k$ can be certified with $O(k \log n)$ bits.



Certificate :

- List of ancestors
- Subtree rooted in a node connected to an ancestor for the graph below v for every v.

MSO certification

Theorem

Every MSO formula can be certified with $O(\log n)$ bits on bounded treedepth graphs.

- Existence of a (constructive and easy to certify) kernel for FO formulas on bounded treedepth graphs.

  Existence already guaranteed for [Gajarský and Hlinený '16] on bounded schrubdepth graphs.

MSO certification

Theorem

Every MSO formula can be certified with $O(\log n)$ bits on bounded treedepth graphs.

- Existence of a (constructive and easy to certify) kernel for FO formulas on bounded treedepth graphs.

  Existence already guaranteed for [Gajarský and Hlinený '16] on bounded schrubdepth graphs.

  $\rightarrow$ Prove that it is a kernel with Ehrenfeucht-Fraissé games.

MSO certification

Theorem

Every MSO formula can be certified with $O(\log n)$ bits on bounded treedepth graphs.

- Existence of a (constructive and easy to certify) kernel for FO formulas on bounded treedepth graphs.

  Existence already guaranteed for [Gajarský and Hlinený '16] on bounded schrubdepth graphs.

  $\rightarrow$ Prove that it is a kernel with Ehrenfeucht-Fraissé games.

  $\rightarrow$ Prove that we can certify this kernel.

MSO certification

Theorem

Every MSO formula can be certified with $O(\log n)$ bits on bounded treedepth graphs.

- Existence of a (constructive and easy to certify) kernel for FO formulas on bounded treedepth graphs.

  Existence already guaranteed for [Gajarský and Hlinený '16] on bounded schrubdepth graphs.

  → Prove that it is a kernel with Ehrenfeucht-Fraissé games.

  → Prove that we can certify this kernel.
- FO and MSO are equivalent for bounded treedepth.

MSO certification

Theorem

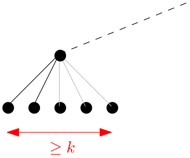
Every MSO formula can be certified with $O(\log n)$ bits on bounded treedepth graphs.

- Existence of a (constructive and easy to certify) kernel for FO formulas on bounded treedepth graphs.

  Existence already guaranteed for [Gajarský and Hlinený '16] on bounded schrubdepth graphs.

  → Prove that it is a kernel with Ehrenfeucht-Fraissé games.

  → Prove that we can certify this kernel.
- FO and MSO are equivalent for bounded treedepth.



MSO certification

Theorem

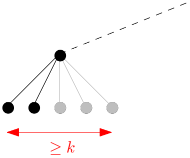
Every MSO formula can be certified with $O(\log n)$ bits on bounded treedepth graphs.

- Existence of a (constructive and easy to certify) kernel for FO formulas on bounded treedepth graphs.

  Existence already guaranteed for [Gajarský and Hlinený '16] on bounded schrubdepth graphs.

  $\rightarrow$ Prove that it is a kernel with Ehrenfeucht-Fraissé games.

  $\rightarrow$ Prove that we can certify this kernel.
- FO and MSO are equivalent for bounded treedepth.



MSO certification

Theorem

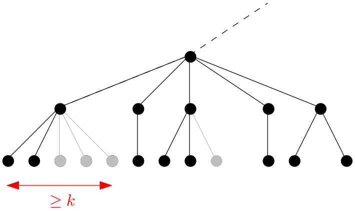
Every MSO formula can be certified with $O(\log n)$ bits on bounded treedepth graphs.

- Existence of a (constructive and easy to certify) kernel for FO formulas on bounded treedepth graphs.

  Existence already guaranteed for [Gajarský and Hlinený '16] on bounded schrubdepth graphs.

  $\rightarrow$ Prove that it is a kernel with Ehrenfeucht-Fraissé games.

  $\rightarrow$ Prove that we can certify this kernel.
- FO and MSO are equivalent for bounded treedepth.



MSO certification

Theorem

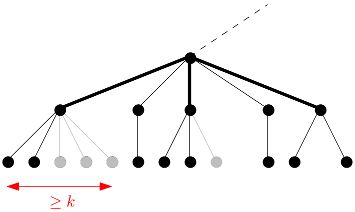
Every MSO formula can be certified with $O(\log n)$ bits on bounded treedepth graphs.

- Existence of a (constructive and easy to certify) kernel for FO formulas on bounded treedepth graphs.

  Existence already guaranteed for [Gajarský and Hlinený '16] on bounded schrubdepth graphs.

  $\rightarrow$ Prove that it is a kernel with Ehrenfeucht-Fraissé games.

  $\rightarrow$ Prove that we can certify this kernel.
- FO and MSO are equivalent for bounded treedepth.



MSO certification

Theorem

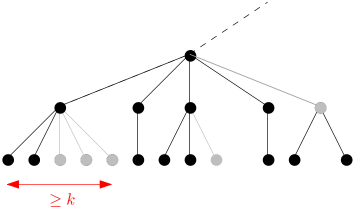
Every MSO formula can be certified with $O(\log n)$ bits on bounded treedepth graphs.

- Existence of a (constructive and easy to certify) kernel for FO formulas on bounded treedepth graphs.

  Existence already guaranteed for [Gajarský and Hlinený '16] on bounded schrubdepth graphs.

  $\rightarrow$ Prove that it is a kernel with Ehrenfeucht-Fraïssé games.

  $\rightarrow$ Prove that we can certify this kernel.
- FO and MSO are equivalent for bounded treedepth.



Extension to treewidth

Theorem [Fraigniaud et al. '21+]

- 3-approximation of tw in $O(\log^2 n)$ bits.
- MSO-certification on bounded treewidth graphs with $O(\log^2 n)$ bits.

Extension to treewidth

Theorem [Fraigniaud et al. '21+]

- 3-approximation of tw in $O(\log^2 n)$ bits.
- MSO-certification on bounded treewidth graphs with $O(\log^2 n)$ bits.

Sketch of the proof :

- [Bodlaender] There exists a tree-decomposition of depth at most $O(\log n)$ where bags have size at most $3k + 2$.

Extension to treewidth

Theorem [Fraigniaud et al. '21+]

- 3-approximation of tw in $O(\log^2 n)$ bits.
- MSO-certification on bounded treewidth graphs with $O(\log^2 n)$ bits.

Sketch of the proof :

- [Bodlaender] There exists a tree-decomposition of depth at most $O(\log n)$ where bags have size at most $3k + 2$.
- We can assume that the subtree below a node is connected.

Extension to treewidth

Theorem [Fraigniaud et al. '21+]

- 3-approximation of tw in $O(\log^2 n)$ bits.
- MSO-certification on bounded treewidth graphs with $O(\log^2 n)$ bits.

Sketch of the proof :

- [Bodlaender] There exists a tree-decomposition of depth at most $O(\log n)$ where bags have size at most $3k + 2$.
- We can assume that the subtree below a node is connected.
- We can assume that the set of "important" bags a vertex belong to is logarithmic.

$\rightarrow$ With techniques of the same flavour $\Rightarrow$ $O(\log^2 n)$ certificates.

Extension to treewidth

Theorem [Fraigniaud et al. '21+]

- 3-approximation of tw in $O(\log^2 n)$ bits.
- MSO-certification on bounded treewidth graphs with $O(\log^2 n)$ bits.

Sketch of the proof :

- [Bodlaender] There exists a tree-decomposition of depth at most $O(\log n)$ where bags have size at most $3k + 2$.
- We can assume that the subtree below a node is connected.
- We can assume that the set of "important" bags a vertex belong to is logarithmic.

$\rightarrow$ With techniques of the same flavour $\Rightarrow$ $O(\log^2 n)$ certificates.

The proof technique for MSO seem a bit different.

Can we go further ?

Can we certify

- treewidth ? A minor-approximation ?

Probably easier than H-minor free graphs.

Can we go further ?

Can we certify

- treewidth ? A minor-approximation ?

  Probably easier than H-minor free graphs.

- schrub-depth ?

  Generalization of treedepth to dense graphs.

Can we go further ?

Can we certify

- treewidth ? A minor-approximation ?

  Probably easier than H-minor free graphs.

- schrub-depth ?

  Generalization of treedepth to dense graphs.

- Δ-freeness ?

  Detecting a Δ is “complicated” in the CONGEST model.

Can we go further ?

Can we certify

- treewidth ? A minor-approximation ?

  Probably easier than H-minor free graphs.

- schrub-depth ?

  Generalization of treedepth to dense graphs.

- Δ-freeness ?

  Detecting a Δ is "complicated" in the CONGEST model.

Can we go further ?

Can we certify

- treewidth ? A minor-approximation ?

  Probably easier than H-minor free graphs.

- schrub-depth ?

  Generalization of treedepth to dense graphs.

- Δ-freeness ?

  Detecting a Δ is "complicated" in the CONGEST model.

Thanks for your attention !