On the complexity of computing the k-restricted edge-connectivity of a graph

Luis P. Montejano[1] Ignasi Sau[2]

Séminaire AIGCo
Montpellier, 11 juin 2015

[1] Département de Mathématiques, Université de Montpellier 2, France

[2] CNRS, LIRMM, Montpellier, France

Outline of the talk

Next section is...

Edge-connectivity

- We consider undirected simple graphs without loops or multiple edges.
- A set $S \subseteq E(G)$ of a graph G is an edge-cut if $G - S$ is disconnected.
- The edge-connectivity $\lambda(G)$ is defined as

$$\lambda(G) = \min\{|S| : S \subseteq E(G) \text{ is an edge-cut}\}.$$

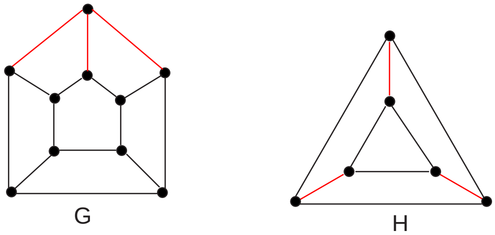
- $\lambda(G)$ can be computed in poly time by a Max Flow algorithm.

Edge-connectivity and minimum degree

- Clearly, $\lambda(G) \leq \delta(G)$, where $\delta(G)$ is the minimum degree of G.

Edge-connectivity and minimum degree

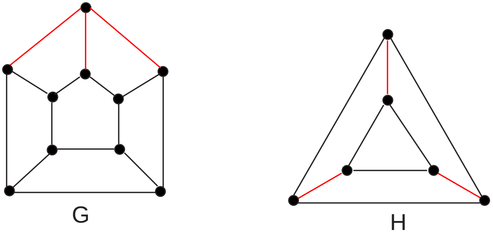
- Clearly, $\lambda(G) \leq \delta(G)$, where $\delta(G)$ is the minimum degree of G.
- A graph G is maximally edge-connected if $\lambda(G) = \delta(G)$.



$$\lambda(G) = \delta(G) = 3 = \lambda(H) = \delta(H).$$

Edge-connectivity and minimum degree

- Clearly, $\lambda(G) \leq \delta(G)$, where $\delta(G)$ is the minimum degree of G.
- A graph G is maximally edge-connected if $\lambda(G) = \delta(G)$.



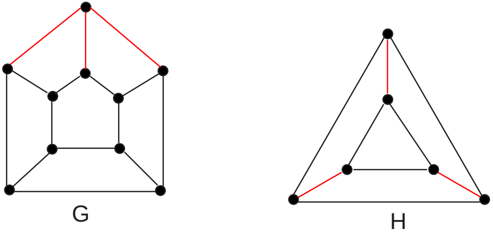
$$\lambda(G) = \delta(G) = 3 = \lambda(H) = \delta(H).$$

- A graph G is superconnected if every minimum edge-cut consists of the edges adjacent to one vertex.

Edge-connectivity and minimum degree

- Clearly, $\lambda(G) \leq \delta(G)$, where $\delta(G)$ is the minimum degree of G.

- A graph G is maximally edge-connected if $\lambda(G) = \delta(G)$.



G is superconnected while H is not.

- A graph G is superconnected if every minimum edge-cut consists of the edges adjacent to one vertex.

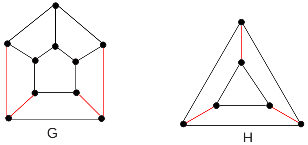
Restricted edge-connectivity

Definition [Esfahanian and Hakimi '88]

An edge-cut S is a restricted edge-cut if every component of $G - S$ has at least 2 vertices.

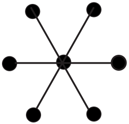
The restricted edge-connectivity $\lambda_2(G)$ of a graph G is defined as

$$\lambda_2(G) = \min\{|S| : S \subseteq E(G) \text{ is a restricted edge-cut}\}.$$



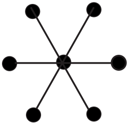
$\lambda_2(G) = 4$ and $\lambda_2(H) = 3$.

Restricted edge-connectivity



λ_2 is not defined for this graph.

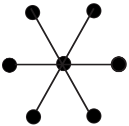
Restricted edge-connectivity



λ_2 is not defined for this graph.

A connected graph G is called λ_2-connected if $\lambda_2(G)$ exists.

Restricted edge-connectivity



λ_2 is not defined for this graph.

A connected graph G is called λ_2-connected if $\lambda_2(G)$ exists.

Theorem [Esfahanian and Hakimi '88]

Every connected graph G that is not a star is λ_2-connected and satisfies $\lambda_2(G) \leq \xi(G)$.

Where $\xi(G) = \min\{d(u) + d(v) - 2 : uv \in E(G)\} \geq 2\delta(G) - 2$.

k-restricted edge-connectivity

In 1994, Fàbrega and Fiol proposed the concept of k-restricted edge-connectivity, where k is a positive integer.

Definition [Fàbrega and Fiol '94]

An edge cut S is a k-restricted edge cut if every component of $G - S$ has at least k vertices.

The k-restricted edge-connectivity $\lambda_k(G)$ of a graph G is defined as

$$\lambda_k(G) = \min\{|S| : S \subseteq E(G) \text{ is a k-restricted edge-cut}\}.$$

k-restricted edge-connectivity

In 1994, Fàbrega and Fiol proposed the concept of k-restricted edge-connectivity, where k is a positive integer.

Definition [Fàbrega and Fiol '94]

An edge cut S is a k-restricted edge cut if every component of $G - S$ has at least k vertices.

The k-restricted edge-connectivity $\lambda_k(G)$ of a graph G is defined as

$$\lambda_k(G) = \min\{|S| : S \subseteq E(G) \text{ is a k-restricted edge-cut}\}.$$

A connected graph G is called λ_k-connected if $\lambda_k(G)$ exists.

k-restricted edge-connectivity

In 1994, Fàbrega and Fiol proposed the concept of k-restricted edge-connectivity, where k is a positive integer.

Definition [Fàbrega and Fiol '94]

An edge cut S is a k-restricted edge cut if every component of $G - S$ has at least k vertices.

The k-restricted edge-connectivity $\lambda_k(G)$ of a graph G is defined as

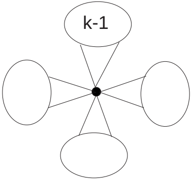
$$\lambda_k(G) = \min\{|S| : S \subseteq E(G) \text{ is a k-restricted edge-cut}\}.$$

A connected graph G is called λ_k-connected if $\lambda_k(G)$ exists.

For any k-restricted cut S of size $\lambda_k(G)$, the graph $G - S$ has exactly two connected components.

k-restricted edge-connectivity

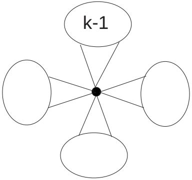
A k-flower is a graph containing a cut vertex u such that every component of $G - u$ has at most $k - 1$ vertices.



λ_k is not defined for k-flowers.

k-restricted edge-connectivity

A k-flower is a graph containing a cut vertex u such that every component of $G - u$ has at most $k-1$ vertices.



λ_k is not defined for k-flowers.

Theorem [Zhang and Yuan '05]

Every connected graph G that is not a k-flower with $k-1 \leq \delta(G)$ is λ_k-connected and satisfies $\lambda_k(G) \leq \xi_k(G)$, where $\xi_k(G) = \min\{|\partial(X)| : |V(X)| = k \text{ and } G[X] \text{ is connected }\}$.

For a set $X \subseteq V(G)$, we denote by $\partial(X)$ the set of edges leaving X.

Then, $\xi_1(G) = \delta(G)$ and $\xi_2(G) = \xi(G)$.

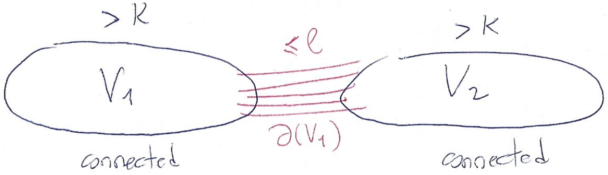
A lot of combinatorial results about λ_k

- Introduction of λ_2: [Esfahanian, Hakimi '88]
- Introduction of λ_k: [Fàbrega and Fiol '94]
- Case $k = 3$: [Bonsma, Ueffing, Volkmann. '02]
- General bounds on λ_k: [Zhang, Yuan '05]
- λ_k in graphs of large girth: [Balbuena, Carmona, Fàbrega, Fiol '97]
- λ_k in triangle-free graphs: [Yuan, Liu '10] [Holtkamp, Meierling, Montejano '12]

Meanwhile, in the parameterized complexity community...

Chitnis, Cygan, Hajiaghayi, and Pilipczuk[2] defined in 2012 this notion:

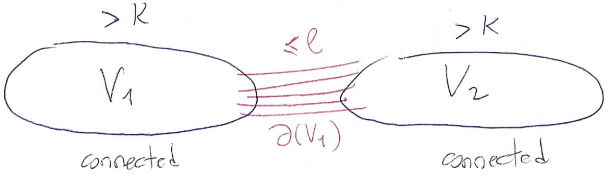
- Let G be a connected graph. A partition (V_1, V_2) of $V(G)$ is a (k, ℓ)-separation if $|V_1|, |V_2| > k$, $|\partial(V_1)| \leqslant \ell$, and $G[V_1]$ and $G[V_2]$ are both connected.



Meanwhile, in the parameterized complexity community...

Chitnis, Cygan, Hajiaghayi, and Pilipczuk[2] defined in 2012 this notion:

- Let G be a connected graph. A partition (V_1, V_2) of $V(G)$ is a (k, ℓ)-separation if $|V_1|, |V_2| > k$, $|\partial(V_1)| \leqslant \ell$, and $G[V_1]$ and $G[V_2]$ are both connected.

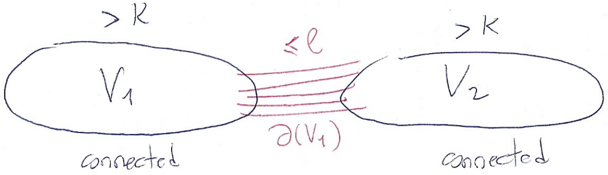


- A graph is (k, ℓ)-connected if it does not have a $(k, \ell - 1)$-separation.

Meanwhile, in the parameterized complexity community...

Chitnis, Cygan, Hajiaghayi, and Pilipczuk[2] defined in 2012 this notion:

- Let G be a connected graph. A partition (V_1, V_2) of $V(G)$ is a (k, ℓ)-separation if $|V_1|, |V_2| > k$, $|\partial(V_1)| \leqslant \ell$, and $G[V_1]$ and $G[V_2]$ are both connected.



- A graph is (k, ℓ)-connected if it does not have a $(k, \ell - 1)$-separation.

★ $\boxed{\text{Both notions are essentially the same!}}$

$\lambda_k(G) \leqslant \ell$ if and only if G admits a $(k-1, \ell)$-separation.

(k, ℓ)-separations are useful for FPT algorithms

Used in a technique known as recursive understanding:

- FPT algorithms for cut problems. [Chitnis, Cygan, Hajiaghayi, Pilipczuk^2 '12]
- A similar notion existed for vertex-cuts. [Kawarabayashi, Thorup '11]
- This technique has proved very useful. [Cygan, Lokshtanov, Pilipczuk^2, Saurabh '14]

  [Kim, Oum, Paul, S., Thilikos '15]

(k, ℓ)-separations are useful for FPT algorithms

Used in a technique known as recursive understanding:

- FPT algorithms for cut problems. [Chitnis, Cygan, Hajiaghayi, Pilipczuk^2 '12]
- A similar notion existed for vertex-cuts. [Kawarabayashi, Thorup '11]
- This technique has proved very useful. [Cygan, Lokshtanov, Pilipczuk^2, Saurabh '14] [Kim, Oum, Paul, S., Thilikos '15]

$\star$ Only one known algorithmic result about (k, ℓ)-separations:

Lemma (Chitnis, Cygan, Hajiaghayi, Pilipczuk^2 '12)

There exists an algorithm that given a n-vertex connected graph G and two integers k, ℓ, either finds a (k, ℓ)-separation, or reports that no such separation exists, in time $(k + \ell)^{O(\min\{k,\ell\})} n^3 \log n$.

(k,ℓ)-separations are useful for FPT algorithms

Used in a technique known as recursive understanding:

- FPT algorithms for cut problems. [Chitnis, Cygan, Hajiaghayi, Pilipczuk^2 '12]
- A similar notion existed for vertex-cuts. [Kawarabayashi, Thorup '11]
- This technique has proved very useful. [Cygan, Lokshtanov, Pilipczuk^2, Saurabh '14] [Kim, Oum, Paul, S., Thilikos '15]

$\star$ Only one known algorithmic result about (k,ℓ)-separations:

Lemma (Chitnis, Cygan, Hajiaghayi, Pilipczuk^2 '12)

There exists an algorithm that given a n-vertex connected graph G and two integers k,ℓ, either finds a (k,ℓ)-separation, or reports that no such separation exists, in time $(k+\ell)^{O(\min\{k,\ell\})}n^3 \log n$.

$\star$ We initiate a systematic study of the complexity of computing the k-restricted edge-connectivity of a graph.

Next section is...

Summary of our results

Problem	Classical complexity	Parameterized complexity with parameter				
		$k+\ell$	k	ℓ	$k+\Delta$	$\ell+\Delta$
Is G λ_k-conn. ?	NPc, even if $\Delta \leqslant 5$	$\star$	FPT	$\star$	FPT	$\star$
$\lambda_k(G) \leqslant \ell$?	NPh, even if G is λ_k-conn.	FPT (known)	W[1]-hard	No poly kernels	FPT	?

Table: Summary of our results, where Δ denotes the maximum degree of the input graph G, and NPc (resp. NPh) stands for NP-complete (resp. NP-hard). The symbol '$\star$' denotes that the problem is not defined for that parameter.

Next section is...

Some words on parameterized complexity

- $\boxed{\text{Idea}}$ given an NP-hard problem with input size n, fix one parameter k of the input to see whether the problem gets more "tractable".

  Example: the size of a VERTEX COVER.

- Given a (NP-hard) problem with input of size n and a parameter k, a fixed-parameter tractable (FPT) algorithm runs in time

$$f(k) \cdot n^{O(1)}, \quad \text{for some function } f.$$

  Examples: k-VERTEX COVER, k-LONGEST PATH.

Determining whether a graph is λ_k-connected is hard

- Given a graph G, if n is even and $k = n/2$, it is NP-complete to determine whether G contains two vertex-disjoint connected subgraphs of order $n/2$ each. [Dyer, Frieze '85]

Determining whether a graph is λ_k-connected is hard

- Given a graph G, if n is even and $k = n/2$, it is NP-complete to determine whether G contains two vertex-disjoint connected subgraphs of order $n/2$ each. [Dyer, Frieze '85]

- This implies that the following problem is NP-hard:

RESTRICTED EDGE-CONNECTIVITY (REC)	
Instance:	A connected graph $G = (V, E)$ and an integer k.
Output:	$\lambda_k(G)$, or a report that G is not λ_k-connected.

Determining whether a graph is λ_k-connected is hard

- Given a graph G, if n is even and $k = n/2$, it is NP-complete to determine whether G contains two vertex-disjoint connected subgraphs of order $n/2$ each. [Dyer, Frieze '85]

- This implies that the following problem is NP-hard:

RESTRICTED EDGE-CONNECTIVITY (REC)	
Instance:	A connected graph $G = (V, E)$ and an integer k.
Output:	$\lambda_k(G)$, or a report that G is not λ_k-connected.

- Even if the input graph G is guaranteed to be λ_k-connected, computing $\lambda_k(G)$ remains hard:

Theorem

The REC problem is NP-hard restricted to λ_k-connected graphs.

Proof of the Theorem

- Proof for n even and $k = n/2$. Reduction from Minimum Bisection in connected 3-regular graphs, which is NP-hard. [Berman, Karpinski '02]

Proof of the Theorem

- Proof for n even and $k = n/2$. Reduction from MINIMUM BISECTION in connected 3-regular graphs, which is NP-hard. [Berman, Karpinski '02]
- Given G, we build G' by adding to G two non-adjacent universal vertices v_1 and v_2. Note that G' is $\lambda_{n/2}$-connected.

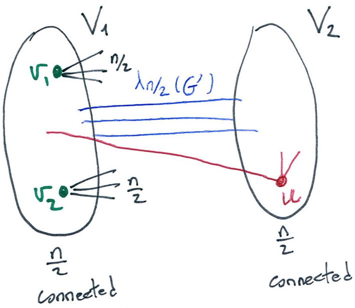
Proof of the Theorem

- Proof for n even and $k = n/2$. Reduction from MINIMUM BISECTION in connected 3-regular graphs, which is NP-hard. [Berman, Karpinski '02]
- Given G, we build G' by adding to G two non-adjacent universal vertices v_1 and v_2. Note that G' is $\lambda_{n/2}$-connected.

$\boxed{\text{Claim}}$ v_1 and v_2 belong to different parts in any optimal solution in G'.

- Proof for n even and $k = n/2$. Reduction from MINIMUM BISECTION in connected 3-regular graphs, which is NP-hard. [Berman, Karpinski '02]
- Given G, we build G' by adding to G two non-adjacent universal vertices v_1 and v_2. Note that G' is $\lambda_{n/2}$-connected.

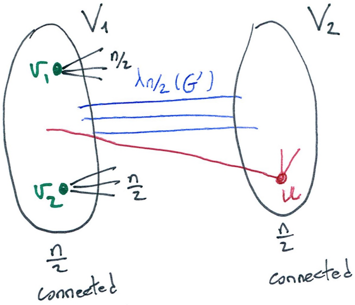
$\boxed{\text{Claim}}$ v_1 and v_2 belong to different parts in any optimal solution in G'.

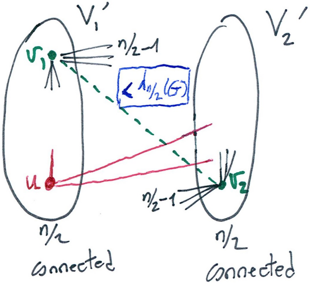


Proof of the Theorem

- Proof for n even and $k = n/2$. Reduction from MINIMUM BISECTION in connected 3-regular graphs, which is NP-hard. [Berman, Karpinski '02]
- Given G, we build G' by adding to G two non-adjacent universal vertices v_1 and v_2. Note that G' is $\lambda_{n/2}$-connected.

$\boxed{\text{Claim}}$ v_1 and v_2 belong to different parts in any optimal solution in G'.

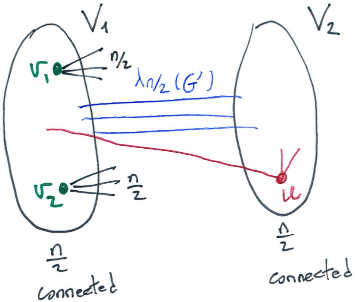


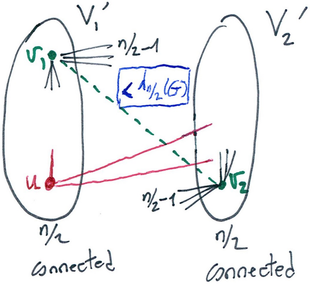


Proof of the Theorem

- Proof for n even and $k = n/2$. Reduction from Minimum Bisection in connected 3-regular graphs, which is NP-hard. [Berman, Karpinski '02]
- Given G, we build G' by adding to G two non-adjacent universal vertices v_1 and v_2. Note that G' is $\lambda_{n/2}$-connected.

Claim v_1 and v_2 belong to different parts in any optimal solution in G'.





- Thus, REC problem in $G' \equiv$ Minimum Bisection problem in G.

A parameterized analysis of the REC problem

Since the REC problem is NP-hard, we parameterize it:

Parameterized Restricted Edge-connectivity (**p**-REC)

Instance: A connected graph G and two integers k and ℓ.
Question: $\lambda_k(G) \leqslant \ell$?
Parameter 1: The integers k and ℓ.
Parameter 2: The integer k.
Parameter 3: The integer ℓ.

A parameterized analysis of the REC problem

Since the REC problem is NP-hard, we parameterize it:

Parameterized Restricted Edge-connectivity (**p**-REC)

Instance:	A connected graph G and two integers k and ℓ.
Question:	$\lambda_k(G) \leqslant \ell$?
Parameter 1:	The integers k and ℓ.
Parameter 2:	The integer k.
Parameter 3:	The integer ℓ.

The **p**-REC problem is FPT when parameterized by both k and ℓ:

Theorem (Chitnis, Cygan, Hajiaghayi, Pilipczuk^2 '12)

There exists an algorithm that given a n-vertex connected graph G and two integers k, ℓ, either finds a (k, ℓ)-separation, or reports that no such separation exists, in time $(k+\ell)^{O(\min\{k,\ell\})} n^3 \log n$.

W[1]-hardness with parameter k only

Theorem

The **p**-REC *problem is W[1]-hard when parameterized by k.*

It is easy to see that the problem is in XP: solvable in time $n^{O(k)}$.

W[1]-hardness with parameter k only

Theorem

The **p**-REC *problem is W[1]-hard when parameterized by k.*

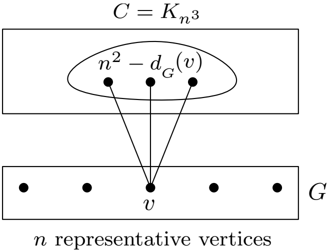
- Reduction from k-Clique: the same as the one for Cutting k Vertices from a Graph, only the analysis changes. [Downey *et al.* '03]

W[1]-hardness with parameter k only

Theorem

The **p**-REC *problem is W[1]-hard when parameterized by k.*

- Reduction from k-CLIQUE: the same as the one for CUTTING k VERTICES FROM A GRAPH, only the analysis changes. [Downey *et al.* '03]
- Given $G \to G'$:

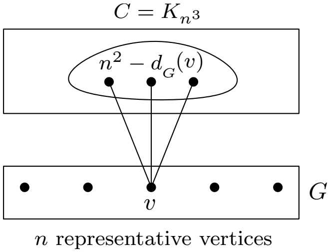


W[1]-hardness with parameter k only

Theorem

The **p**-REC *problem is W[1]-hard when parameterized by k.*

- Reduction from k-Clique: the same as the one for Cutting k Vertices from a Graph, only the analysis changes. [Downey *et al.* '03]
- Given $G \to G'$:



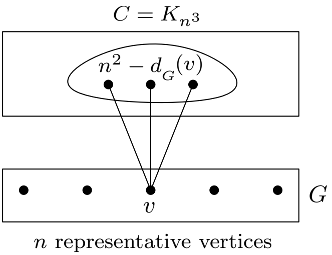
- Consider $\ell = kn^2 - 2\binom{k}{2}$ and take $k \leq n/2$.

W[1]-hardness with parameter k only

Theorem

The **p**-REC *problem is W[1]-hard when parameterized by k.*

- Reduction from k-Clique: the same as the one for Cutting k Vertices from a Graph, only the analysis changes. [Downey *et al.* '03]
- Given $G \to G'$:



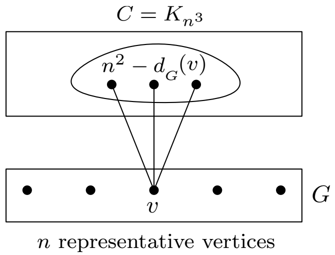
- Consider $\ell = kn^2 - 2\binom{k}{2}$ and take $k \leq n/2$.
- $\boxed{\text{Claim 1}}$ If $K \subseteq V(G)$ is a k-clique in G, then $|\partial(K)| = \ell$.

W[1]-hardness with parameter k only

Theorem

The **p**-REC *problem is W[1]-hard when parameterized by k.*

- Reduction from k-Clique: the same as the one for Cutting k Vertices from a Graph, only the analysis changes. [Downey *et al.* '03]
- Given $G \to G'$:



- Consider $\ell = kn^2 - 2\binom{k}{2}$ and take $k \leq n/2$.
- Claim 1: If $K \subseteq V(G)$ is a k-clique in G, then $|\partial(K)| = \ell$.
- Claim 2: If $K \subseteq V(G')$ such that $G[K]$ and $G' - K$ are connected, $|K| \geq k$, $|V(G') \setminus K| \geq k$, and $|\partial(K)| \leq \ell$, then ...

Let's recap...

Theorem

Given a graph G and a positive integer k, determining whether G is λ_k-connected is NP-*hard*.

Let's recap...

Theorem

Given a graph G and a positive integer k, determining whether G is λ_k-connected is NP-hard.

Theorem

Given a graph G and two integers k, ℓ such that G is λ_k-connected, determining whether $\lambda_k(G) \leqslant \ell$ is W[1]-hard when parameterized by k.

Let's recap...

Theorem

Given a graph G and a positive integer k, determining whether G is λ_k-connected is NP-hard.

Theorem

Given a graph G and two integers k, ℓ such that G is λ_k-connected, determining whether $\lambda_k(G) \leqslant \ell$ is W[1]-hard when parameterized by k.

Theorem

Given a graph G and a positive integer k, determining whether G is λ_k-connected is FPT when parameterized by k.

The proof is based on a simple application of the technique of splitters.

Let's recap...

Theorem

Given a graph G and a positive integer k, determining whether G is λ_k-connected is NP-hard.

Theorem

Given a graph G and two integers k, ℓ such that G is λ_k-connected, determining whether $\lambda_k(G) \leqslant \ell$ is W[1]-hard when parameterized by k.

Theorem

Given a graph G and a positive integer k, determining whether G is λ_k-connected is FPT when parameterized by k.

The proof is based on a simple application of the technique of splitters.

★ Parameterized complexity of $\boxed{\lambda_k(G) \leqslant \ell?}$ with parameter ℓ?

Parameterized complexity with parameter ℓ

Theorem

The **p**-REC *problem does not admit polynomial kernels when parameterized by* ℓ*, unless* *coNP* $\subseteq$ *NP*/*poly*.

Parameterized complexity with parameter ℓ

Theorem

The $\mathbf{p}$-REC *problem does not admit polynomial kernels when parameterized by* ℓ*, unless* $coNP \subseteq NP/poly$*.*

- A kernel for a parameterized problem Π is an algorithm that given (x, k) outputs, in time polynomial in $|x| + k$, an instance (x', k') s.t.:
  - ⋆ $(x, k) \in \Pi$ if and only if $(x', k') \in \Pi$, and
  - ⋆ Both $\boxed{|x'|, k' \leqslant g(k)}$, where g is some computable function.

Parameterized complexity with parameter ℓ

Theorem

The $\mathbf{p}$-REC *problem does not admit polynomial kernels when parameterized by* ℓ*, unless* $coNP \subseteq NP/poly$.

- A kernel for a parameterized problem Π is an algorithm that given (x, k) outputs, in time polynomial in $|x| + k$, an instance (x', k') s.t.:
  - ⋆ $(x, k) \in \Pi$ if and only if $(x', k') \in \Pi$, and
  - ⋆ Both $\boxed{|x'|, k' \leqslant g(k)}$, where g is some computable function.
- If $g(k) = k^{O(1)}$: we say that Π admits a polynomial kernel.

Parameterized complexity with parameter ℓ

Theorem

The **p**-REC *problem does not admit polynomial kernels when parameterized by* ℓ, *unless* $coNP \subseteq NP/poly$.

- A kernel for a parameterized problem Π is an algorithm that given (x, k) outputs, in time polynomial in $|x| + k$, an instance (x', k') s.t.:
  - $(x, k) \in \Pi$ if and only if $(x', k') \in \Pi$, and
  - Both $\boxed{|x'|, k' \leqslant g(k)}$, where g is some computable function.
- If $g(k) = k^{O(1)}$: we say that Π admits a polynomial kernel.
- Folklore result: $\boxed{\Pi \text{ is } \mathrm{FPT} \Leftrightarrow \Pi \text{ admits a kernel}}$

Parameterized complexity with parameter ℓ

Theorem

The **p**-REC *problem does not admit polynomial kernels when parameterized by ℓ, unless coNP $\subseteq$ NP/poly.*

- A kernel for a parameterized problem Π is an algorithm that given (x, k) outputs, in time polynomial in $|x| + k$, an instance (x', k') s.t.:
  - ⋆ $(x, k) \in \Pi$ if and only if $(x', k') \in \Pi$, and
  - ⋆ Both $\boxed{|x'|, k' \leqslant g(k)}$, where g is some computable function.
- If $g(k) = k^{O(1)}$: we say that Π admits a polynomial kernel.
- Folklore result: $\boxed{\Pi \text{ is FPT} \Leftrightarrow \Pi \text{ admits a kernel}}$
- Question: which FPT problems admit polynomial kernels?

Parameterized complexity with parameter ℓ

Theorem

The **p**-REC *problem does not admit polynomial kernels when parameterized by ℓ, unless coNP $\subseteq$ NP/poly.*

- A kernel for a parameterized problem Π is an algorithm that given (x, k) outputs, in time polynomial in $|x| + k$, an instance (x', k') s.t.:
  - ⋆ $(x, k) \in \Pi$ if and only if $(x', k') \in \Pi$, and
  - ⋆ Both $\boxed{|x'|, k' \leqslant g(k)}$, where g is some computable function.
- If $g(k) = k^{O(1)}$: we say that Π admits a polynomial kernel.
- Folklore result: $\boxed{\Pi \text{ is FPT} \;\Leftrightarrow\; \Pi \text{ admits a kernel}}$
- Question: which FPT problems admit polynomial kernels?
- It is possible to prove that polynomial kernels are unlikely to exist.

[Bodlaender, Downey, Fellows, Hermelin '08] [Bodlaender, Thomassé, Yeo '09] [Bodlaender, Jansen, Kratsch '11]

Non-existence of polynomial kernels with parameter ℓ

- The proof is inspired by the one to prove that the MIN BISECTION does not admit polynomial kernels. [van Bevern *et al.* '13]
- Main difference: both parts left out by the edge-cut are connected.

Non-existence of polynomial kernels with parameter ℓ

- The proof is inspired by the one to prove that the Min Bisection does not admit polynomial kernels. [van Bevern *et al.* '13]
- Main difference: both parts left out by the edge-cut are connected.
- We use the technique of cross-composition [Bodlaender, Jansen, Kratsch '11]

Non-existence of polynomial kernels with parameter ℓ

- The proof is inspired by the one to prove that the MIN BISECTION does not admit polynomial kernels. [van Bevern *et al.* '13]
- Main difference: both parts left out by the edge-cut are connected.
- We use the technique of $\boxed{\text{cross-composition}}$ [Bodlaender, Jansen, Kratsch '11]

  Cross-composition from MAX CUT (which is NP-hard) to EDGE-WEIGHTED **p**-REC parameterized by ℓ is a poly-time algorithm that, given t instances $(G_1, p_1), \ldots, (G_t, p_t)$ of MAX CUT, constructs one instance (G^*, k, ℓ) of EDGE-WEIGHTED **p**-REC such that:

  1. (G^*, k, ℓ) is YES iff one of the t instances of MAX CUT is YES, and
  2. ℓ is polynomially bounded as a function of $\max_{1 \leqslant i \leqslant t} |V(G_i)|$.

Non-existence of polynomial kernels with parameter ℓ

- The proof is inspired by the one to prove that the MIN BISECTION does not admit polynomial kernels. [van Bevern *et al.* '13]
- Main difference: both parts left out by the edge-cut are connected.
- We use the technique of $\boxed{\text{cross-composition}}$ [Bodlaender, Jansen, Kratsch '11]

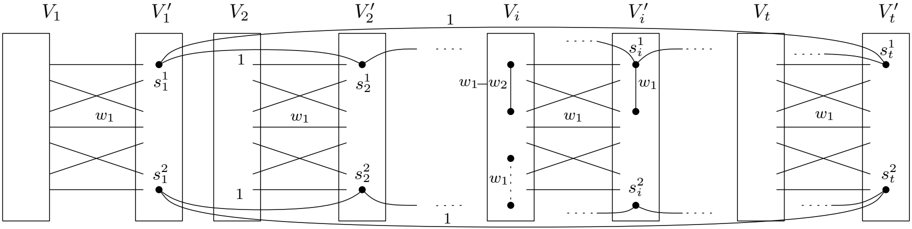
  Cross-composition from MAX CUT (which is NP-hard) to EDGE-WEIGHTED **p**-REC parameterized by ℓ is a poly-time algorithm that, given t instances $(G_1, p_1), \ldots, (G_t, p_t)$ of MAX CUT, constructs one instance (G^*, k, ℓ) of EDGE-WEIGHTED **p**-REC such that:

  1. (G^*, k, ℓ) is YES iff one of the t instances of MAX CUT is YES, and
  2. ℓ is polynomially bounded as a function of $\max_{1 \leqslant i \leqslant t} |V(G_i)|$.

- We may safely assume that t is odd, that for each $1 \leqslant i \leqslant t$ we have $|V(G_i)| =: n$ and $p_i =: p$, and that $1 \leqslant p \leqslant n^2$.

Idea of the proof

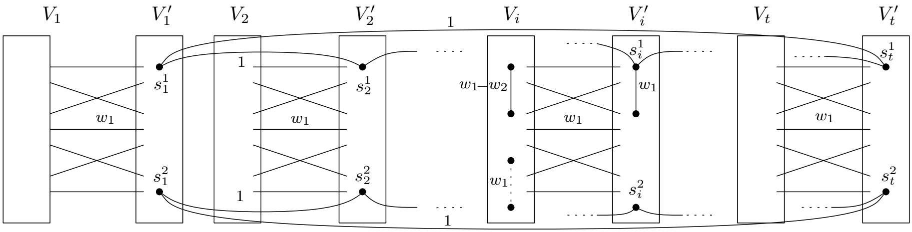
Given $(G_1, p), \ldots, (G_t, p)$, we create G^* as follows:



- We define $w_1 := 5n^2$ and $w_2 := 5$.
- And we set $k := |V(G^*)|/2$ and $\ell := w_1 n^2 - w_2 p + 4$.

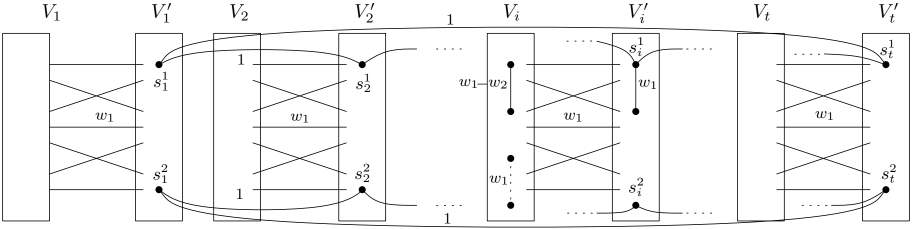
Idea of the proof

Given $(G_1, p), \ldots, (G_t, p)$, we create G^* as follows:



- We define $w_1 := 5n^2$ and $w_2 := 5$.
- And we set $k := |V(G^*)|/2$ and $\ell := w_1 n^2 - w_2 p + 4$.
- k is not polynomially bounded in terms of n, but this is not a problem since the parameter is ℓ, which is bounded by $5n^4$.
- This construction can be performed in polynomial time in $t \cdot n$.

Given $(G_1, p), \ldots, (G_t, p)$, we create G^* as follows:



- We define $w_1 := 5n^2$ and $w_2 := 5$.
- And we set $k := |V(G^*)|/2$ and $\ell := w_1 n^2 - w_2 p + 4$.
- k is not polynomially bounded in terms of n, but this is not a problem since the parameter is ℓ, which is bounded by $5n^4$.
- This construction can be performed in polynomial time in $t \cdot n$.

$\boxed{\text{Claim}}$ (G^*, k, ℓ) is a YES-instance of EDGE-WEIGHTED **p**-REC iff there exists $i \in \{1, \ldots, t\}$ such that (G_i, p) is a YES-instance of MAX CUT.

Considering the maximum degree as a parameter

Considering the $\Delta(G)$ as an extra parameter turns the problem easier?

Considering the maximum degree as a parameter

Considering the $\Delta(G)$ as an extra parameter turns the problem easier?

Theorem

Determining whether a connected graph G is λ_k-connected is NP-complete when k is part of the input, even if $\Delta(G) \leqslant 5$.

Considering the maximum degree as a parameter

Considering the $\Delta(G)$ as an extra parameter turns the problem easier?

Theorem

Determining whether a connected graph G is λ_k-connected is NP-complete when k is part of the input, even if $\Delta(G) \leqslant 5$.

Theorem

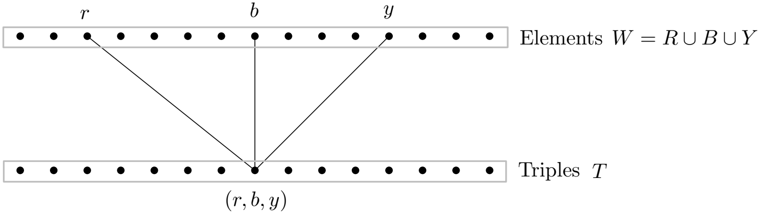
The **p**-REC *problem is* FPT *when parameterized by k and the maximum degree Δ of the input graph.*

Algorithm based on a simple exhaustive search + Min Cut algorithm.

Idea of the NP-completeness reduction

- Reduction from the 3-DIMENSIONAL MATCHING (3DM) problem:

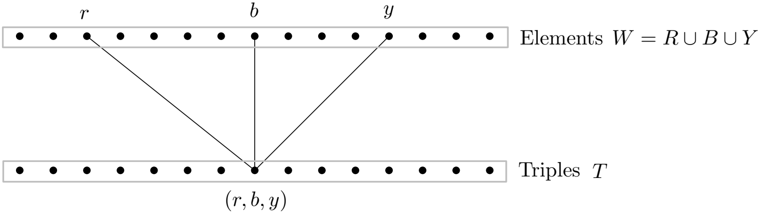
  Given a set $W = R \cup B \cup Y$, where R, B, Y are disjoint sets with $|R| = |B| = |Y| = m$, and a set of triples $T \subseteq R \times B \times Y$, the question is whether there exists a matching $M \subseteq T$ covering W, i.e., $|M| = m$ and each element of $W = R \cup B \cup Y$ occurs in exactly one triple of M.



Idea of the NP-completeness reduction

- Reduction from the 3-DIMENSIONAL MATCHING (3DM) problem:

  Given a set $W = R \cup B \cup Y$, where R, B, Y are disjoint sets with $|R| = |B| = |Y| = m$, and a set of triples $T \subseteq R \times B \times Y$, the question is whether there exists a matching $M \subseteq T$ covering W, i.e., $|M| = m$ and each element of $W = R \cup B \cup Y$ occurs in exactly one triple of M.

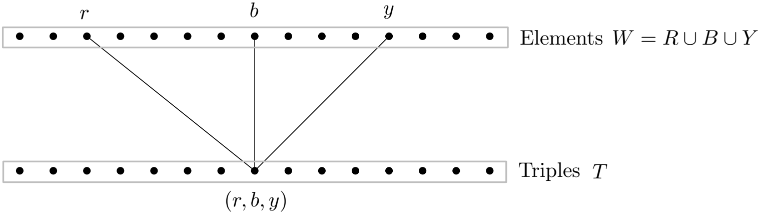


- 3DM is NP-complete even if each element of W appears in 2 or 3 triples only. [Dyer, Frieze '86]

Idea of the NP-completeness reduction

- Reduction from the 3-Dimensional Matching (3DM) problem:

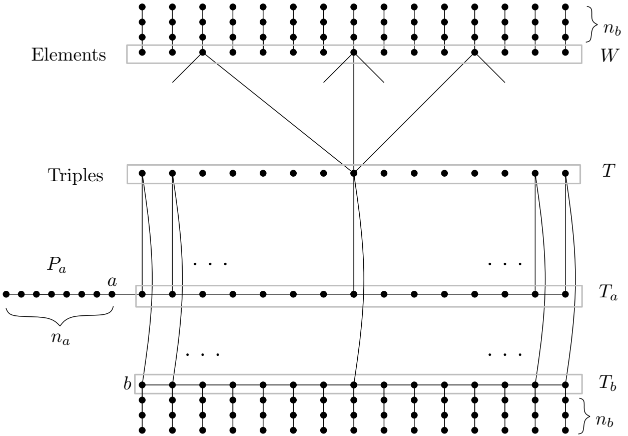
  Given a set $W = R \cup B \cup Y$, where R, B, Y are disjoint sets with $|R| = |B| = |Y| = m$, and a set of triples $T \subseteq R \times B \times Y$, the question is whether there exists a matching $M \subseteq T$ covering W, i.e., $|M| = m$ and each element of $W = R \cup B \cup Y$ occurs in exactly one triple of M.



- 3DM is NP-complete even if each element of W appears in 2 or 3 triples only. [Dyer, Frieze '86]
- Our reduction is an appropriate modification of one given in [Dyer, Frieze '85]

Idea of the NP-completeness reduction (2)

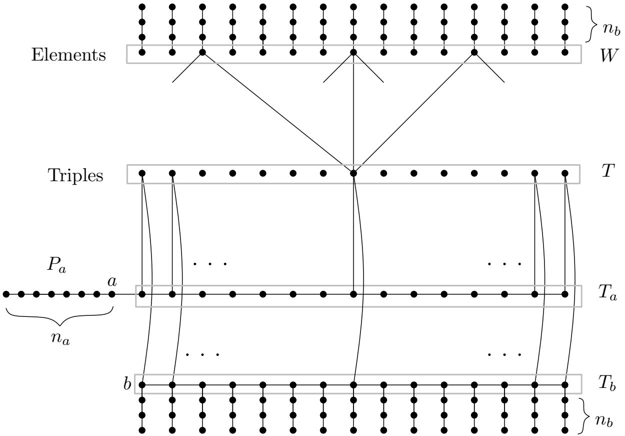
Given an instance (W, T) of 3DM, we build a graph G with $\Delta(G) \leqslant 5$:



Where $n_b = 2m^3$ and $n_a = (3m + |T|)n_b + 5m - |T| - 1$.

Idea of the NP-completeness reduction (2)

Given an instance (W, T) of 3DM, we build a graph G with $\Delta(G) \leqslant 5$:



Where $n_b = 2m^3$ and $n_a = (3m + |T|)n_b + 5m - |T| - 1$.

$\boxed{\text{Claim}}$ G contains two disjoint connected subgraphs of order $n/2 \Leftrightarrow$ T contains a matching covering W.

Next section is...

Conclusions and further research

Problem	Classical complexity	Parameterized complexity with parameter				
		$k+\ell$	k	ℓ	$k+\Delta$	$\ell+\Delta$
Is G λ_k-conn. ?	NPc, even if $\Delta \leqslant 5$	$\star$	FPT	$\star$	FPT	$\star$
$\lambda_k(G) \leqslant \ell$?	NPh, even if G is λ_k-conn.	FPT (known)	W[1]-hard	No poly kernels	FPT	?

Conclusions and further research

Problem	Classical complexity	Parameterized complexity with parameter				
		$k+\ell$	k	ℓ	$k+\Delta$	$\ell+\Delta$
Is G λ_k-conn. ?	NPc, even if $\Delta \leqslant 5$	$\star$	FPT	$\star$	FPT	$\star$
$\lambda_k(G) \leqslant \ell$?	NPh, even if G is λ_k-conn.	FPT (known)	W[1]-hard	No poly kernels	FPT	?

- Main open question: is the problem FPT when parameterized by ℓ?

Conclusions and further research

Problem	Classical complexity	Parameterized complexity with parameter				
		$k+\ell$	k	ℓ	$k+\Delta$	$\ell+\Delta$
Is G λ_k-conn. ?	NPc, even if $\Delta \leqslant 5$	$\star$	FPT	$\star$	FPT	$\star$
$\lambda_k(G) \leqslant \ell$?	NPh, even if G is λ_k-conn.	FPT (known)	W[1]-hard	No poly kernels	FPT	?

- Main open question: is the problem FPT when parameterized by ℓ?

  For Min Bisection, the non-existence of polynomial kernels was known before the problem was recently proved to be FPT. [Cygan *et al.* '14]

Conclusions and further research

Problem	Classical complexity	Parameterized complexity with parameter				
		$k+\ell$	k	ℓ	$k+\Delta$	$\ell+\Delta$
Is G λ_k-conn. ?	NPc, even if $\Delta \leqslant 5$	$\star$	FPT	$\star$	FPT	$\star$
$\lambda_k(G) \leqslant \ell$?	NPh, even if G is λ_k-conn.	FPT (known)	W[1]-hard	No poly kernels	FPT	?

- Main open question: is the problem FPT when parameterized by ℓ?

  For Min Bisection, the non-existence of polynomial kernels was known before the problem was recently proved to be FPT. [Cygan *et al.* '14]

- We don't know it even for the combined parameter $\ell + \Delta$.

  Adding Δ as a parameter may not make things easier, as Min Bisection is as hard in 3-regular graphs as in general graphs. [Berman, Karpinski '02]

Conclusions and further research

Problem	Classical complexity	Parameterized complexity with parameter				
		$k+\ell$	k	ℓ	$k+\Delta$	$\ell+\Delta$
Is G λ_k-conn. ?	NPc, even if $\Delta \leqslant 5$	$\star$	FPT	$\star$	FPT	$\star$
$\lambda_k(G) \leqslant \ell$?	NPh, even if G is λ_k-conn.	FPT (known)	W[1]-hard	No poly kernels	FPT	?

- Main open question: is the problem FPT when parameterized by ℓ?

  For MIN BISECTION, the non-existence of polynomial kernels was known before the problem was recently proved to be FPT. [Cygan *et al.* '14]

- We don't know it even for the combined parameter $\ell + \Delta$.

  Adding Δ as a parameter may not make things easier, as MIN BISECTION is as hard in 3-regular graphs as in general graphs. [Berman, Karpinski '02]

- Polynomial kernels with parameter $k+\ell$?

Gràcies!