Sparse graphs with bounded induced cycle packing number have logarithmic treewidth

Marthe Bonamy[1] Édouard Bonnet[2] Hugues Déprés[2] Louis Esperet[3]
Colin Geniet[2] Claire Hilaire[1] Stéphan Thomassé[2] Alexandra Wesolek[4]

[1]Univ. Bordeaux, France

[2]ENS de Lyon, France

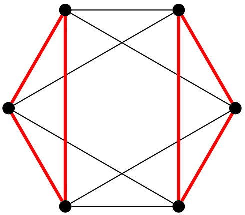
[3]Univ. Grenoble, France

[4]Simon Fraser Univ., Canada

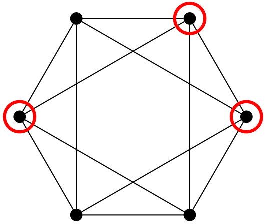
Symposium on Discrete Algorithms 2023

Erdős Pósa property

Cycle packing $cp(G)$:
maximum size of a collection of vertex-disjoint cycles in G.



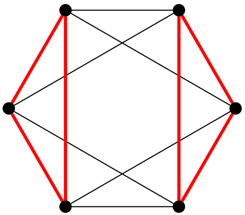
Feedback-vertex-set $fvs(G)$:
minimum number of vertices required to intersect all cycles of G.

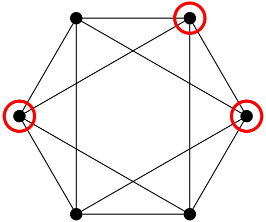


Erdős Pósa property

Cycle packing $cp(G)$:
maximum size of a collection of vertex-disjoint cycles in G.

Feedback-vertex-set $fvs(G)$:
minimum number of vertices required to intersect all cycles of G.





Theorem (Erdős–Pósa)

$$fvs(G) \leq f(cp(G)) \text{ with } f(k) = O(k \log k)$$

A graph with $cp(G)$ bounded is a tree plus a bounded number of vertices.
$\Rightarrow$ algorithmically simple graphs

Odd cycles packing

Odd cycle packing $ocp(G)$: same with only odd cycles.

Theorem (Fiorini, Joret, Weltge, Yuditsky, '21)

In graphs with $ocp(G) \leq k$, MAXIMUM INDEPENDENT SET *can be solved in polynomial time.*

Odd cycles packing

Odd cycle packing $ocp(G)$: same with only odd cycles.

Theorem (Fiorini, Joret, Weltge, Yuditsky, '21)

In graphs with $ocp(G) \leq k$, MAXIMUM INDEPENDENT SET *can be solved in polynomial time.*

Induced odd cycle packing $iocp(G)$: only consider packings of non-adjacent odd cycles.

Theorem (Bonamy et al., '18)

For graphs with $iocp(G)$ bounded, VC-dimension bounded, and linear size independent sets, there is an EPTAS for MAXIMUM INDEPENDENT SET.

Applications to disk and unit ball graphs.

Induced cycle packing

Induced cycle packing $icp(G)$:
maximum number of vertex-disjoint *and non-adjacent* cycles in G.

Induced cycle packing

Induced cycle packing $icp(G)$:
maximum number of vertex-disjoint *and non-adjacent* cycles in G.

We study the class of graphs with $icp(G) \leq k$ (k constant).
Problems:

- Testing $icp(G) \leq k$
- Algorithms in this class for independent set, ...

Induced cycle packing

Induced cycle packing $icp(G)$:
maximum number of vertex-disjoint *and non-adjacent* cycles in G.

We study the class of graphs with $icp(G) \leq k$ (k constant).
Problems:

- Testing $icp(G) \leq k$
- Algorithms in this class for independent set, ...

Question

Does $icp(G) \leq k$ imply $fvs(G) \leq f(k)$?

Induced cycle packing

Induced cycle packing $icp(G)$:
maximum number of vertex-disjoint *and non-adjacent* cycles in G.

We study the class of graphs with $icp(G) \leq k$ (k constant).
Problems:

- Testing $icp(G) \leq k$
- Algorithms in this class for independent set, ...

Question

Does $icp(G) \leq k$ imply $fvs(G) \leq f(k)$?

No: cliques have $icp(K_t) = 1$ but $fvs(K_t) = t - 2$.

Induced cycle packing

Induced cycle packing $icp(G)$:
maximum number of vertex-disjoint *and non-adjacent* cycles in G.

We study the class of graphs with $icp(G) \leq k$ (k constant).
Problems:

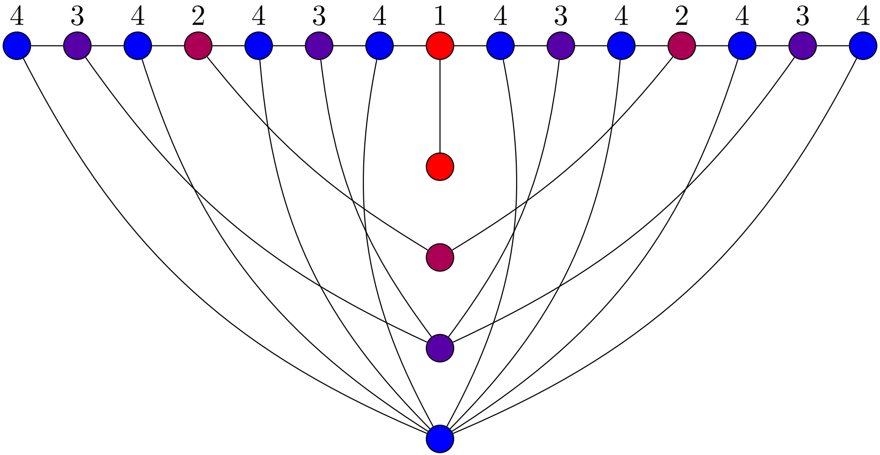
- Testing $icp(G) \leq k$
- Algorithms in this class for independent set, ...

Question

Does $icp(G) \leq k$ and no $K_{t,t}$ subgraph imply $fvs(G) \leq f(k,t)$?

Still no!

$icp(G) = 1$ and FVS unbounded.



Feedback vertex set is logarithmic

Theorem

If G is a graph with $icp(G) \leq k$ and without $K_{t,t}$ subgraph, then

$$fvs(G) \leq f(k,t) \cdot \log n$$

Feedback vertex set is logarithmic

Theorem

If G is a graph with $icp(G) \leq k$ and without $K_{t,t}$ subgraph, then

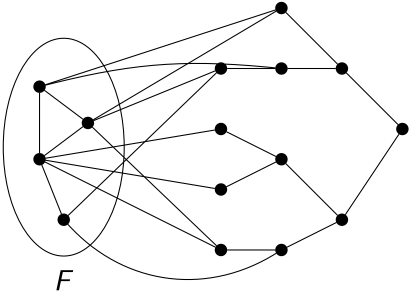
$$fvs(G) \leq f(k,t) \cdot \log n$$

Some problems with algorithms in time $2^{O(tw(G))} \cdot poly(n)$:

- Maximum independent set
- 3-coloring
- Hamiltonian cycle
- ...
- Testing $icp(G) \leq k$ [Mi. Pilipczuk, '22]

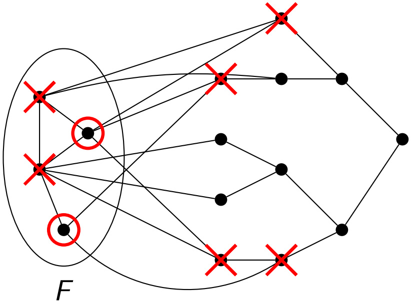
When $fvs(G)$ is logarithmic in n, these algorithms are polynomial.

Solving Maximum Independent Set



F feedback vertex set of size $O(\log n)$.

Solving Maximum Independent Set

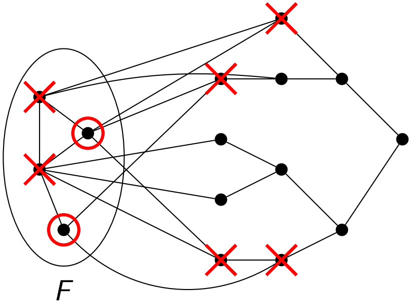


F feedback vertex set of size $O(\log n)$.

We construct I independent.
For each $v \in F$, branch on v:

- either pick $v \in I$, and delete $N[v]$,
- or $v \notin I$, and delete v.

Solving Maximum Independent Set



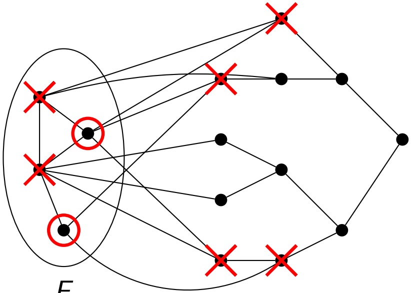
F feedback vertex set of size $O(\log n)$.

We construct I independent.
For each $v \in F$, branch on v:

- either pick $v \in I$, and delete $N[v]$,
- or $v \notin I$, and delete v.

After this, only a forest is left $\Rightarrow$ pick leafs greedily.

Solving Maximum Independent Set



F feedback vertex set of size $O(\log n)$.

We construct I independent.
For each $v \in F$, branch on v:

- either pick $v \in I$, and delete $N[v]$,
- or $v \notin I$, and delete v.

After this, only a forest is left $\Rightarrow$ pick leafs greedily.

Branching is polynomial because F is logarithmic.

Solving MIS in the dense case

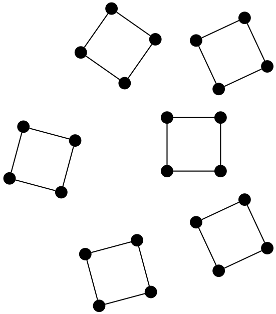
Theorem

For any fixed k, MAXIMUM INDEPENDENT SET can be solved in quasipolynomial time $n^{O(\log n)}$ on graphs with $icp(G) \leq k$.

Solving MIS in the dense case

Theorem

For any fixed k, Maximum Independent Set can be solved in quasipolynomial time $n^{O(\log n)}$ on graphs with $icp(G) \leq k$.

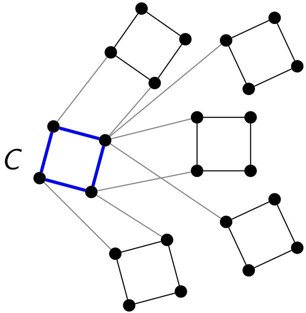


Let S the set of cycles with length 4.

Solving MIS in the dense case

Theorem

For any fixed k, Maximum Independent Set *can be solved in quasipolynomial time $n^{O(\log n)}$ on graphs with icp$(G) \leq k$.*



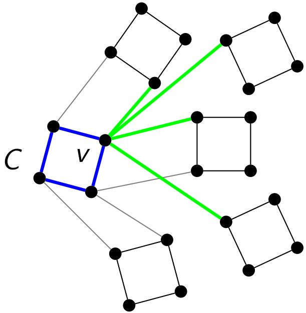
Let S the set of cycles with length 4.

Fix $C \in S$. All cycles are adjacent to C.

Solving MIS in the dense case

Theorem

For any fixed k, Maximum Independent Set *can be solved in quasipolynomial time $n^{O(\log n)}$ on graphs with $icp(G) \leq k$.*



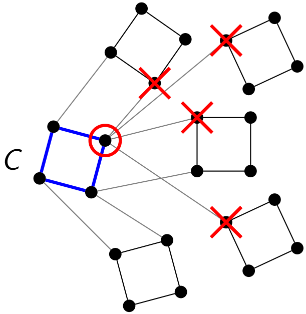
Let S the set of cycles with length 4.

Fix $C \in S$. All cycles are adjacent to C.
Thus some $v \in C$ is adjacent to $1/4$ of the cycles of S.

Solving MIS in the dense case

Theorem

For any fixed k, Maximum Independent Set *can be solved in quasipolynomial time* $n^{O(\log n)}$ *on graphs with* $icp(G) \leq k$.



Let S the set of cycles with length 4.

Fix $C \in S$. All cycles are adjacent to C.
Thus some $v \in C$ is adjacent to $1/4$ of the cycles of S.

Branch on v:

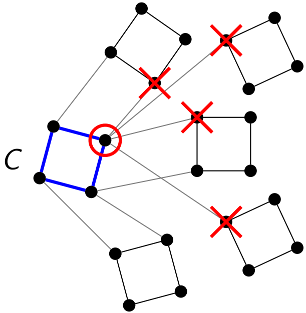
- Take v and delete $N(v) \Rightarrow$ destroys $1/4$ cycles in S,
- Delete $v \Rightarrow$ destroys C

This kind of branching is quasipolynomial.

Solving MIS in the dense case

Theorem

For any fixed k, Maximum Independent Set *can be solved in quasipolynomial time $n^{O(\log n)}$ on graphs with $icp(G) \leq k$.*



Let S the set of cycles with length 4.

Fix $C \in S$. All cycles are adjacent to C.
Thus some $v \in C$ is adjacent to $1/4$ of the cycles of S.

Branch on v:

- Take v and delete $N(v) \Rightarrow$ destroys $1/4$ cycles in S,
- Delete $v \Rightarrow$ destroys C

This kind of branching is quasipolynomial.

When $S = \varnothing$, we are in the $K_{2,2}$-free case.

Back to the main theorem

Theorem

If G is a graph with $icp(G) \leq k$ and without $K_{t,t}$ subgraph, then $fvs(G) \leq f(k,t) \cdot \log n$.

Back to the main theorem

Theorem

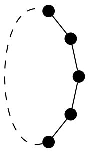
If G is a graph with $icp(G) \leq k$ and with girth > 10, then G has average degree $\leq 2k+2$.

Back to the main theorem

Theorem

If G is a graph with $icp(G) \leq k$ and with girth > 10, then G has average degree $\leq 2k+2$.

Pick C cycle with minimal length.

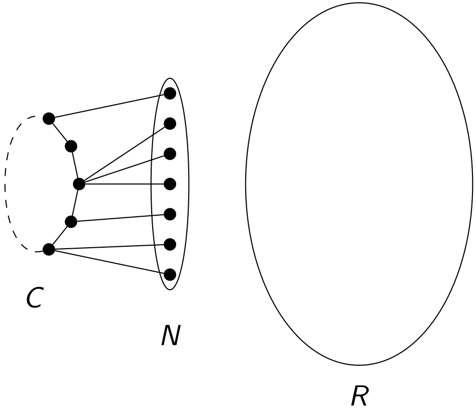


C

Back to the main theorem

Theorem

If G is a graph with $icp(G) \leq k$ and with girth > 10, then G has average degree $\leq 2k + 2$.

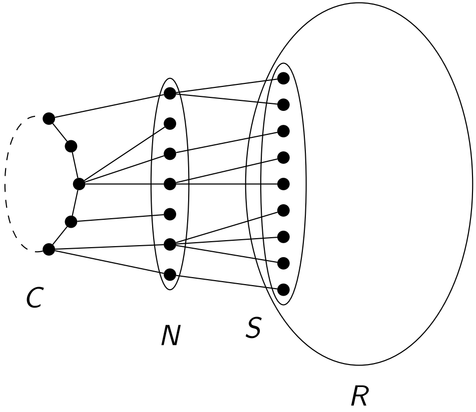


Pick C cycle with minimal length.
Let N its neighbourhood, $R = G \setminus (C \cup N)$,

Back to the main theorem

Theorem

If G is a graph with $icp(G) \leq k$ and with girth > 10, then G has average degree $\leq 2k + 2$.

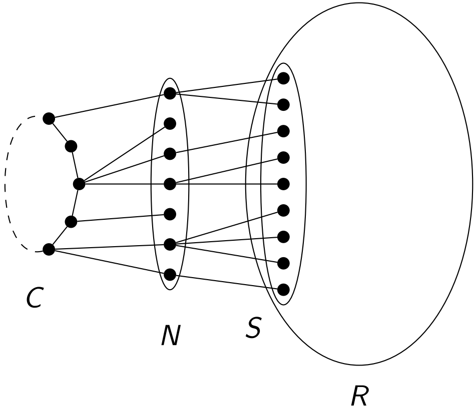


Pick C cycle with minimal length.
Let N its neighbourhood, $R = G \setminus (C \cup N)$, and S the second neighbourhood of C.

Back to the main theorem

Theorem

If G is a graph with $icp(G) \leq k$ and with girth > 10, then G has average degree $\leq 2k+2$.



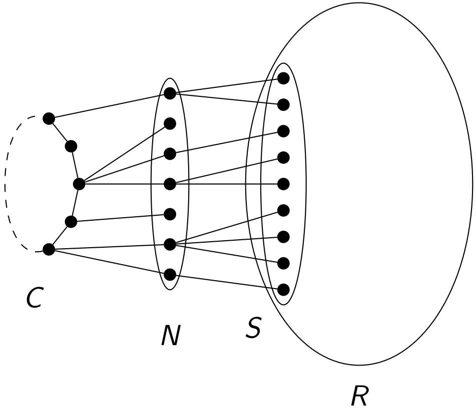
Pick C cycle with minimal length.
Let N its neighbourhood, $R = G \setminus (C \cup N)$, and S the second neighbourhood of C.

C is the only cycle in $G[C \cup N \cup S]$, otherwise C would not be minimal
$\Rightarrow$ average degree ≤ 2.

Back to the main theorem

Theorem

If G is a graph with $icp(G) \leq k$ and with girth > 10, then G has average degree $\leq 2k+2$.



Pick C cycle with minimal length.
Let N its neighbourhood, $R = G \setminus (C \cup N)$,
and S the second neighbourhood of C.

C is the only cycle in $G[C \cup N \cup S]$,
otherwise C would not be minimal
$\Rightarrow$ average degree ≤ 2.

$G[R]$ is disjoint from C, so $icp(G[R]) \leq k-1$.
$\Rightarrow$ average degree $\leq 2k$ by induction.

Summary

For graphs with $icp(G) \leq k$ and no $K_{t,t}$ subgraph: (sparse setting)

- Feedback vertex set is logarithmic + tight up to the constant.

Summary

For graphs with $icp(G) \leq k$ and no $K_{t,t}$ subgraph: (sparse setting)

- Feedback vertex set is logarithmic + tight up to the constant.
- Polynomial algorithm for independent set, and many other problems.
- Polynomial algorithm to compute $icp(G)$.

Summary

For graphs with $icp(G) \leq k$ and no $K_{t,t}$ subgraph: (sparse setting)

- Feedback vertex set is logarithmic + tight up to the constant.
- Polynomial algorithm for independent set, and many other problems.
- Polynomial algorithm to compute $icp(G)$.

For graphs with $icp(G) \leq k$: (dense setting)

- Quasi-polynomial algorithms for independent set and 3-coloring.

Summary

For graphs with $icp(G) \leq k$ and no $K_{t,t}$ subgraph: (sparse setting)

- Feedback vertex set is logarithmic + tight up to the constant.
- Polynomial algorithm for independent set, and many other problems.
- Polynomial algorithm to compute $icp(G)$.

For graphs with $icp(G) \leq k$: (dense setting)

- Quasi-polynomial algorithms for independent set and 3-coloring.

Related result:

Theorem (Nguyen, Scott, Seymour + Le, '22)

In graphs with $icp(G) \leq k$, there are at most $|V(G)|^{f(k)}$ induced paths.

Implies a polynomial algorithm to test $icp(G) \leq k$.

Open Questions

- In the dense settings, can quasi-polynomial algorithms be improved to be polynomial?
- Any FPT algorithms with $icp(G)$ as parameter?
- What about restricting packing of specific types of cycles?
  (E.g., packing nonadjacent induced cycles of length ≥ 4.)

Open Questions

- In the dense settings, can quasi-polynomial algorithms be improved to be polynomial?
- Any FPT algorithms with $icp(G)$ as parameter?
- What about restricting packing of specific types of cycles?
  (E.g., packing nonadjacent induced cycles of length ≥ 4.)

Thank you!