Elimination Distance to Dominated Clusters

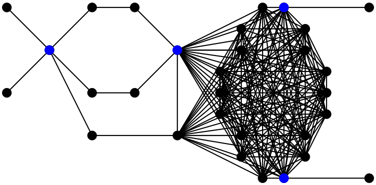
Nicole Schirrmacher
(joint work with Sebastian Siebertz and Alexandre Vigny)

LoGAlg 2025

November 21, 2025

Dominating Set

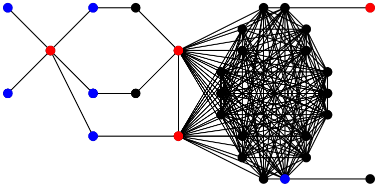
- vertex set $D \in V(G)$
- every vertex is in D or has a neighbor in D



$d = 4$

Dominated Cluster Deletion

- given an undirected graph G and integers k and d
- can we delete k vertices such that every remaining connected component has a dominating set of size at most d



$k = 4$, $d = 1$

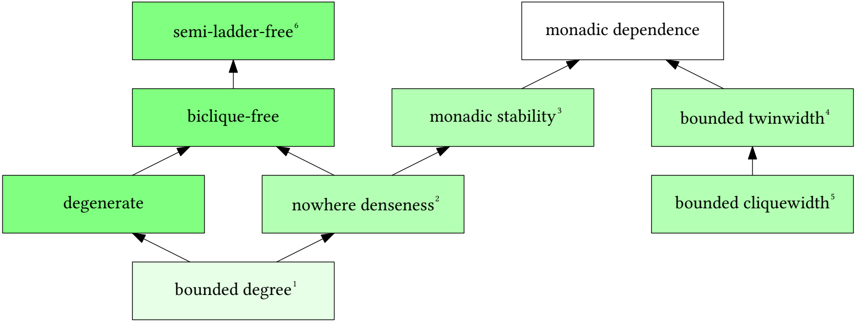
Dominated Cluster Deletion

Theorem [Bentert, Fellows, Golovach, Rosamond, Saurabh, 24]

The Dominated Cluster Deletion problem can be solved in time $f(k, d) \cdot n^{\mathcal{O}(d)}$ for a function f depending on k and d.

- fpt by $k + d + \Delta$ (maximum degree Δ)
- fpt by $k + d + c$ (degeneracy c) left as open question

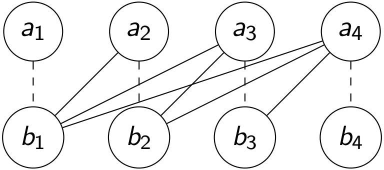
Graph Classes Overview



[1]Bentert, Fellows, Golovach, Rosamond, Saurabh, 24
[2]Grohe, Kreutzer, Siebertz, 17
[3]Dreier, Eleftheriadis, Mählmann, McCarty, Pilipczuk, Torunczyk, 23
[4]Bonnet, Kim, Thomassé, Watrigan, 21
[5]Courcelle, Makowsky, Rotics, 00
[6]Schirrmacher, Siebertz, Vigny, 25

Semi-Ladder

- $2n$ distinct vertices $a_1, \ldots, a_n, b_1, \ldots, b_n$
- $\{a_i, b_j\} \in E(G)$ for all $i, j \leq n$ with $i > j$
- $\{a_i, b_i\} \notin E(G)$ for all $i \leq n$



- a class of graphs is *semi-ladder-free* if there exists a constant ℓ such that the graphs do not contain a semi-ladder of order ℓ

Domination-Type Problems

Theorem [Fabianski, Pilipczuk, Siebertz, Torunczyk, 19]

Let $\mathscr{C}$ be a class of graphs with semi-ladder index ℓ and let δ be a domination-type problem. Then, there is an algorithm that solves the domination-type problem δ on graphs G from $\mathscr{C}$ in time $f(\ell, |\delta|) \cdot m$.

- Partial Domination Set
  - vertex set $D \in V(G)$ and integer k
  - every vertex is either one of at most k deleted vertices or dominated by D

Main Results

Theorem 1 [S., Siebertz, Vigny, 25]

The Dominated Cluster Deletion problem can be solved in time $f(k, d, \ell) \cdot n^{\mathcal{O}(1)}$ for a computable function f where ℓ is the semi-ladder index of the input graph.

Theorem 2 [S., Siebertz, Vigny, 25]

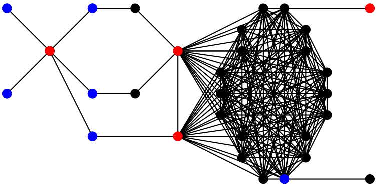
The Elimination Distance to Dominated Clusters problem can be solved in time $f(k, d, \ell) \cdot n^{\mathcal{O}(1)}$ for a computable function f where ℓ is the semi-ladder index of the input graph.

Dominated Cluster Problems

- DOMINATED CLUSTER DELETION
  - given an undirected graph G and integers k and d
  - can we delete k vertices such that every remaining connected component has a dominating set of size d
- ELIMINATION DISTANCE TO DOMINATED CLUSTERS
  - given an undirected graph G and integers k and d
  - can we *recursively* delete vertices up to depth k such that every remaining connected component has a dominating set of size at most d

Elimination Distance to Dominated Clusters

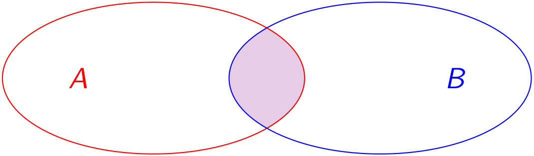
- given an undirected graph G and integers k and d
- can we *recursively* delete vertices up to depth k such that every remaining connected component has a dominating set of size at most d



$k = 3$, $d = 1$

Unbreakable Graphs

- separation of G: pair (A, B) of vertex subsets such that $A \cup B = V(G)$ and there are no edges between $A - B$ and $B - A$

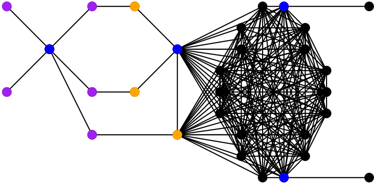


- a graph G is (q, k)-unbreakable if there is no separation (A, B) of order at most k such that $|A| \geq q$ and $|B| \geq q$

Elimination Distance to Dominated Clusters on Unbreakable Graphs

- skeleton: vertices of the solution that break the graph in connected components

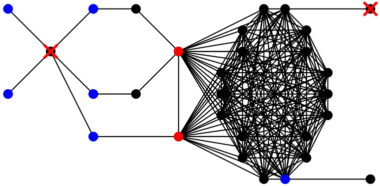
  X dominating set of size at most $q + d$
  Y neighborhood of the small degree vertices of X
  Z neighborhood of the small degree vertices of Y



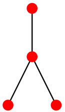
$q = 12$, $k = 3$, $d = 1$

Elimination Distance to Dominated Clusters on Unbreakable Graphs

- pick vertices for the skeleton
- compute (partial) dominating set for the large connected component
- brute-force in small connected components
- guess the tree order of the skeleton



$q = 12$, $k = 3$, $d = 1$



Dynamic Programming

Theorem [Cygan, Lokshtanov, Pilipczuk, Pilipczuk, Saurabh, 19]

For every k, there exists $q = f(k)$ such that for every G, there is a tree decomposition of G such that:

- Every bag is (q, k)-unbreakable in the subgraph G' of G induced by the union of all descendant bags.
- Adjacent bags have intersection of size at most q.

Such a decomposition is computable in time $f(k) \cdot nm$.

Elimination Distance to Dominated Clusters

- unbreakable graphs
  - compute skeleton
  - compute (partial) dominating sets
- dynamic programming
  - unbreakable tree decomposition
  - annotated versions of the dominated cluster problems

Theorem [S., Siebertz, Vigny, 25]

The Elimination Distance to Dominated Clusters problem can be solved in time $f(k, d, \ell) \cdot n^{\mathcal{O}(1)}$ for a computable function f where ℓ is the semi-ladder index of the input graph.

Treedepth

Conjecture [S., Siebertz, Vigny, 25]

Treedepth is NP-hard on some graph class with bounded maximum degree.

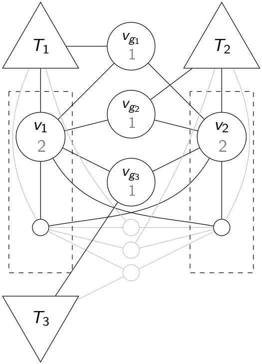
Theorem [Dirks, S., Siebertz, Vigny, 25]

Weighted treedepth is NP-hard on graph classes with maximum degree 10.

Weighted Treedepth on Bounded Degree Graphs

Theorem [Dirks, S., Siebertz, Vigny, 25]

Weighted treedepth is NP-hard on graph classes with maximum degree 10.



reduction from the vertex cover problem on cubic graphs

Conclusion

Theorem [S., Siebertz, Vigny, 25]

The Elimination Distance to Dominated Clusters problem can be solved in time $f(k, d, \ell) \cdot n^{\mathcal{O}(1)}$ for a computable function f where ℓ is the semi-ladder index of the input graph.

Conjecture [S., Siebertz, Vigny, 25]

Treedepth is NP-hard on some graph class with bounded maximum degree.

Thank you very much for your attention!