Problems in NP can Admit Double-Exponential Lower Bounds when Parameterized by Treewidth or Vertex Cover

Florent Foucaud, Esther Galby, Liana Khazaliya,
Shaohua Li, Fionn Mc Inerney, Roohani Sharma, Prafullkumar Tale

June 4, 2024

Part 1.

(In)tractability and Treewidth

Intractable problems and approaches

Fixed-parameter tractability is a framework to deal with intractable problems:

- Choose a complexity parameter k independent of the input size n
- Find an OPT solution in time $f(k) \cdot n^{\mathcal{O}(1)}$ for some function f

Develop algorithms for graphs which are large but have a small solution size

...or simply structured

Intractable problems and approaches

Fixed-parameter tractability is a framework to deal with intractable problems:

- Choose a complexity parameter k independent of the input size n
- Find an OPT solution in time $f(k) \cdot n^{\mathcal{O}(1)}$ for some function f

Develop algorithms for graphs which are large but have a small solution size

...or simply structured

Intractable problems and approaches

Fixed-parameter tractability is a framework to deal with intractable problems:

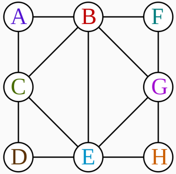
- Choose a complexity parameter k independent of the input size n
- Find an OPT solution in time $f(k) \cdot n^{\mathcal{O}(1)}$ for some function f

Develop algorithms for graphs which are large but have a small solution size

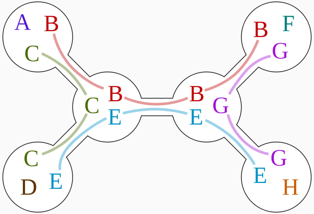
...or simply structured

Treewidth

Def. A tree decomposition of G is a pair $\mathcal{T} = (T, \{X_t\}_{t \in V(T)})$, where T is a tree whose every node t is assigned a vertex subset $X_t \subseteq V(G)$, called a bag, with following conditions:



- $\mathcal{T}1$. $\bigcup_{t \in V(T)} X_t = V(G)$;
- $\mathcal{T}2$. For every $vw \in E(G)$, there exists a node t of T such that bag X_t contains both v and w;
- $\mathcal{T}3$. For every $v \in V(G)$, the set $T_v = \{t \in V(T) | v \in X_t\}$ induces a connected subtree of T.



Def. The width of $\mathcal{T}$ is $\max_{t \in V(T)} |X_t| - 1$.

Def. The treewidth $\text{tw}(G)$ is the **minimum** width over all tree decompositions of G.

Meta-theorems

Many NP-hard problems are FPT parameterized by treewidth via dynamic programming on the tree decomposition.

For a given signature τ, monadic second order logic has

- element-variables $(x, y, z, \dots)$ and set-variables $(X, Y, Z, \dots)$
- relations $=$ (equation) and $x \in X$ (membership), as well as relations from τ
- quantifiers $\exists$ and $\forall$, as well as operators $\wedge$, $\vee$, $\neg$

If φ is a sentence, we write $G \models \varphi$ to indicate that φ holds on G (i.e., G is a model of φ)

Theorem [Courcelle'90]

For a $\mathtt{MSO}_1$ sentence φ and graph G one can decide whether $G \models \varphi$ in time $f(\mathrm{tw}(G), |\varphi|)n$ for some function f.

Conditional Lower Bounds

Exponential Time Hypothesis (ETH) [Impagliazzo, Paturi, 1990]

Roughly, 3-SAT on n variables cannot be solved in time $2^{o(n)}$.

Conditional lower bounds for tw are usually $2^{o(\mathrm{tw})}$, $2^{o(\mathrm{tw} \log \mathrm{tw})}$ or $2^{o(\mathrm{poly}(\mathrm{tw}))}$.

Rarer results: Unless the ETH fails,

- QSAT WITH k ALTERNATIONS admits a lower bound of a tower of exponents of height k in the treewidth of the primal graph [Fichte, Hecher, Pfandler, 2020]
- k-CHOOSABILITY and k-CHOOSABILITY DELETION admit double- and triple-exponential lower bounds in treewidth, respectively [Marx, Mitsou, 2016]
- $\exists\forall$-CSP admits a double-exponential lower bound in the vertex cover number [Lampis, Mitsou, 2017]

Conditional Lower Bounds

Exponential Time Hypothesis (ETH) [Impagliazzo, Paturi, 1990]

Roughly, 3-SAT on n variables cannot be solved in time $2^{o(n)}$.

Conditional lower bounds for tw are usually $2^{o(\mathrm{tw})}$, $2^{o(\mathrm{tw} \log \mathrm{tw})}$ or $2^{o(\mathrm{poly}(\mathrm{tw}))}$.

Rarer results: Unless the ETH fails,

- QSAT WITH k ALTERNATIONS admits a lower bound of a tower of exponents of height k in the treewidth of the primal graph PSPACE-complete [Fichte, Hecher, Pfandler, 2020]
- k-CHOOSABILITY and k-CHOOSABILITY DELETION admit double- and triple-exponential lower bounds in treewidth, respectively Π_2^p-complete and Σ_3^p-complete [Marx, Mitsou, 2016]
- $\exists\forall$-CSP admits a double-exponential lower bound in the vertex cover number Σ_2^p-complete [Lampis, Mitsou, 2017]

Part 2.

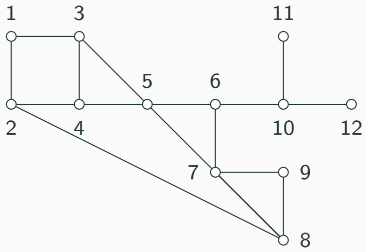
Distance-Based Graph Problem(s)

Metric Dimension

[Slater '75, Harary, Melter '76]

<u>**Def.**</u> A resolving set is a $S \subseteq V(G)$ such that $\forall u, v \in V$, $\exists z \in S$ with $d(z, u) \neq d(z, v)$.

<u>**Def.**</u> The <u>minimum size</u> of a resolving set of G is the metric dimension of G.

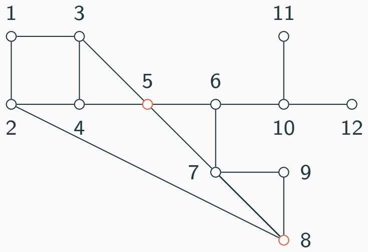


Metric Dimension

[Slater '75, Harary, Melter '76]

Def. A resolving set is a $S \subseteq V(G)$ such that $\forall u, v \in V$, $\exists z \in S$ with $d(z, u) \neq d(z, v)$.

Def. The minimum size of a resolving set of G is the metric dimension of G.

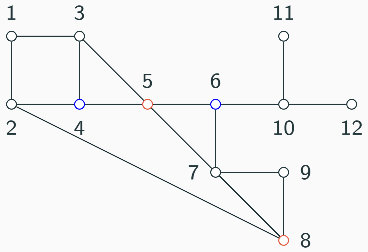


Metric Dimension

[Slater '75, Harary, Melter '76]

Def. A resolving set is a $S \subseteq V(G)$ such that $\forall u, v \in V$, $\exists z \in S$ with $d(z, u) \neq d(z, v)$.

Def. The minimum size of a resolving set of G is the metric dimension of G.

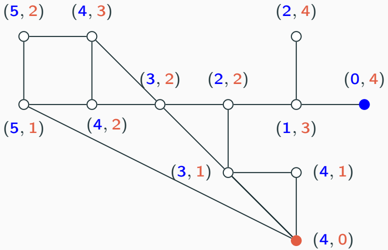


Vertices 4 and 6 are **not** resolved by 5 nor 8.

Metric Dimension [Slater '75, Harary, Melter '76]

Def. A resolving set is a $S \subseteq V(G)$ such that $\forall u, v \in V$, $\exists z \in S$ with $d(z, u) \neq d(z, v)$.

Def. The minimum size of a resolving set of G is the metric dimension of G.



Observation. For any twins $u, v \in V(G)$ and any resolving set S of G, $S \cap \{u, v\} \neq \emptyset$.

Metric Dimension (MDim)

Metric Dimension

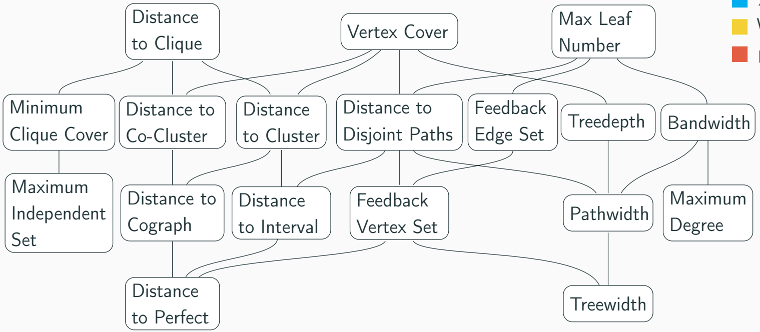
Input: An undirected simple graph G and a positive integer k

Question: Is $\text{md}(G) \leq k$?

Polynomial-time	
Trees	[Slater'75]
Cographs	[Epstein et al'15]
Outerplanar	[Diaz et al'17]

NP-complete	
Arbitrary	[Garey, Johnson'79]
Split	[Epstein et al'15]
Bipartite	[Epstein et al'15]
Co-bipartite	[Epstein et al'15]
Planar	[Diaz et al'17]
Interval	[Foucaud et al'17]

Parameterized complexity of Metric Dimension



- FPT ($f(k) \cdot n^{O(1)}$-time algorithm)
- XP ($n^{f(k)}$-time algorithm)
- W[1]-hard (not FPT unless FPT = W[1])
- para-NP-hard (not XP unless P = NP)

n: size of input

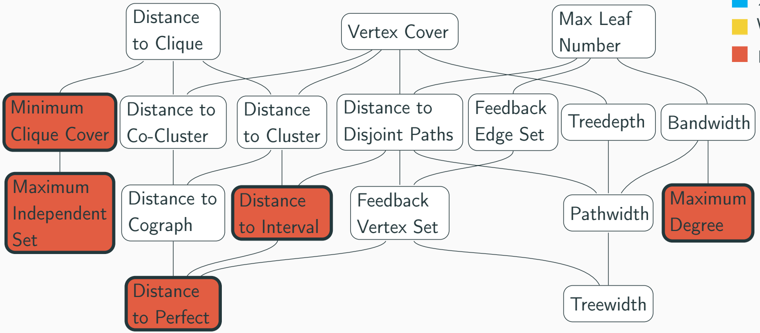
k: size of parameter

A lower parameter is connected to a higher one if it is upper bounded by a function of the higher one

Parameterized complexity of Metric Dimension

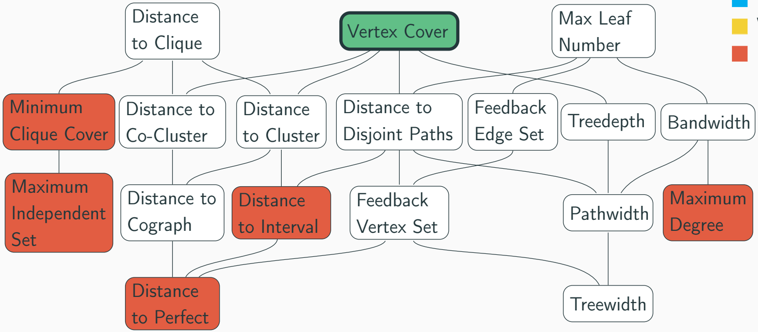
- FPT ($f(k) \cdot n^{O(1)}$-time algorithm)
- XP ($n^{f(k)}$-time algorithm)
- W[1]-hard (not FPT unless FPT = W[1])
- para-NP-hard (not XP unless P = NP)

n: size of input
k: size of parameter



From NP-hardness results on previous slide

Parameterized complexity of Metric Dimension



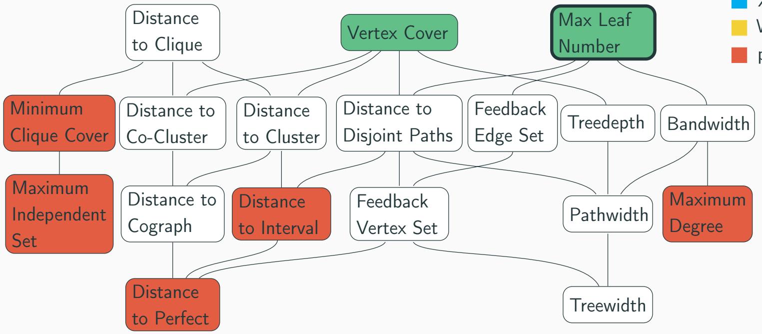
- FPT ($f(k) \cdot n^{O(1)}$-time algorithm)
- XP ($n^{f(k)}$-time algorithm)
- W[1]-hard (not FPT unless FPT = W[1])
- para-NP-hard (not XP unless P = NP)

n: size of input

k: size of parameter

W[2]-hard parameterised by solution size [Hartung, Nichterlein '13]

Parameterized complexity of Metric Dimension



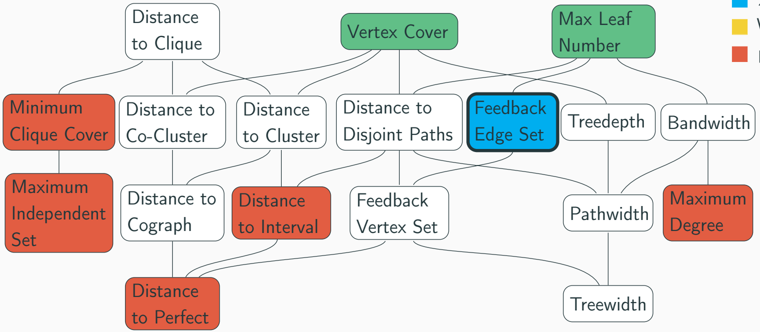
- FPT ($f(k) \cdot n^{O(1)}$-time algorithm)
- XP ($n^{f(k)}$-time algorithm)
- W[1]-hard (not FPT unless FPT = W[1])
- para-NP-hard (not XP unless P = NP)

n: size of input

k: size of parameter

[Eppstein '15]

Parameterized complexity of Metric Dimension



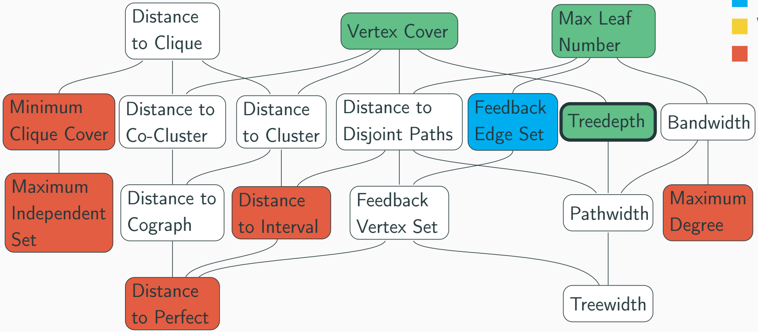
- FPT ($f(k) \cdot n^{O(1)}$-time algorithm)
- XP ($n^{f(k)}$-time algorithm)
- W[1]-hard (not FPT unless FPT = W[1])
- para-NP-hard (not XP unless P = NP)

n: size of input

k: size of parameter

[Epstein et al '15]

Parameterized complexity of Metric Dimension

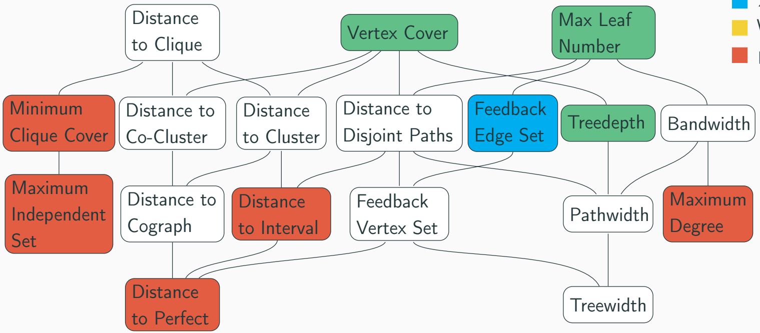


- FPT ($f(k) \cdot n^{O(1)}$-time algorithm)
- XP ($n^{f(k)}$-time algorithm)
- W[1]-hard (not FPT unless FPT = W[1])
- para-NP-hard (not XP unless P = NP)

n: size of input
k: size of parameter

[Gima et al '21]

Parameterized complexity of Metric Dimension

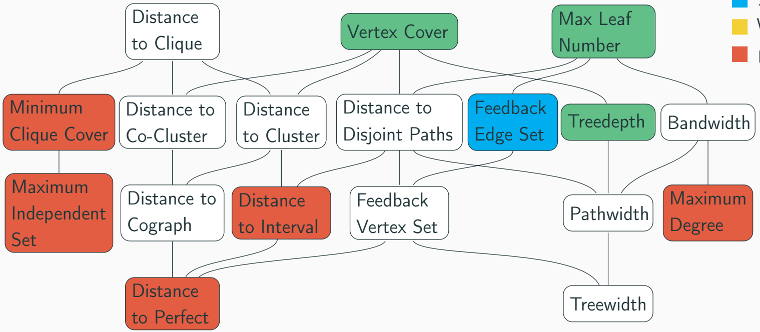


- FPT ($f(k) \cdot n^{O(1)}$-time algorithm)
- XP ($n^{f(k)}$-time algorithm)
- W[1]-hard (not FPT unless FPT = W[1])
- para-NP-hard (not XP unless P = NP)

n: size of input
k: size of parameter

FPT parameterised by treelength + max degree [Belmonte et al '17]
and clique-width + diameter [Gima et al '21]

Parameterized complexity of Metric Dimension



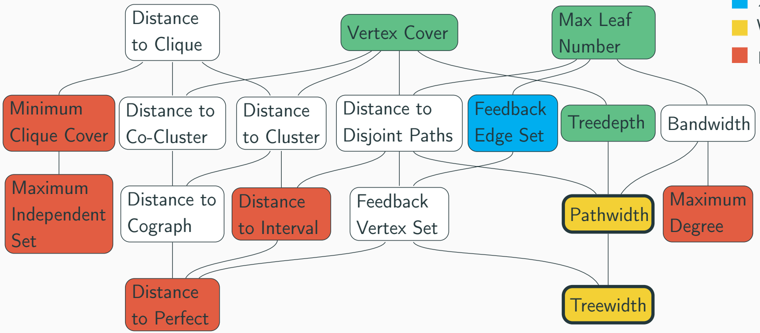
- FPT ($f(k) \cdot n^{O(1)}$-time algorithm)
- XP ($n^{f(k)}$-time algorithm)
- W[1]-hard (not FPT unless FPT = W[1])
- para-NP-hard (not XP unless P = NP)

n: size of input
k: size of parameter

Q1: Complexity parameterised by **Feedback Vertex Set**? [Hartung, Nichterlein '13]

Q2: Complexity parameterised by **treewidth**? [Eppstein '15], [Belmonte et al '17], [Díaz et al '17]

Parameterized complexity of Metric Dimension



- FPT ($f(k) \cdot n^{O(1)}$-time algorithm)
- XP ($n^{f(k)}$-time algorithm)
- W[1]-hard (not FPT unless FPT = W[1])
- para-NP-hard (not XP unless P = NP)

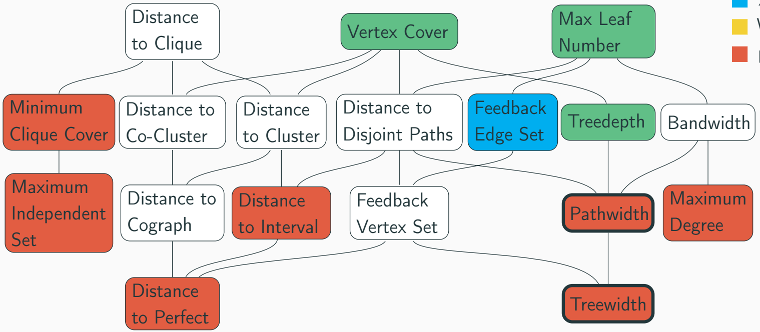
n: size of input
k: size of parameter

Q1: Complexity parameterised by **Feedback Vertex Set**? [Hartung, Nichterlein '13]

Q2: Complexity parameterised by **treewidth**? [Eppstein '15], [Belmonte et al '17], [Díaz et al '17]

Q2 answered first by [Bonnet, Purohit '21].

Parameterized complexity of Metric Dimension



- FPT ($f(k) \cdot n^{O(1)}$-time algorithm)
- XP ($n^{f(k)}$-time algorithm)
- W[1]-hard (not FPT unless FPT = W[1])
- para-NP-hard (not XP unless P = NP)

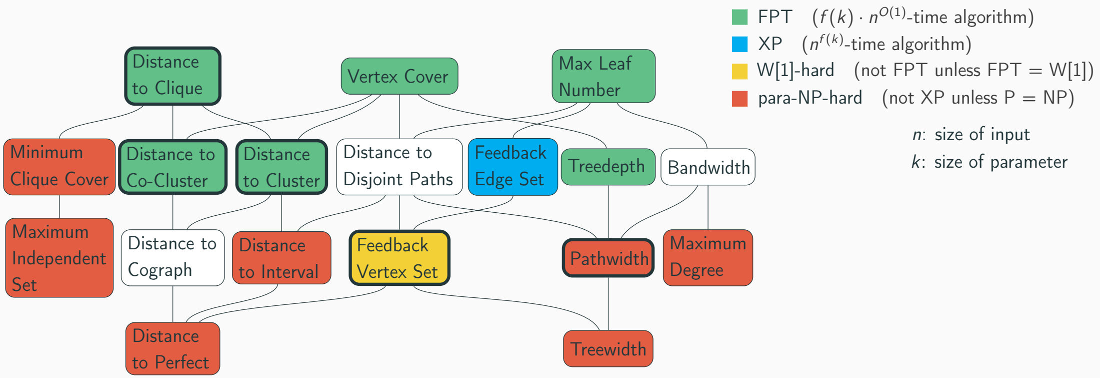
n: size of input
k: size of parameter

Q1: Complexity parameterised by **Feedback Vertex Set**? [Hartung, Nichterlein '13]

Q2: Complexity parameterised by **treewidth**? [Eppstein '15], [Belmonte et al '17], [Díaz et al '17]

Q2 answered first by [Bonnet, Purohit '21]. Then, improved by [Li, Pilipczuk '22]

Parameterized complexity of Metric Dimension



Q1: Complexity parameterised by **Feedback Vertex Set**? [Hartung, Nichterlein '13]

Q2: Complexity parameterised by **treewidth**? [Eppstein '15], [Belmonte et al '17], [Díaz et al '17]

Q1 answered for the combined parameter Feedback Vertex Set + Pathwidth
[Galby, Khazaliya, Mc Inerney, Sharma, Tale '23]

Part 3.

Our Technique and MDim

3-Partitioned 3-SAT

3-Partitioned 3-SAT [Lampis, Melissinos, Vasilakis, 2023]

Input: 3-CNF formula ϕ with a partition of its variables into 3 disjoint sets X^α, X^β, and X^γ such that $|X^\alpha| = |X^\beta| = |X^\gamma| = n$ and each clause contains at most one variable from each of X^α, X^β, and X^γ

Question: Is ϕ satisfiable?

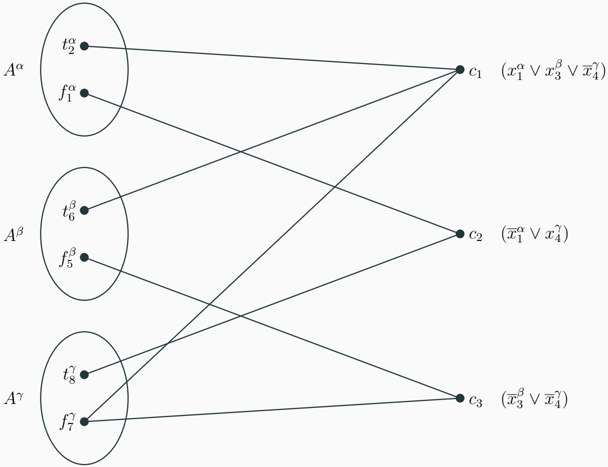
Theorem [Lampis, Melissinos, Vasilakis, 2023]

Unless the ETH fails, 3-Partitioned 3-SAT does not admit an algorithm running in time $2^{o(n)}$.

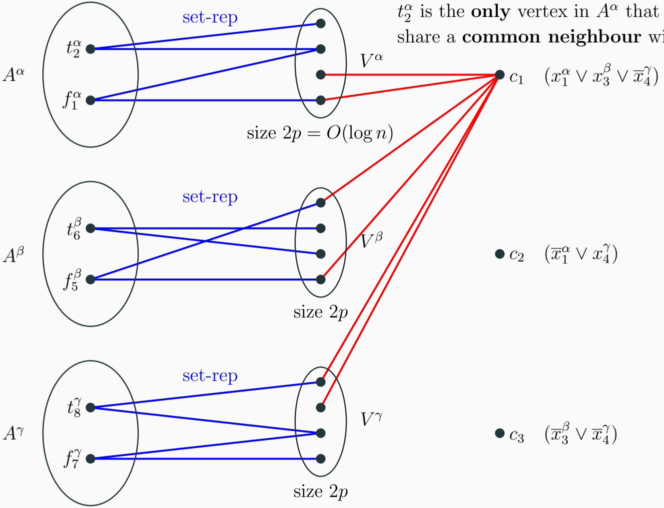
Encode SAT with small separator

$X^\alpha := \{x_1^\alpha, \ldots, x_n^\alpha\}$, t_{2i}^α represents x_i^α, and f_{2i-1}^α represents $\overline{x}_i^\alpha$.

So, for each x_i^α, there is a pair f_{2i-1}^α and t_{2i}^α.



Set-Representation Gadget



t_2^α is the **only** vertex in A^α that **does not** share a **common neighbour** with c_1.

Let F_p be the collection of subsets of $\{1, \ldots, 2p\}$ that contain exactly p integers.

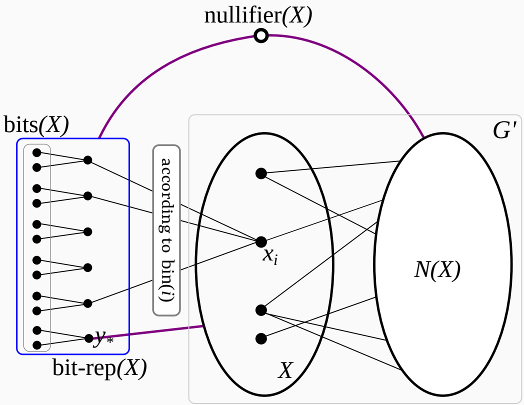
No set in F_p is contained in another set in F_p (Sperner family).

There exists $p = O(\log n)$ s.t. $\binom{2p}{p} \geq 2n$. We define a 1-to-1 function

$$\text{set-rep} : \{1, \ldots, 2n\} \rightarrow F_p.$$

Bit-representation Gadget

Observation. For any twins $u, v \in V(G)$ and any resolving set S of G, $S \cap \{u, v\} \neq \emptyset$.



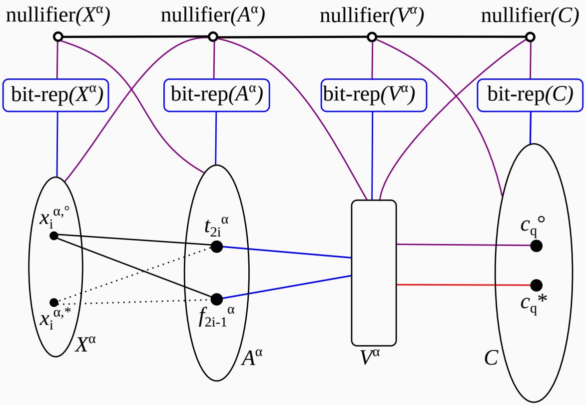
Purple edges represent all possible edges

- For any resolving set S, $|S \cap \text{bits}(X)| \geq \log(|X|) + 1$
- $|S \cap \text{bits}(X)|$ distinguishes each vertex in $X \cup \text{bit-rep}(X)$ from every other vertex in G
- nullifier(X) guarantees that the rest part of $V(G)$ does not affected by the gadget

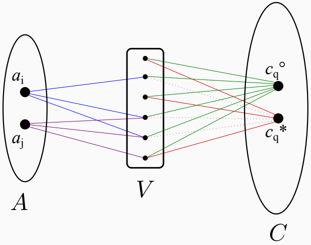
Lower bound for Metric Dimension parameterized by tw

Theorem [FGKLMST, 2023]

Unless the ETH fails, Metric Dimension does not admit an algorithm running in time $2^{f(\text{diam})^{o(\text{tw})}} \cdot n^{O(1)}$, for any computable function f.



Purple — all possible edges
Blue — set-rep
Red — complementary to blue



Budget: $3n$ vertices

(excluding bit-rep gadgets)

Part 4.

Other Results and Applications

Geodetic Set and Strong MDim

GEODETIC SET
Input: An undirected simple graph G
Question: Does there exist $S \subseteq V(G)$ such that $|S| \leq k$ and, for any vertex $u \in V(G)$, there are two vertices $s_1, s_2 \in S$ such that a shortest path from s_1 to s_2 contains u?

Theorem [FGKLMST, 2023]

Unless the ETH fails, GEODETIC SET does not admit algorithms running in time $2^{f(\text{diam})^{o(\text{tw})}} \cdot n^{O(1)}$, for any computable function f.

Strong Metric Dimension

STRONG METRIC DIMENSION
Input: An undirected simple graph G
Question: Does there exist $S \subseteq V(G)$ such that $|S| \leq k$ and, for any pair of vertices $u, v \in V(G)$, there exists a vertex $w \in S$ such that either u lies on some shortest path between v and w, or v lies on some shortest path between u and w?

Theorem [FGKLMST, 2023]

Unless the ETH fails, STRONG METRIC DIMENSION does not admit algorithms running in time $2^{2^{o(\mathrm{vc})}} \cdot n^{O(1)}$, for any computable function f. This also implies the problem does not admit a kernelization algorithm that outputs an instance with $2^{o(vc)}$ vertices

Match with the Algorithms

Theorem **[FGKLMST, 2023]**

Unless the ETH fails, Metric Dimension and Geodetic Set do not admit algorithms running in time $2^{f(\text{diam})^{o(\text{tw})}} \cdot n^{O(1)}$, for any computable function f.

Theorem **[FGKLMST, 2023]**

Unless the ETH fails, Strong Metric Dimension does not admit algorithms running in time $2^{2^{o(\text{vc})}} \cdot n^{O(1)}$, for any computable function f. This also implies the problem does not admit a kernelization algorithm that outputs an instance with $2^{o(vc)}$ vertices

Theorem **[FGKLMST, 2023]**

- Metric Dimension and Geodetic Set admit algorithms running in time $2^{\text{diam}^{\mathcal{O}(\text{tw})}} \cdot n^{\mathcal{O}(1)}$.
- Strong Metric Dimension admits an algorithm running in time $2^{2^{\mathcal{O}(\text{vc})}} \cdot n^{\mathcal{O}(1)}$ and a kernel with $2^{\mathcal{O}(\text{vc})}$ vertices.

Applications of the Technique

Theorem [Chalopin, Chepoi, Mc Inerney, Ratel, COLT 2024]

Unless the ETH fails, POSITIVE NON-CLASHING TEACHING DIMENSION FOR BALLS IN GRAPHS does not admit a $2^{2^{o(\mathrm{vc})}} \cdot n^{O(1)}$-time algorithm, nor a kernelization algorithm outputting a kernel with $2^{o(\mathrm{vc})}$ vertices.

Theorem [Chakraborty, Foucaud, Majumdar, Tale, 2024]

Unless the ETH fails, LOCATING-DOMINATING SET (respectively, TEST COVER) does not admit a $2^{2^{o(\mathrm{tw})}} \cdot n^{O(1)}$-time (respectively, $2^{2^{o(\mathrm{tw})}} \cdot (|U| + |\mathcal{F}|)^{O(1)}$-time) algorithm.

Part 5.

Open Problems

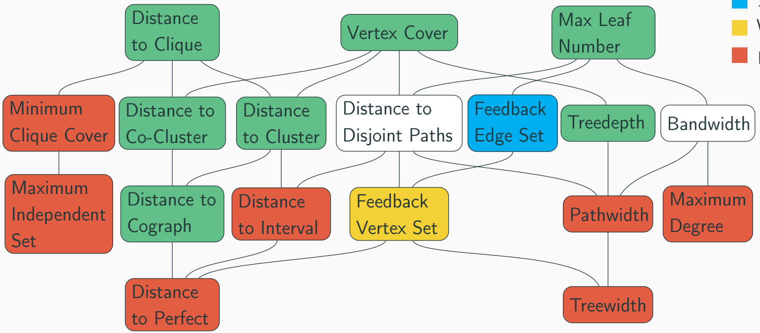
Open Questions

Q1: Are there certain properties shared by distance-based graph problems, that imply such running times? Is there a possible way to generalize our approach to a broader class of problems.

Q2: For which classic problems in NP are the best known FPT algorithms parameterized by tw, $\mathtt{vc}$ (or other parameters) double-exponential?

Q3: For which classic problems do the best known kernelization algorithms output a kernel with $2^{O(\mathtt{vc})}$ vertices?

… and for Metric Dimension



- FPT ($f(k) \cdot n^{O(1)}$-time algorithm)
- XP ($n^{f(k)}$-time algorithm)
- W[1]-hard (not FPT unless FPT = W[1])
- para-NP-hard (not XP unless P = NP)

n: size of input

k: size of parameter

Q4: XP or para-NP-hard parameterised by Feedback Vertex Set?

Q5: W[1]-hard or FPT parameterised by Feedback Edge Set?

Q6: Distance to Disjoint Paths? Bandwidth?

Thank you for your attention!

Further directions

Q1: Are there certain properties shared by distance-based graph problems, that imply such running times? Is there a possible way to generalize our approach to a broader class of problems.

Q2: For which classic problems in NP are the best known FPT algorithms parameterized by tw, $\mathtt{vc}$ (or other parameters) double-exponential?

Q3: For which classic problems do the best known kernelization algorithms output a kernel with $2^{O(\mathtt{vc})}$ vertices?

For Metric Dimension:

Q4: XP or para-NP-hard parameterised by Feedback Vertex Set?

Q5: W[1]-hard or FPT parameterised by Feedback Edge Set?

Q6: Distance to Disjoint Paths? Bandwidth?

Contents