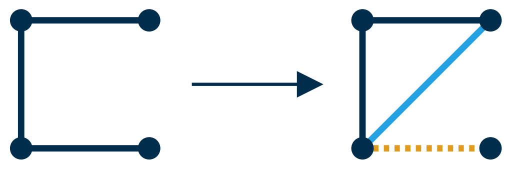
Parameterized Algorithms

Bounded Search Trees

Thomas Bläsius

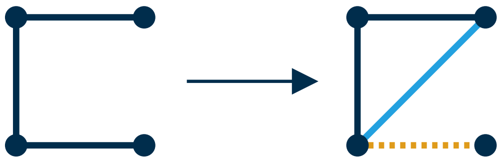
One problem after another

Problem: CLUSTER EDITING
Given a graph G and a parameter k, can G be transformed into a disjoint union of cliques with k edge insertions/deletions?



One problem after another

Problem: CLUSTER EDITING
Given a graph G and a parameter k, can G be transformed into a disjoint union of cliques with k edge insertions/deletions?



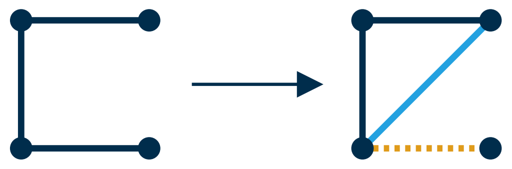
Problem: INDEPENDENT SET
Given a d-degenerate graph G and a parameter $k + d$, does G have an independent set of size k?

d-degenerate: every subgraph has a vertex of degree $\leq d$

independent set: no edges between them

One problem after another

Problem: CLUSTER EDITING
Given a graph G and a parameter k, can G be transformed into a disjoint union of cliques with k edge insertions/deletions?

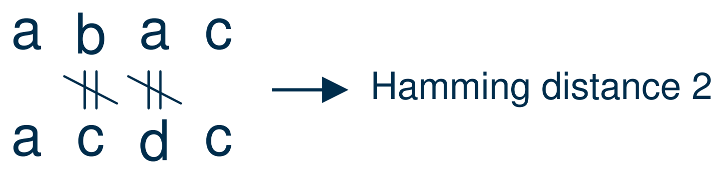


Problem: INDEPENDENT SET
Given a d-degenerate graph G and a parameter $k + d$, does G have an independent set of size k?

d-degenerate: every subgraph has a vertex of degree $\leq d$

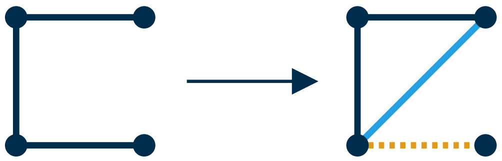
independent set: no edges between them

Problem: CLOSEST STRING
Given n strings and a parameter d, is there a string with Hamming distance $\leq d$ to each of the n strings?



One problem after another

Problem: CLUSTER EDITING
Given a graph G and a parameter k, can G be transformed into a disjoint union of cliques with k edge insertions/deletions?



Problem: INDEPENDENT SET
Given a d-degenerate graph G and a parameter $k + d$, does G have an independent set of size k?

d-degenerate: every subgraph has a vertex of degree $\leq d$

independent set: no edges between them

Problem: CLOSEST STRING
Given n strings and a parameter d, is there a string with Hamming distance $\leq d$ to each of the n strings?

a b a c
$\quad\;\nparallel\;\nparallel$
a c d c → Hamming distance 2

Problem: ODD CYCLE TRANSVERSAL
Given a perfect graph and a parameter k, is there a vertex set of size k that contains a vertex from every odd cycle?

perfect: $\omega = \chi$ for every induced subgraph

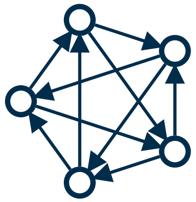
ω: clique number

χ: chromatic number

Even more problems

Problem: FEEDBACK VERTEX SET (on tournaments)
Given a tournament graph and a parameter k, can we delete k vertices such that the graph becomes acyclic?

tournament graph: directed graph with exactly one edge for every vertex pair

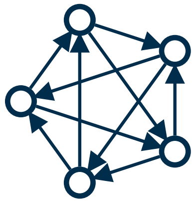


Even more problems

Problem: FEEDBACK VERTEX SET (on tournaments)
Given a tournament graph and a parameter k, can we delete k vertices such that the graph becomes acyclic?

Problem: FEEDBACK ARC SET (on tournaments)
Given a tournament graph and a parameter k, can we delete k edges such that the graph becomes acyclic?

tournament graph: directed graph with exactly one edge for every vertex pair

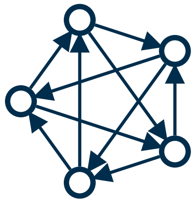


Even more problems

Problem: FEEDBACK VERTEX SET (on tournaments)
Given a tournament graph and a parameter k, can we delete k vertices such that the graph becomes acyclic?

Problem: FEEDBACK ARC SET (on tournaments)
Given a tournament graph and a parameter k, can we delete k edges such that the graph becomes acyclic?

tournament graph: directed graph with exactly one edge for every vertex pair



Your task

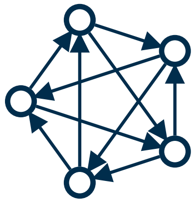
- show that these problems are in FPT using bounded search trees
- quantity over quality: exact running time not relevant

Even more problems

Problem: FEEDBACK VERTEX SET (on tournaments)
Given a tournament graph and a parameter k, can we delete k vertices such that the graph becomes acyclic?

Problem: FEEDBACK ARC SET (on tournaments)
Given a tournament graph and a parameter k, can we delete k edges such that the graph becomes acyclic?

tournament graph: directed graph with exactly one edge for every vertex pair



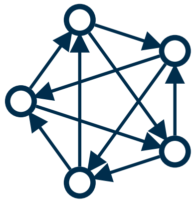
Your task

- show that these problems are in FPT using bounded search trees
- quantity over quality: exact running time not relevant
- but: clean arguments are important

Even more problems

Problem: FEEDBACK VERTEX SET (on tournaments)
Given a tournament graph and a parameter k, can we delete k vertices such that the graph becomes acyclic?

tournament graph: directed graph with exactly one edge for every vertex pair



Problem: FEEDBACK ARC SET (on tournaments)
Given a tournament graph and a parameter k, can we delete k edges such that the graph becomes acyclic?

Your task

- show that these problems are in FPT using bounded search trees
- quantity over quality: exact running time not relevant
- but: clean arguments are important

Clean arguments

- Can you explain it to your peers?
- Is the proof correct and complete?
- Can you simplify the proof?