What's next?
Future directions in Parameterized Complexity

Saket Saurabh

The Institute of Mathematical Sciences, India
and University of Bergen, Norway,
(Inputs from Piotr Faliszewski, Fedor V. Fomin, Daniel Lokshtanov, Dániel Marx,
Shmuel Onn, Marcin Pilipczuk, Michal Pilipczuk, Meirav Zehavi)

Recent Advances in Parameterized Complexity
Tel-Aviv, Israel
December 7, 2017

Classical Directions

- FPT versus W[1]
- Kernelization
- Optimality-Program

Future Directions

Future Directions

- Pareto Optimality

Future Directions

- Pareto Optimality
- FPT in P

Future Directions

- Pareto Optimality
- FPT in P
- FPT Counting

Future Directions

- Pareto Optimality
- FPT in P
- FPT Counting
- FPT Approximation

Future Directions

- Pareto Optimality
- FPT in P
- FPT Counting
- FPT Approximation
- Lossy Kernels

Future Directions

- Pareto Optimality
- FPT in P
- FPT Counting
- FPT Approximation
- Lossy Kernels
- Beyond Graph Algorithms

Future Directions

- Pareto Optimality
- FPT in P
- FPT Counting
- FPT Approximation
- Lossy Kernels
- Beyond Graph Algorithms
  - Computational Social Choice Theory
  - Computational Geometry
  - Mathematical Programming

FPT versus W[1]

Even Set

Input: Set system $\mathcal{S}$ over a universe U, integer k.
Parameter: k
Question: Does there exist a *nonempty* set $X \subseteq U$ of size at most k such that $|X \cap S|$ is even for every $S \in \mathcal{S}$?

Essentially equivalent formulations:

- With graphs and neighborhoods.
- Minimum circuit in a binary matroid.
- Mininum distance in a linear code over a binary alphabet.

Weighted Directed Feedback Vertex Set

Input: A directed graph D, a weight function $w : V(D) \to \mathbb{Z}^+$, and an integer k.
Parameter: k
Find: Among all sets (if exists), $X \subseteq V(D)$, of size at most k such that $D \setminus X$ is a directed acyclic graph, find the one with minimum weight.

Weighted Directed Feedback Vertex Set

Input: A directed graph D, a weight function $w : V(D) \to \mathbb{Z}^+$, and an integer k.
Parameter: k
Find: Among all sets (if exists), $X \subseteq V(D)$, of size at most k such that $D \setminus X$ is a directed acyclic graph, find the one with minimum weight.

- Such Algorithm is known for Undirected Settings – $5^k n^{O(1)}$
- Directed Feedback Vertex Set solvable in time $k^{O(k)} n^{O(1)}$

3-Terminal Directed Multicut

Input: A directed graph D, vertex pairs s_i, t_i, $i \in \{1, 2, 3\}$, and an integer k.
Parameter: k
Question: Does there exist a set $X \subseteq V(D)$ of size at most k such that in $D \setminus X$ there is no directed path from s_i to t_i.

3-Terminal Directed Multicut

Input: A directed graph D, vertex pairs s_i, t_i, $i \in \{1, 2, 3\}$, and an integer k.
Parameter: k
Question: Does there exist a set $X \subseteq V(D)$ of size at most k such that in $D \setminus X$ there is no directed path from s_i to t_i.

- 4-Terminal Directed Multicut is W[1]-hard.

3-Terminal Directed Multicut

Input: A directed graph D, vertex pairs s_i, t_i, $i \in \{1, 2, 3\}$, and an integer k.
Parameter: k
Question: Does there exist a set $X \subseteq V(D)$ of size at most k such that in $D \setminus X$ there is no directed path from s_i to t_i.

- 4-Terminal Directed Multicut is W[1]-hard.
- 3-Terminal Directed Multicut reduces to Directed Odd Cycle Transversal. Later, was recently shown to be W[1]-hard.

3-Terminal Directed Multicut

Input: A directed graph D, vertex pairs s_i, t_i, $i \in \{1, 2, 3\}$, and an integer k.
Parameter: k
Question: Does there exist a set $X \subseteq V(D)$ of size at most k such that in $D \setminus X$ there is no directed path from s_i to t_i.

- 4-Terminal Directed Multicut is W[1]-hard.
- 3-Terminal Directed Multicut reduces to Directed Odd Cycle Transversal. Later, was recently shown to be W[1]-hard.
- Directed Odd Cycle Transversal admits factor 2 FPT approximation and thus 3-Terminal Directed Multicut admits actor 2 FPT approximation.

Graph Isomorphism/rankwidth

Input: Undirected graphs G and H.
Parameter: $rw(G)$
Question: Is G and H isomorphism?

- Admits an algorithm with running time $n^{O(k)}$.

Graph Isomorphism/rankwidth

Input: Undirected graphs G and H.
Parameter: $rw(G)$
Question: Is G and H isomorphism?

- Admits an algorithm with running time $n^{O(k)}$.
- Is it FPT?

Kernelization

Directed Feedback Vertex Set

Input: A directed graph D, and an integer k.
Parameter: k
Find: Does there exist a set, $X \subseteq V(D)$, of size at most k such that $D \setminus X$ is a directed acyclic graph?

Directed Feedback Vertex Set

Input: A directed graph D, and an integer k.
Parameter: k
Find: Does there exist a set, $X \subseteq V(D)$, of size at most k such that $D \setminus X$ is a directed acyclic graph?

- Polynomial kernel is known for undirected graphs – $O(k^2)$
- Directed Feedback Vertex Set solvable in time $k^{O(k)} n^{O(1)}$
- For special cases such as tournaments, or together with some structural parameter the problem is known to admit polynomial kernel.

Directed Feedback Vertex Set

Input: A directed graph D, and an integer k.
Parameter: k
Find: Does there exist a set, $X \subseteq V(D)$, of size at most k such that $D \setminus X$ is a directed acyclic graph?

- Polynomial kernel is known for undirected graphs – $O(k^2)$
- Directed Feedback Vertex Set solvable in time $k^{O(k)}n^{O(1)}$
- For special cases such as tournaments, or together with some structural parameter the problem is known to admit polynomial kernel.
- Even open for planar digraphs.

Planar Vertex Deletion

Input: An undirected graph G, and an integer k.
Parameter: k
Find: Does there exist a set, $X \subseteq V(G)$, of size at most k such that $G \setminus X$ is planar?

Planar Vertex Deletion

Input: An undirected graph G, and an integer k.
Parameter: k
Find: Does there exist a set, $X \subseteq V(G)$, of size at most k such that $G \setminus X$ is planar?

- Planar Vertex Deletion is solvable in time $k^{O(k)}n$
- Polynomial kernel is known for Planar-$\mathcal{F}$-Deletion – $k^{f(\mathcal{F})}$ (deleting k-vertices such that the resulting graph does not contain any graph in $\mathcal{F}$ as a minor). Here, $\mathcal{F}$ contains at least one planar graph.
- Planar Vertex Deletion is $\mathcal{F}$-Deletion for $\mathcal{F} = \{K_{3,3}, K_5\}$.
- The problem is known to admit polynomial kernel when parameterized by vertex-cover number.

Multiway Cut

Input: An undirected graph G, a set of terminals T and an integer k.
Parameter: k
Find: Does there exist a set, $X \subseteq (V(G) \setminus T)$, of size at most k such that in $G \setminus X$ there is no path between any vertices of T?

Multiway Cut

Input: An undirected graph G, a set of terminals T and an integer k.
Parameter: k
Find: Does there exist a set, $X \subseteq (V(G) \setminus T)$, of size at most k such that in $G \setminus X$ there is no path between any vertices of T?

- Multiway Cut is solvable in time $2^k n^{O(1)}$
- Polynomial kernel is known when X is allowed to contain terminals.
- The problem is known to admit polynomial kernel when the number of terminals (t) are fixed $k^{O(t)}$
- So unknown when there are arbitrary number t of terminals.

Deterministic Polynomial kernels

- Odd Cycle Transversal (deleting k vertices such that the resulting graph is bipartite)

Deterministic Polynomial kernels

- Odd Cycle Transversal (deleting k vertices such that the resulting graph is bipartite)
- Vertex Cover Above LP (does there exist k vertices such that the resulting graph is edgeless? Note that this is the same problem as Vertex Cover, but the name Vertex Cover Above LP is used in the context of above guarantee parameterization with an optimum solution to a linear programming relaxation as a lower bound)

Deterministic Polynomial kernels

- Odd Cycle Transversal (deleting k vertices such that the resulting graph is bipartite)
- Vertex Cover Above LP (does there exist k vertices such that the resulting graph is edgeless? Note that this is the same problem as Vertex Cover, but the name Vertex Cover Above LP is used in the context of above guarantee parameterization with an optimum solution to a linear programming relaxation as a lower bound)
- Almost 2-SAT (we are given a 2-CNF formula ϕ and an integer k, and the question is whether one can delete at most k clauses from ϕ to make it satisfiable)

Deterministic Polynomial kernels

- Odd Cycle Transversal
- Vertex Cover Above LP
- Almost 2-SAT

Deterministic Polynomial kernels

- Odd Cycle Transversal
- Vertex Cover Above LP
- Almost 2-SAT

- Randomized polynomial kernel is known

Deterministic Polynomial kernels

- Odd Cycle Transversal
- Vertex Cover Above LP
- Almost 2-SAT

- Randomized polynomial kernel is known
- All this kernels are based on matroid based techniques, specifically, computation of "representative families".

Deterministic Polynomial kernels

- ODD CYCLE TRANSVERSAL
- VERTEX COVER ABOVE LP
- ALMOST 2-SAT

- Randomized polynomial kernel is known
- All this kernels are based on matroid based techniques, specifically, computation of "representative families".
- Source of randomization: known linear representation of gammoids are randomized.

Deterministic Polynomial kernels

- Odd Cycle Transversal
- Vertex Cover Above LP
- Almost 2-SAT

- Randomized polynomial kernel is known
- All this kernels are based on matroid based techniques, specifically, computation of "representative families".
- Source of randomization: known linear representation of gammoids are randomized.
- Deterministic kernel for Odd Cycle Transversal is open for planar graphs, even parameterized by vertex-cover number.

Polynomial kernels: k-Path

What about polynomial Turing kernels?

- k-Path

Polynomial kernels: k-Path

What about polynomial Turing kernels?

- k-Path

define turning kernels

- There is a randomized (deterministic) algorithm running in time $1.66^k n^{O(1)}$ ($2.59^k n^{O(1)}$)

Polynomial kernels: k-PATH

What about polynomial Turing kernels?

- k-PATH

define turning kernels

- There is a randomized (deterministic) algorithm running in time $1.66^k n^{O(1)}$ ($2.59^k n^{O(1)}$)
- Turing kernels are known for classes such as planar graphs, graphs of bounded degree, H-minor free graphs.

Interval Completion

Input: An undirected graph G, and an integer k.
Parameter: k
Question: Does there exist a set $X \subseteq \binom{V(G)}{2}$ of size at most k such that in $G + X$ is an interval graph?

- Interval Completion is solvable in time $O(6^k(n+m))$ and $O(k^{O(\sqrt{k})} n^{57+O(1)})$

Interval Completion

Input: An undirected graph G, and an integer k.
Parameter: k
Question: Does there exist a set $X \subseteq \binom{V(G)}{2}$ of size at most k such that in $G + X$ is an interval graph?

- Interval Completion is solvable in time $O(6^k(n+m))$ and $O(k^{O(\sqrt{k})} n^{57+O(1)})$
- Does there exist polynomial kernel for the problem?

Framework

- Design a framework to rule out polynomial Turing kernels for problems.

Optimality-Program

Optimality-Program

- $n^{O(k)}$ possible but $n^{o(k)}$ is not possible

Optimality-Program

- $n^{O(k)}$ possible but $n^{o(k)}$ is not possible
- $n^{f(k)}$ possible but $n^{o(f(k))}$ is not possible

Optimality-Program

- $n^{O(k)}$ possible but $n^{o(k)}$ is not possible
- $n^{f(k)}$ possible but $n^{o(f(k))}$ is not possible
- $2^{f(k)}n^{O(1)}$ possible but $2^{o(f(k))}n^{O(1)}$ is not possible

Optimality-Program

- $n^{O(k)}$ possible but $n^{o(k)}$ is not possible
- $n^{f(k)}$ possible but $n^{o(f(k))}$ is not possible
- $2^{f(k)}n^{O(1)}$ possible but $2^{o(f(k))}n^{O(1)}$ is not possible
- c^k is possible but $(c-\epsilon)^k$ is not possible

Optimality-Program

- $n^{O(k)}$ possible but $n^{o(k)}$ is not possible
- $n^{f(k)}$ possible but $n^{o(f(k))}$ is not possible
- $2^{f(k)}n^{O(1)}$ possible but $2^{o(f(k))}n^{O(1)}$ is not possible
- c^k is possible but $(c-\epsilon)^k$ is not possible
- Most of these lower bounds are based on conjectures such as FPT not equal to W[1],ETH, SETH, SeCoCo

Directed Feedback Vertex Set

Input: A directed graph D, and an integer k.
Parameter: k
Find: Does there exist a set, $X \subseteq V(D)$, of size at most k such that $D \setminus X$ is a directed acyclic graph?

Directed Feedback Vertex Set

Input: A directed graph D, and an integer k.
Parameter: k
Find: Does there exist a set, $X \subseteq V(D)$, of size at most k such that $D \setminus X$ is a directed acyclic graph?

- Directed Feedback Vertex Set solvable in time $k^{O(k)}n^{O(1)}$

Directed Feedback Vertex Set

Input: A directed graph D, and an integer k.
Parameter: k
Find: Does there exist a set, $X \subseteq V(D)$, of size at most k such that $D \setminus X$ is a directed acyclic graph?

- Directed Feedback Vertex Set solvable in time $k^{O(k)}n^{O(1)}$
- Does the problem admit $c^k n^{O(1)}$ time algorithm? Or, it is not possible to get an algorithm with running time $k^{o(k)}n^{O(1)}$?

Directed Feedback Vertex Set

Input: A directed graph D, and an integer k.
Parameter: k
Find: Does there exist a set, $X \subseteq V(D)$, of size at most k such that $D \setminus X$ is a directed acyclic graph?

- Directed Feedback Vertex Set solvable in time $k^{O(k)} n^{O(1)}$
- Does the problem admit $c^k n^{O(1)}$ time algorithm? Or, it is not possible to get an algorithm with running time $k^{o(k)} n^{O(1)}$?
- Even open for planar digraphs.

Directed Feedback Vertex Set

Input: A directed graph D, and an integer k.
Parameter: k
Find: Does there exist a set, $X \subseteq V(D)$, of size at most k such that $D \setminus X$ is a directed acyclic graph?

- Directed Feedback Vertex Set solvable in time $k^{O(k)} n^{O(1)}$
- Does the problem admit $c^k n^{O(1)}$ time algorithm? Or, it is not possible to get an algorithm with running time $k^{o(k)} n^{O(1)}$?
- Even open for planar digraphs.
- What about simpler task: can we get an FPT algorithm with running time $O(\frac{k}{c})^k n^{O(1)})$, where c is some fixed constant say, 100, or more ambitiously $c = \log k$.

Vertex Disjoint Cycle Packing

Input: An undirected graph G, and an integer k.
Parameter: k
Find: Does there exist a k vertex disjoint cycles?

Vertex Disjoint Cycle Packing

Input: An undirected graph G, and an integer k.
Parameter: k
Find: Does there exist a k vertex disjoint cycles?

- Vertex Disjoint Cycle Packing was known to be solvable in time $2^{O(k \log^2 k)} n^{O(1)}$ [using Erdős+ Pósa, treewidth and dynamic programming]

Vertex Disjoint Cycle Packing

Input: An undirected graph G, and an integer k.
Parameter: k
Find: Does there exist a k vertex disjoint cycles?

- Vertex Disjoint Cycle Packing was known to be solvable in time $2^{O(k \log^2 k)} n^{O(1)}$ [using Erdős+ Pósa, treewidth and dynamic programming]
- Recently, it was improved to $2^{O(\frac{k \log^2 k}{\log \log k})} n^{O(1)}$

Vertex Disjoint Cycle Packing

Input: An undirected graph G, and an integer k.
Parameter: k
Find: Does there exist a k vertex disjoint cycles?

- Vertex Disjoint Cycle Packing was known to be solvable in time $2^{O(k \log^2 k)} n^{O(1)}$ [using Erdős+ Pósa, treewidth and dynamic programming]
- Recently, it was improved to $2^{O(\frac{k \log^2 k}{\log \log k})} n^{O(1)}$
- Does the problem admit $c^k n^{O(1)}$ time or $2^{O(k \log k)} n^{O(1)}$algorithm? Or, it is not possible to get an algorithm with running time $k^{o(k)} n^{O(1)}$? Proving tight upper and lower bound for this problem is very interesting.

Integer Linear Programming (ILP)

Input: Given matrices $A \times \mathbb{Z}^{m \times k}$ and $b \in \mathbb{Z}^{m \times 1}$, and an integer k.
Parameter: k
Find: Does there exists a vector $\bar{x} \in \mathbb{Z}^{k \times 1}$ satisfying the m inequalities, that is, $A\bar{x} \leq b$?

Integer Linear Programming (ILP)

Input: Given matrices $A \times \mathbb{Z}^{m \times k}$ and $b \in \mathbb{Z}^{m \times 1}$, and an integer k.
Parameter: k
Find: Does there exists a vector $\bar{x} \in \mathbb{Z}^{k \times 1}$ satisfying the m inequalities, that is, $A\bar{x} \leq b$?

- Lenstra showed that ILP is FPT with running time doubly exponential in k. Later, Kannan provided an algorithm for ILP running in time $k^{O(k)} n^{O(1)}$

Integer Linear Programming (ILP)

Input: Given matrices $A \times \mathbb{Z}^{m \times k}$ and $b \in \mathbb{Z}^{m \times 1}$, and an integer k.
Parameter: k
Find: Does there exists a vector $\bar{x} \in \mathbb{Z}^{k \times 1}$ satisfying the m inequalities, that is, $A\bar{x} \leq b$?

- Lenstra showed that ILP is FPT with running time doubly exponential in k. Later, Kannan provided an algorithm for ILP running in time $k^{O(k)} n^{O(1)}$
- It is known that the problem can not admit $2^{o(k)} n^{O(1)}$.

Integer Linear Programming (ILP)

Input: Given matrices $A \times \mathbb{Z}^{m \times k}$ and $b \in \mathbb{Z}^{m \times 1}$, and an integer k.
Parameter: k
Find: Does there exists a vector $\bar{x} \in \mathbb{Z}^{k \times 1}$ satisfying the m inequalities, that is, $A\bar{x} \leq b$?

- Lenstra showed that ILP is FPT with running time doubly exponential in k. Later, Kannan provided an algorithm for ILP running in time $k^{O(k)} n^{O(1)}$
- It is known that the problem can not admit $2^{o(k)} n^{O(1)}$.
- Does the problem admit $c^k n^{O(1)}$ time algorithm? Or, it is not possible to get an algorithm with running time $k^{o(k)} n^{O(1)}$?

Integer Linear Programming (ILP)

Input: Given matrices $A \times \mathbb{Z}^{m\times k}$ and $b \in \mathbb{Z}^{m\times 1}$, and an integer k.
Parameter: k
Find: Does there exists a vector $\bar{x} \in \mathbb{Z}^{k\times 1}$ satisfying the m inequalities, that is, $A\bar{x} \leq b$?

- Lenstra showed that ILP is FPT with running time doubly exponential in k. Later, Kannan provided an algorithm for ILP running in time $k^{O(k)}n^{O(1)}$
- It is known that the problem can not admit $2^{o(k)}n^{O(1)}$.
- Does the problem admit $c^k n^{O(1)}$ time algorithm? Or, it is not possible to get an algorithm with running time $k^{o(k)}n^{O(1)}$?
- What about simpler task: can we get an FPT algorithm with running time $O((\frac{k}{c})^k n^{O(1)})$, where c is some fixed constant say, 100, or more ambitiously $c = \log k$.

Multicut

Input: An undirected graph G, vertex pairs s_i, t_i, $i \in \{1, \ldots, p\}$, and an integer k.
Parameter: k
Question: Does there exist a set $X \subseteq V(G)$ of size at most k such that in $G \setminus X$ there is no path from s_i to t_i.

MULTICUT

Input: An undirected graph G, vertex pairs s_i, t_i, $i \in \{1, \ldots, p\}$, and an integer k.
Parameter: k
Question: Does there exist a set $X \subseteq V(G)$ of size at most k such that in $G \setminus X$ there is no path from s_i to t_i.

- Analogously, we can define EDGE MULTICUT.

Multicut

Input: An undirected graph G, vertex pairs s_i, t_i, $i \in \{1, \ldots, p\}$, and an integer k.
Parameter: k
Question: Does there exist a set $X \subseteq V(G)$ of size at most k such that in $G \setminus X$ there is no path from s_i to t_i.

- Analogously, we can define Edge Multicut.
- Multicut is known to be solvable in time $O(2^{O(k^3)} nm)$

Multicut

Input: An undirected graph G, vertex pairs s_i, t_i, $i \in \{1, \ldots, p\}$, and an integer k.
Parameter: k
Question: Does there exist a set $X \subseteq V(G)$ of size at most k such that in $G \setminus X$ there is no path from s_i to t_i.

- Analogously, we can define Edge Multicut.
- Multicut is known to be solvable in time $O(2^{O(k^3)}nm)$
- Does Multicut admits an algorithm with running time $O(2^{o(k^3)}n^{O(1)})$? Or is this optimal?

Multicut

Input: An undirected graph G, vertex pairs s_i, t_i, $i \in \{1, \ldots, p\}$, and an integer k.
Parameter: k
Question: Does there exist a set $X \subseteq V(G)$ of size at most k such that in $G \setminus X$ there is no path from s_i to t_i.

- Analogously, we can define Edge Multicut.
- Multicut is known to be solvable in time $O(2^{O(k^3)}nm)$
- Does Multicut admits an algorithm with running time $O(2^{o(k^3)}n^{O(1)})$? Or is this optimal?
- What about on planar graphs?

???

Input: An undirected graph G, and an integer k.
Parameter: k
Question: Does G has an independent set of size k or a clique of size k?

???

Input: An undirected graph G, and an integer k.
Parameter: k
Question: Does G has an independent set of size k or a clique of size k?

- Is it FPT?

Ramsey

Input: An undirected graph G, and an integer k.
Parameter: k
Question: Does G has an independent set of size$k$ or a clique of size k?

- Is it FPT?

Ramsey

Input: An undirected graph G, and an integer k.
Parameter: k
Question: Does G has an independent set of size$k$ or a clique of size k?

- Is it FPT?
- If $k \leq \log n$ then the answer is yes.

Ramsey

Input: An undirected graph G, and an integer k.
Parameter: k
Question: Does G has an independent set of size$k$ or a clique of size k?

- Is it FPT?
- If $k \leq \log n$ then the answer is yes. Else, $n \leq 2^k$ and thus we can try all possible subset of size k and check whether this is clique or an independent set.

RAMSEY

Input: An undirected graph G, and an integer k.
Parameter: k
Question: Does G has an independent set of size$k$ or a clique of size k?

- Is it FPT?
- If $k \leq \log n$ then the answer is yes. Else, $n \leq 2^k$ and thus we can try all possible subset of size k and check whether this is clique or an independent set. $\binom{2^k}{k}$ — $2^{O(k^2)}n^{O(1)}$.

RAMSEY

Input: An undirected graph G, and an integer k.
Parameter: k
Question: Does G has an independent set of size$k$ or a clique of size k?

- Is it FPT?
- If $k \leq \log n$ then the answer is yes. Else, $n \leq 2^k$ and thus we can try all possible subset of size k and check whether this is clique or an independent set. $\binom{2^k}{k}$ — $2^{O(k^2)}n^{O(1)}$.
- Does RAMSEY admits an algorithm with running time $O(2^{o(k^2)}n^{O(1)})$? Or is this optimal?

RAMSEY

Input: An undirected graph G, and an integer k.
Parameter: k
Question: Does G has an independent set of size$k$ or a clique of size k?

- Is it FPT?
- If $k \leq \log n$ then the answer is yes. Else, $n \leq 2^k$ and thus we can try all possible subset of size k and check whether this is clique or an independent set. $\binom{2^k}{k}$ — $2^{O(k^2)} n^{O(1)}$.
- Does RAMSEY admits an algorithm with running time $O(2^{o(k^2)} n^{O(1)})$? Or is this optimal? My conjecture that this is optimal!

Disjoint Paths/Minor Testing

- The best known parameter dependence for the k-disjoint paths problem and H-minor testing seems to be triple exponential. [Kawarabayashi and Wollan 2010] using [Chekuri and Chuzhoy 2014].
- For planar graphs, $2^{2^{\text{poly}(k)}} n^{O(1)}$ algorithm. [Adler et al. 2011]
- Are there $2^{\text{poly}(k)} n^{O(1)}$ time algorithms for planar or general graphs?
- Or these algorithms are optimal?

Rectangle Stabbing

Input: A set R of n axis-parallel non-overlapping rectangles embedded in a plane, a set L of vertical and horizontal lines embedded in the plane, and an integer k.
Parameter: k
Question: Does there exist a set $L' \subseteq L$ with $|L'| \leq k$ such that every rectangle from R is stabbed by at least one line from L'?

- Known to admit an algorithm with running time $2^{O(k^2 \log k)} n^{O(1)}$.

Rectangle Stabbing

Input: A set R of n axis-parallel non-overlapping rectangles embedded in a plane, a set L of vertical and horizontal lines embedded in the plane, and an integer k.
Parameter: k
Question: Does there exist a set $L' \subseteq L$ with $|L'| \leq k$ such that every rectangle from R is stabbed by at least one line from L'?

- Known to admit an algorithm with running time $2^{O(k^2 \log k)} n^{O(1)}$.

Rectangle Stabbing

Input: A set R of n axis-parallel non-overlapping rectangles embedded in a plane, a set L of vertical and horizontal lines embedded in the plane, and an integer k.
Parameter: k
Question: Does there exist a set $L' \subseteq L$ with $|L'| \leq k$ such that every rectangle from R is stabbed by at least one line from L'?

- Known to admit an algorithm with running time $2^{O(k^2 \log k)} n^{O(1)}$.
- Does Rectangle Stabbing admits an algorithm with running time $O(2^{O(k^2)} n^{O(1)})$? Or is this optimal?

Rectangle Stabbing

Input: A set R of n axis-parallel non-overlapping rectangles embedded in a plane, a set L of vertical and horizontal lines embedded in the plane, and an integer k.
Parameter: k
Question: Does there exist a set $L' \subseteq L$ with $|L'| \leq k$ such that every rectangle from R is stabbed by at least one line from L'?

- Known to admit an algorithm with running time $2^{O(k^2 \log k)} n^{O(1)}$.
- Does Rectangle Stabbing admits an algorithm with running time $O(2^{O(k^2)} n^{O(1)})$? Or is this optimal?
- Does the problem admits a polynomial kernel?

Biclique

Input: An undirected graph G, and an integer k.
Parameter: k
Question: Does there exist two vertex disjoint sets, say X and Y on at least k vertices such that there is an edge between every pair of vertices (biclique)?

- Admits an algorithm with running time n^{2k+2}.

BICLIQUE

Input: An undirected graph G, and an integer k.
Parameter: k
Question: Does there exist two vertex disjoint sets, say X and Y on at least k vertices such that there is an edge between every pair of vertices (biclique)?

- Admits an algorithm with running time n^{2k+2}.
- Under randomized ETH, known not to admit $f(k)n^{o(\sqrt{k})}$ algorithm.

Biclique

Input: An undirected graph G, and an integer k.
Parameter: k
Question: Does there exist two vertex disjoint sets, say X and Y on at least k vertices such that there is an edge between every pair of vertices (biclique)?

- Admits an algorithm with running time n^{2k+2}.
- Under randomized ETH, known not to admit $f(k)n^{o(\sqrt{k})}$ algorithm.
- Show that there is no $n^{o(k)}$ algorithm unless something fails.

Clique

Input: An undirected graph G, and an integer k.
Parameter: k
Question: Does there exist clique on at least k vertices?

- Admits an algorithm with running time $n^{\frac{\omega k}{3}}$. Here, ω is the exponent of matrix multiplication.

CLIQUE

Input: An undirected graph G, and an integer k.
Parameter: k
Question: Does there exist clique on at least k vertices?

- Admits an algorithm with running time $n^{\frac{\omega k}{3}}$. Here, ω is the exponent of matrix multiplication.
- Under ETH, known not to admit $f(k)n^{o(k)}$ algorithm.

Clique

Input: An undirected graph G, and an integer k.
Parameter: k
Question: Does there exist clique on at least k vertices?

- Admits an algorithm with running time $n^{\frac{\omega k}{3}}$. Here, ω is the exponent of matrix multiplication.
- Under ETH, known not to admit $f(k)n^{o(k)}$ algorithm.
- Assuming SETH or something else show that there is no $n^{\alpha k}$ algorithm for some fixed constant α.

Partitioned Subgraph Isomorphism

Input: Undirected graphs G and$H$ and a coloring function $f : V(H) \to [|V(G)|]$.
Parameter: $|E(G)|$
Question: Does there exist subgraph, X in H that is isomorphic to G and X contains exactly one vertex from each color class?

- Admits an algorithm with running time $n^{O(k)}$.

Partitioned Subgraph Isomorphism

Input: Undirected graphs G and$H$ and a coloring function $f : V(H) \to [|V(G)|]$.
Parameter: $|E(G)|$
Question: Does there exist subgraph, X in H that is isomorphic to G and X contains exactly one vertex from each color class?

- Admits an algorithm with running time $n^{O(k)}$.
- Under ETH, known not to admit $f(k)n^{o(k/\log k)}$ algorithm.

Partitioned Subgraph Isomorphism

Input: Undirected graphs G and$H$ and a coloring function $f : V(H) \to [|V(G)|]$.
Parameter: $|E(G)|$
Question: Does there exist subgraph, X in H that is isomorphic to G and X contains exactly one vertex from each color class?

- Admits an algorithm with running time $n^{O(k)}$.
- Under ETH, known not to admit $f(k)n^{o(k/\log k)}$ algorithm.
- Close the gap between upper and lower bound.

DIRECTED k PATH

Input: A directed graphs G and a non-negative integer k.
Parameter: k
Question: Does there exist a directed path on k vertices in G?

- Admits an algorithm with running time $2^k n^{O(1)}$.

Directed k Path

Input: A directed graphs G and a non-negative integer k.
Parameter: k
Question: Does there exist a directed path on k vertices in G?

- Admits an algorithm with running time $2^k n^{O(1)}$.
- Could we show that it does not admit an algorithm with running time $(2-\epsilon)^k n^{O(1)}$.

Pareto-Optimality

Multicut

Input: An undirected graph G, vertex pairs s_i, t_i, $i \in \{1, \ldots, p\}$, and an integer k.
Parameter: k
Question: Does there exist a set $X \subseteq V(G)$ of size at most k such that in $G \setminus X$ there is no path from s_i to t_i.

- Analogously, we can define Edge Multicut.
- Multicut is known to be solvable in time $O(2^{O(k^3)}nm)$

Multicut

Input: An undirected graph G, vertex pairs s_i, t_i, $i \in \{1, \ldots, p\}$, and an integer k.
Parameter: k
Question: Does there exist a set $X \subseteq V(G)$ of size at most k such that in $G \setminus X$ there is no path from s_i to t_i.

- Analogously, we can define Edge Multicut.
- Multicut is known to be solvable in time $O(2^{O(k^3)}nm)$
- Does Multicut admits an algorithm with running time $O(2^{o(k^3)}(n+m))$? Or is this optimal? What about any $f(k)(n+m)$?

Interval Completion

Input: An undirected graph G, and an integer k.
Parameter: k
Question: Does there exist a set $X \subseteq \binom{V(G)}{2}$ of size at most k such that in $G + X$ is an interval graph?

- Interval Completion is solvable in time $O(6^k(n+m))$ and $O(k^{O(\sqrt{k})}n^{57+O(1)})$

Interval Completion

Input: An undirected graph G, and an integer k.
Parameter: k
Question: Does there exist a set $X \subseteq \binom{V(G)}{2}$ of size at most k such that in $G + X$ is an interval graph?

- Interval Completion is solvable in time $O(6^k(n+m))$ and $O(k^{O(\sqrt{k})} n^{57+O(1)})$
- Does there exist $2^{o(k)}(n+m)$ algorithm for the problem?

Interval Completion

Input: An undirected graph G, and an integer k.
Parameter: k
Question: Does there exist a set $X \subseteq \binom{V(G)}{2}$ of size at most k such that in $G + X$ is an interval graph?

- Interval Completion is solvable in time $O(6^k(n+m))$ and $O(k^{O(\sqrt{k})} n^{57+O(1)})$
- Does there exist $2^{o(k)}(n+m)$ algorithm for the problem?
- Similar question can be asked for Chordal Completion?

Graph Isomorphism/treewidth

Input: Undirected graphs G and H.
Parameter: $tw(G)$
Question: Is G and H isomorphism?

- Admits an algorithm with running time $O(2^{O(k^5 \log k)} n^5)$.

Graph Isomorphism/treewidth

Input: Undirected graphs G and H.
Parameter: $tw(G)$
Question: Is G and H isomorphism?

- Admits an algorithm with running time $O(2^{O(k^5 \log k)} n^5)$.
- A challenging question would be to improve it to $O(2^{O(k \log k)} n^{O(1)})$ or even $O(2^{O(k)} n^{O(1)})$

Graph Isomorphism/treewidth

Input: Undirected graphs G and H.
Parameter: $tw(G)$
Question: Is G and H isomorphism?

- Admits an algorithm with running time $O(2^{O(k^5 \log k)} n^5)$.
- A challenging question would be to improve it to $O(2^{O(k \log k)} n^{O(1)})$ or even $O(2^{O(k)} n^{O(1)})$
- Is it possible $f(k)n^2$ or $f(k)n$.

- MINIMUM BISECTION
- Many of the cut problems such as DIRECTED MULTIWAY CUT, k-WAY CUT

FPT in P

Diameter/Treewidth

- Diameter of a graph of treewidth t can be computed in time $2^{O(t \log t)} n^{1+o(1)}$

- Diameter of a graph of treewidth t can be computed in time $2^{O(t \log t)} n^{1+o(1)}$ but can not be computed in time $2^{o(t)} n^{2-\epsilon}$.

Diameter/Treewidth

- Diameter of a graph of treewidth t can be computed in time $2^{O(t \log t)} n^{1+o(1)}$ but can not be computed in time $2^{o(t)} n^{2-\epsilon}$. Could be close this gap?

Diameter/Treewidth

- Diameter of a graph of treewidth t can be computed in time $2^{O(t \log t)} n^{1+o(1)}$ but can not be computed in time $2^{o(t)} n^{2-\epsilon}$. Could be close this gap?
- What about parameterizing Diameter by feedback vertex set number, that is, Diameter/FVS?

Diameter/Treewidth

- Diameter of a graph of treewidth t can be computed in time $2^{O(t \log t)} n^{1+o(1)}$ but can not be computed in time $2^{o(t)} n^{2-\epsilon}$. Could be close this gap?
- What about parameterizing Diameter by feedback vertex set number, that is, Diameter/FVS? The same reduction shows that diameter on graphs of vertex cover number k can not be computed in time $2^{o(k)} n^{2-\epsilon}$. It is possible to get $2^{O(k)} n^{1+o(1)}$ for diameter parameterized by vertex cover.

FPT Counting

Square root phenomenon

Are there $2^{O(\sqrt{k}\cdot \text{polylog}(k))} n^{O(1)}$ time FPT algorithms for following counting problems on planar graphs?

- k-path
- k-mathching
- k disjoint triangles
- k independent set

Square root phenomenon

Are there $2^{O(\sqrt{k} \cdot \text{polylog}(k))} n^{O(1)}$ time FPT algorithms for following counting problems on planar graphs?

- k-path
- k-mathching
- k disjoint triangles
- k independent set

See more in Daniel's talk!

FPT Approximation

FPT approximation

Maximum Clique

Given G and integer k, in time $f(k)n^{O(1)}$ either

- find a $g(k)$-clique (for some unbounded nondecreasing function g) or
- correctly state that there is no k-clique.

FPT approximation

MAXIMUM CLIQUE

Given G and integer k, in time $f(k)n^{O(1)}$ either

- find a $g(k)$-clique (for some unbounded nondecreasing function g) or
- correctly state that there is no k-clique.

- Known not to admits any $f(k)$ FPT approximation under gap-ETH.

FPT approximation

Maximum Clique

Given G and integer k, in time $f(k)n^{O(1)}$ either

- find a $g(k)$-clique (for some unbounded nondecreasing function g) or
- correctly state that there is no k-clique.

- Known not to admits any $f(k)$ FPT approximation under gap-ETH.
- Could we prove this under conjecture such as FPT$\neq$W[1]?

FPT approximation

MAXIMUM CLIQUE

Given G and integer k, in time $f(k)n^{O(1)}$ either

- find a $g(k)$-clique (for some unbounded nondecreasing function g) or
- correctly state that there is no k-clique.

- Known not to admits any $f(k)$ FPT approximation under gap-ETH.
- Could we prove this under conjecture such as FPT$\neq$W[1]?
- There is a recent paper that does for MINIMUM DOMINATING SET.

Some other problems

- FPT approximation for Bandwidth on general graphs.

Some other problems

- FPT approximation for BANDWIDTH on general graphs. Known to be W-hard even for trees and admits an FPT-approximaiton on trees.

Some other problems

- FPT approximation for BANDWIDTH on general graphs. Known to be W-hard even for trees and admits an FPT-approximaiton on trees.
- Obtain a constant factor approximation for TREEDEPTH in time $c^k n^{O(1)}$?

Lossy Kernels

Lossy Kernels

- Does CONNECTED VERTEX COVER, DISJOINT FACTORS or Disjoint Cycle Packing admit an EPSAKS?

Lossy Kernels

- Does Connected Vertex Cover, Disjoint Factors or Disjoint Cycle Packing admit an EPSAKS?
- Does Edge Clique Cover admit a constant factor approximate kernel of polynomial size?

Lossy Kernels

- Does Connected Vertex Cover, Disjoint Factors or Disjoint Cycle Packing admit an EPSAKS?
- Does Edge Clique Cover admit a constant factor approximate kernel of polynomial size?
- Does Directed Feedback Vertex Set admit a constant factor approximate kernel of polynomial size?

Lossy Kernels

- Does Connected Vertex Cover, Disjoint Factors or Disjoint Cycle Packing admit an EPSAKS?
- Does Edge Clique Cover admit a constant factor approximate kernel of polynomial size?
- Does Directed Feedback Vertex Set admit a constant factor approximate kernel of polynomial size?
- Does Multiway Cut or Subset Feedback Vertex Set have a PSAKS?

Lossy Kernels

- Does Connected Vertex Cover, Disjoint Factors or Disjoint Cycle Packing admit an EPSAKS?
- Does Edge Clique Cover admit a constant factor approximate kernel of polynomial size?
- Does Directed Feedback Vertex Set admit a constant factor approximate kernel of polynomial size?
- Does Multiway Cut or Subset Feedback Vertex Set have a PSAKS?
- Does Disjoint Hole Packing admit a PSAKS? Here a *hole* in a graph G is an induced cycle of length 4 or more.

Lossy Kernels

- Does Connected Vertex Cover, Disjoint Factors or Disjoint Cycle Packing admit an EPSAKS?
- Does Edge Clique Cover admit a constant factor approximate kernel of polynomial size?
- Does Directed Feedback Vertex Set admit a constant factor approximate kernel of polynomial size?
- Does Multiway Cut or Subset Feedback Vertex Set have a PSAKS?
- Does Disjoint Hole Packing admit a PSAKS? Here a *hole* in a graph G is an induced cycle of length 4 or more.
- Does Optimal Linear Arrangement parameterized by vertex cover admit a constant factor approximate kernel of polynomial size, or even a PSAKS?

Lossy Kernels

- Does Connected Vertex Cover, Disjoint Factors or Disjoint Cycle Packing admit an EPSAKS?
- Does Edge Clique Cover admit a constant factor approximate kernel of polynomial size?
- Does Directed Feedback Vertex Set admit a constant factor approximate kernel of polynomial size?
- Does Multiway Cut or Subset Feedback Vertex Set have a PSAKS?
- Does Disjoint Hole Packing admit a PSAKS? Here a *hole* in a graph G is an induced cycle of length 4 or more.
- Does Optimal Linear Arrangement parameterized by vertex cover admit a constant factor approximate kernel of polynomial size, or even a PSAKS?
- Does Treewidth admit an constant factor approximate kernel of polynomial size? Here even a Turing kernel (with a constant factor approximation) would be very interesting.

Lossy Kernels

- Does Connected Vertex Cover, Disjoint Factors or Disjoint Cycle Packing admit an EPSAKS?
- Does Edge Clique Cover admit a constant factor approximate kernel of polynomial size?
- Does Directed Feedback Vertex Set admit a constant factor approximate kernel of polynomial size?
- Does Multiway Cut or Subset Feedback Vertex Set have a PSAKS?
- Does Disjoint Hole Packing admit a PSAKS? Here a *hole* in a graph G is an induced cycle of length 4 or more.
- Does Optimal Linear Arrangement parameterized by vertex cover admit a constant factor approximate kernel of polynomial size, or even a PSAKS?
- Does Treewidth admit an constant factor approximate kernel of polynomial size? Here even a Turing kernel (with a constant factor approximation) would be very interesting.
- See more in the paper!

FPT and Computational Social Choice Theory

Look at the slides of Piotr!

FPT and Computational Geometry

- Constant Factor Approximation for Art Gallery problem (given a polygon can you guard this by at most k guards with this respect to different visibility definition) in FPT time or show that no such approximation algorithm exists. It is known to be W[1]-hard.

- Constant Factor Approximation for Art Gallery problem (given a polygon can you guard this by at most k guards with this respect to different visibility definition) in FPT time or show that no such approximation algorithm exists. It is known to be W[1]-hard.
- Several of these problems have combinatorial upper bounds. For example, one can guard a polygon with n vertices by $\frac{n}{3}$ guards.

- Constant Factor Approximation for Art Gallery problem (given a polygon can you guard this by at most k guards with this respect to different visibility definition) in FPT time or show that no such approximation algorithm exists. It is known to be W[1]-hard.
- Several of these problems have combinatorial upper bounds. For example, one can guard a polygon with n vertices by $\frac{n}{3}$ guards. Is the problem of guarding by at most $\frac{n}{3} - k$ guards FPT by k.

- Constant Factor Approximation for Art Gallery problem (given a polygon can you guard this by at most k guards with this respect to different visibility definition) in FPT time or show that no such approximation algorithm exists. It is known to be W[1]-hard.
- Several of these problems have combinatorial upper bounds. For example, one can guard a polygon with n vertices by $\frac{n}{3}$ guards. Is the problem of guarding by at most $\frac{n}{3} - k$ guards FPT by k.
- There are several such upper bounds in the computational geometry literature. Studying below the lower bounds for Art Gallery Problems is an interesting research avenue.

FPT and Mathematical Programming

Some Directions

More applications of n-fold integer programming and multiway tables to parameterized complexity and approximation algorithms

Further development of n-fold integer programming theory

Complexity of n-fold IP with r,s,t parameters but A unary/binary input?

Faster algorithm for tables and in particular $3 \times 3 \times n$ tables ?

Is the Graver complexity of $3 \times m \times n$ tables $g(m) = 3^{m-1}$?
What about $g(m_1, \ldots, m_k)$ for $m_1 \times \cdots \times m_k \times n$ tables ?

Is 4-block n-fold IP fixed-parameter tractable ?

Complexity of Graver bases detection and output sensitive computation?

Integer Linear Programming (ILP)

Input: Given matrices $A \times \mathbb{Z}^{m \times k}$ and $b \in \mathbb{Z}^{m \times 1}$, and an integer k.
Parameter: k
Find: Does there exists a vector $\bar{x} \in \mathbb{Z}^{k \times 1}$ satisfying the m inequalities, that is, $A\bar{x} \leq b$?

Integer Linear Programming (ILP)

Input: Given matrices $A \times \mathbb{Z}^{m \times k}$ and $b \in \mathbb{Z}^{m \times 1}$, and an integer k.
Parameter: k
Find: Does there exists a vector $\bar{x} \in \mathbb{Z}^{k \times 1}$ satisfying the m inequalities, that is, $A\bar{x} \leq b$?

- Lenstra showed that ILP is FPT with running time doubly exponential in k. Later, Kannan provided an algorithm for ILP running in time $k^{O(k)} n^{O(1)}$

Integer Linear Programming (ILP)

Input: Given matrices $A \times \mathbb{Z}^{m \times k}$ and $b \in \mathbb{Z}^{m \times 1}$, and an integer k.
Parameter: k
Find: Does there exists a vector $\bar{x} \in \mathbb{Z}^{k \times 1}$ satisfying the m inequalities, that is, $A\bar{x} \leq b$?

- Lenstra showed that ILP is FPT with running time doubly exponential in k. Later, Kannan provided an algorithm for ILP running in time $k^{O(k)} n^{O(1)}$
- It is known that the problem can not admit $2^{o(k)} n^{O(1)}$.

Integer Linear Programming (ILP)

Input: Given matrices $A \times \mathbb{Z}^{m \times k}$ and $b \in \mathbb{Z}^{m \times 1}$, and an integer k.
Parameter: k
Find: Does there exists a vector $\bar{x} \in \mathbb{Z}^{k \times 1}$ satisfying the m inequalities, that is, $A\bar{x} \leq b$?

- Lenstra showed that ILP is FPT with running time doubly exponential in k. Later, Kannan provided an algorithm for ILP running in time $k^{O(k)} n^{O(1)}$
- It is known that the problem can not admit $2^{o(k)} n^{O(1)}$.
- Does the problem admit $c^k n^{O(1)}$ time algorithm? Or, it is not possible to get an algorithm with running time $k^{o(k)} n^{O(1)}$?

Integer Linear Programming (ILP)

Input: Given matrices $A \times \mathbb{Z}^{m \times k}$ and $b \in \mathbb{Z}^{m \times 1}$, and an integer k.
Parameter: k
Find: Does there exists a vector $\bar{x} \in \mathbb{Z}^{k \times 1}$ satisfying the m inequalities, that is, $A\bar{x} \leq b$?

- Lenstra showed that ILP is FPT with running time doubly exponential in k. Later, Kannan provided an algorithm for ILP running in time $k^{O(k)} n^{O(1)}$
- It is known that the problem can not admit $2^{o(k)} n^{O(1)}$.
- Does the problem admit $c^k n^{O(1)}$ time algorithm? Or, it is not possible to get an algorithm with running time $k^{o(k)} n^{O(1)}$?
- What about simpler task: can we get an FPT algorithm with running time $O(\frac{k}{c})^k n^{O(1)})$, where c is some fixed constant say, 100, or more ambitiously $c = \log k$.

Integer Quadratic Programming (IQP)

Input: An $n \times n$ integer matrix Q, an $m \times n$ integer matrix A and an m-dimensional integer vector b.
Parameter: n or $n + m$
Find: A vector $x \in \mathbb{Z}^n$ minimizing $x^T Qx$, subject to $Ax \leq b$.

Integer Quadratic Programming (IQP)

Input: An $n \times n$ integer matrix Q, an $m \times n$ integer matrix A and an m-dimensional integer vector b.
Parameter: n or $n + m$
Find: A vector $x \in \mathbb{Z}^n$ minimizing $x^T Q x$, subject to $Ax \leq b$.

- Is it FPT?

Integer Quadratic Programming (IQP)

Input: An $n \times n$ integer matrix Q, an $m \times n$ integer matrix A and an m-dimensional integer vector b.
Parameter: n or $n + m$
Find: A vector $x \in \mathbb{Z}^n$ minimizing $x^T Q x$, subject to $Ax \leq b$.

- Is it FPT?
- Known to be FPT parameterized by $n + \alpha$. Here α is the largest absolute value of an entry of Q and A.

Integer Quadratic Programming (IQP)

Input: An $n \times n$ integer matrix Q, an $m \times n$ integer matrix A and an m-dimensional integer vector b.
Parameter: n or $n + m$
Find: A vector $x \in \mathbb{Z}^n$ minimizing $x^T Qx$, subject to $Ax \leq b$.

- Is it FPT?
- Known to be FPT parameterized by $n + \alpha$. Here α is the largest absolute value of an entry of Q and A.
- Investigating the parameterized complexity of special cases of (integer) mathematical programming with degree-bounded polynomials in the objective function and linear constraints looks like an exciting research direction.

Thank You!