Report from Dagstuhl Seminar 17041

Randomization in Parameterized Complexity

Edited by

Marek Cygan[1], Fedor V. Fomin[2], Danny Hermelin[3], and Magnus Wahlström[4]

1 **University of Warsaw, PL,** cygan@mimuw.edu.pl
2 **University of Bergen, NO,** fomin@ii.uib.no
3 **Ben Gurion University – Beer Sheva, IL,** hermelin@bgu.ac.il
4 **Royal Holloway University of London, GB,** magnus.wahlstrom@rhul.ac.uk

Abstract

Dagstuhl Seminar 17041 "Randomization in Parameterized Complexity" took place from January 22nd to January 27th 2017 with the objective to bridge the gap between randomization and parameterized complexity theory. This report documents the talks held during the seminar as well as the open questions arised in the discussion sessions.

Seminar January 22–27, 2017 – http://www.dagstuhl.de/17041

1998 ACM Subject Classification F.2.2 Nonnumerical Algorithms and Problems, G.2.1 Combinatorics, G.2.2 Graph Theory

Keywords and phrases fixed-parameter tractability, intractability, parameterized complexity, randomness

Digital Object Identifier 10.4230/DagRep.7.1.103

Edited in cooperation with Marc Roth

1 Executive Summary

Marek Cygan
Fedor V. Fomin
Danny Hermelin
Magnus Wahlström

Randomization plays a prominent role in many subfields of theoretical computer science. Typically, this role is twofold: On the one hand, randomized methods can be used to solve essentially classical problems easier or more efficiently. In many cases, this allows for simpler, faster, and more appealing solutions for problems that have rather elaborate deterministic algorithms; in other cases, randomization provides the only known way to cope with the problem (e.g. polynomial identity testing, or deciding whether there exists a perfect matching with exactly b red edges in an edge-colored bipartite graph). On the other hand, there are also cases where randomness is intrinsic to the question being asked, such as the study of properties of random objects, and the search for algorithms which are efficient on average for various input distributions.

Parameterized complexity is an approach of handling computational intractability, where the main idea is to analyze the complexity of problems in finer detail by considering additional problem parameters beyond the input size. This area has enjoyed much success in recent years, and has yielded several new algorithmic approaches that help us tackle computationally challenging problems. While randomization already has an important role in parameterized complexity, for instance in techniques such as color-coding or randomized contractions, there is a common opinion within researchers of the field that the full potential of randomization has yet to be fully tapped.

The goal of this seminar was to help bridge this gap, by bringing together experts in the areas of randomized algorithms and parameterized complexity. In doing so, we hope to:

- Establish domains for simpler and/or more efficient FPT algorithms via randomization.
- Identify problems which intrinsically need randomization.
- Study parameterized problems whose instances are generated by some underlying distribution.
- Stimulate the development of a broadened role of randomness within parameterized complexity.

2 Table of Contents

3 Overview of Talks

3.1 Hardness in P

Amir Abboud (Stanford University, US)

The class P attempts to capture the efficiently solvable computational tasks. It is full of practically relevant problems, with varied and fascinating combinatorial structure.

In this talk, I will give an overview of a rapidly growing body of work that seeks a better understanding of the structure within P. Inspired by NP-hardness, the main tool in this approach are combinatorial reductions. Combining these reductions with a small set of plausible conjectures, we obtain tight lower bounds on the time complexity of many of the most important problems in P.

3.2 Towards Hardness of Approximation for Polynomial Time Problems

Arturs Backurs (MIT – Cambridge, US)

Proving hardness of approximation is a major challenge in the field of fine-grained complexity and conditional lower bounds in P. How well can the Longest Common Subsequence (LCS) or the Edit Distance be approximated by an algorithm that runs in near-linear time? In this paper, we make progress towards answering these questions. We introduce a framework that exhibits barriers for truly subquadratic and deterministic algorithms with good approximation guarantees. Our framework highlights a novel connection between deterministic approximation algorithms for natural problems in P and circuit lower bounds.

In particular, we discover a curious connection of the following form: if there exists a $\delta > 0$ such that for all $\epsilon > 0$ there is a deterministic $(1+\epsilon)$-approximation algorithm for LCS on two sequences of length n over an alphabet of size $n^{o(1)}$ that runs in $O(n^{2-\delta})$ time, then a certain plausible hypothesis is refuted, and the class E^{NP} does not have non-uniform linear size Valiant Series-Parallel circuits. Thus, designing a "truly subquadratic PTAS" for LCS is as hard as resolving an old open question in complexity theory.

3.3 Directed Hamiltonicity parameterized by the largest independent set

Andreas Björklund (Lund University, SE)

Joint work of Andreas Björklund, Petteri Kaski, Ioannis Koutis

We present a Monte Carlo algorithm deciding Hamiltonicity in n-vertex directed graphs in $O^*(3^{n-\mathsf{mis}(G)})$ time and polynomial space, where $\mathsf{mis}(G)$ is the size of the largest independent set in the graph. In particular, in bipartite graphs we get a $O^*(1.733^n)$ time and polynomial space algorithm improving over the $O^*(1.888^n)$ time and exponential space algorithm by Cygan et al. from STOC 2013.

3.4 Fine-grained dichotomies for the Tutte plane and Boolean #CSP

Cornelius Brand (Universität des Saarlandes, DE)

Joint work of Cornelius Brand, Holger Dell, Marc Roth

Jaeger, Vertigan, and Welsh proved a dichotomy for the complexity of evaluating the Tutte polynomial at fixed points: The evaluation is #P-hard almost everywhere, and the remaining points admit polynomial-time algorithms. Dell, Husfeldt, and Wahlén and Husfeldt and Taslaman, in combination with Curticapean, extended the #P-hardness results to tight lower bounds under the counting exponential time hypothesis #ETH, with the exception of the line $y = 1$, which was left open. We complete the dichotomy theorem for the Tutte polynomial under #ETH by proving that the number of all acyclic subgraphs of a given n-vertex graph cannot be determined in time $\exp(o(n))$ unless #ETH fails. Another dichotomy theorem we strengthen is the one of Creignou and Hermann for counting the number of satisfying assignments to a constraint satisfaction problem instance over the Boolean domain. We prove that all #P-hard cases are also hard under #ETH. The main ingredient is to prove that the number of independent sets in bipartite graphs with n vertices cannot be computed in time $\exp(o(n))$ unless #ETH fails. In order to prove our results, we use the block interpolation idea by Curticapean and transfer it to systems of linear equations that might not directly correspond to interpolation.

3.5 A Near-Linear Pseudopolynomial Time Algorithm for Subset Sum

Karl Bringmann (MPI für Informatik – Saarbrücken, DE)

Main reference K. Bringmann, "A Near-Linear Pseudopolynomial Time Algorithm for Subset Sum", in Proc. of the 28th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA 2017), pp. 1073–1084, SIAM, 2017.
URL http://dx.doi.org/10.1137/1.9781611974782.69

Given a set Z of n positive integers and a target value t, the SubsetSum problem asks whether any subset of Z sums to t. A textbook pseudopolynomial time algorithm by Bellman from 1957 solves SubsetSum in time $O(nt)$. Here we present a simple randomized algorithm running in time $\tilde{O}(n+t)$. This improves upon a classic result and is likely to be near-optimal, since it matches conditional lower bounds from SetCover and k-Clique. One of our main tools originated in the field of parameterized algorithms. We also present a new algorithm with pseudopolynomial time and polynomial space.

3.6 Relatively recent insights into counting small patterns

Radu Curticapean (Hungarian Academy of Sciences – Budapest, HU)

Joint work of Radu Curticapean, Holger Dell, Dániel Marx

We consider the problem of counting subgraphs. More specifically, we look at the following problems $\#\mathsf{Sub}(C)$ for fixed graph classes C: Given as input a graph H from C (the pattern) and another graph G (the host), the task is to count the occurrences of H as a subgraph in G. Our goal is to understand which properties of the pattern class C make the problem $\#\mathsf{Sub}(C)$ easy/hard. For instance, for the class of stars, we can solve this problem in linear time. For the class of paths however, it subsumes counting Hamiltonian paths and is hence $\#\mathsf{P}$-hard.

As it turns out, the notion of $\#\mathsf{P}$-hardness fails to give a sweeping dichotomy for the problems $\#\mathsf{Sub}(C)$, since there exist classes C of intermediate complexity. However, adopting the framework of fixed-parameter tractability, and parameterizing by the size of the pattern, it was shown in 2014 how to classify the problems $\#\mathsf{Sub}(C)$ as either polynomial-time solvable or $\#\mathsf{W}[1]$-hard: A class C lies on the polynomial-time side of this dichotomy iff the graphs appearing in C have vertex-covers of constant size.

In this talk, we introduce a new technique that allows us to view the subgraph counting problem from a new perspective. In particular, it allows for the following applications:

1. A greatly simplified proof of the 2014 dichotomy result, together with almost-tight lower bounds under ETH, which were not achievable before.
2. Faster algorithms for counting k-edge subgraphs, such as k-matchings, with running time n^{ck} for constants $c < 1$.

3.7 Finding Detours is Fixed-parameter Tractable

Holger Dell (Universität des Saarlandes, DE)

URL https://arxiv.org/abs/1607.07737
Joint work of Ivona Bezáková, Radu Curticapean, Holger Dell, Fedor V. Fomin

We consider the following natural "above guarantee" parameterization of the classical Longest Path problem: For given vertices s and t of a graph G, and an integer k, the problem Longest Detour asks for an (s, t)-path in G that is at least k longer than a shortest (s, t)-path. Using insights into structural graph theory, we prove that Longest Detour is fixed-parameter tractable (FPT) on undirected graphs and actually even admits a single-exponential algorithm, that is, one of running time $\mathsf{exp}(O(k)) \cdot \mathsf{poly}(n)$. This matches (up to the base of the exponential) the best algorithms for finding a path of length at least k.

Furthermore, we study the related problem Exact Detour that asks whether a graph G contains an (s, t)-path that is exactly k longer than a shortest (s, t)-path. For this problem, we obtain a randomized algorithm with running time about 2.746^k, and a deterministic algorithm with running time about 6.745^k, showing that this problem is FPT as well. Our algorithms for Exact Detour apply to both undirected and directed graphs.

3.8 Average-Case Analysis of Parameterized Problems

Tobias Friedrich (Hasso-Plattner-Institut – Potsdam, DE)

Joint work of Karl Bringmann, Tobias Friedrich, Danny Hermelin, Christian Hercher, Nikolaos Fountoulakis

Many computational problems are NP-hard and are therefore generally believed not to be solvable in polynomial time. Additional assumptions on the inputs are necessary to solve such problems efficiently. Two typical approaches are (i) parameterized complexity where we assume that a certain parameter of the instances is small, and (ii) average-case complexity where we assume a certain probability distribution on the inputs. There is a vast literature on both approaches, but very little about their intersection. Nevertheless, combining these two approaches seems natural and potentially useful in practice. The talk presents the following line of results:

- A hierarchy of parameterized average-case complexity classes [2].
- The W[1]-complete problem k-clique drops to an average-case analog of FPT for homogeneous Erdős-Rényi random graphs of all densities [2] and for inhomogeneous Chung-Lu random graphs with power-law exponent $\gamma > 2$ [4, 5].
- The bounded search tree paradigm allows analyzing average-case run times for a very relaxed graph model that only assumes stochastic independence of the edges. This is demonstrated for the parameterized problems k-Clique, Vertex Cover, and Hitting Set [unpublished].
- The Edge Cover Problem has no kernel of subexponential size in the worst-case (unless $\mathsf{P} = \mathsf{NP}$). We study a well-known set of reduction rules and prove that random intersection graphs are reduced completely by these rules [3].
- The geometric problem of computing the hypervolume indicator is W[1]-hard in the worst-case, but can be solved in expected FPT-time if the input is distributed at random on a d-dimensional simplex [1].

References

1 Karl Bringmann and Tobias Friedrich. Parameterized average-case complexity of the hypervolume indicator. In *Genetic and Evolutionary Computation Conference (GECCO)*, pages 575–582. ACM, 2013.

2 Nikolaos Fountoulakis, Tobias Friedrich, and Danny Hermelin. On the average-case complexity of parameterized clique. *Theoretical Computer Science*, 576:18–29, 2015.

3 Tobias Friedrich and Christian Hercher. On the kernel size of clique cover reductions for random intersection graphs. *Journal of Discrete Algorithms*, 34:128–136, 2015.

4 Tobias Friedrich and Anton Krohmer. Parameterized clique on scale-free networks. In *International Symposium on Algorithms and Computation (ISAAC)*, volume 7676 of *Lecture Notes in Computer Science*, pages 659–668. Springer, 2012.

5 Tobias Friedrich and Anton Krohmer. Parameterized clique on inhomogeneous random graphs. *Discrete Applied Mathematics*, 184:130–138, 2015.

3.9 Spanning Circuits in Regular Matroids

Petr A. Golovach (University of Bergen, NO)

Joint work of Fedor V. Fomin, Petr A. Golovach, Daniel Lokshtanov, Saket Saurabh
Main reference F. V. Fomin, P. A. Golovach, D. Lokshtanov, S. Saurabh, "Spanning Circuits in Regular Matroids", in Proc. of the 28th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA 2017), pp. 1433–1441, SIAM, 2017.
URL http://dx.doi.org/10.1137/1.9781611974782.93

We consider the fundamental Matroid Theory problem of finding a circuit in a matroid spanning a set T of given terminal elements. For graphic matroids this corresponds to the problem of finding a simple cycle passing through a set of given terminal edges in a graph. The algorithmic study of the problem on regular matroids, a superclass of graphic matroids, was initiated by Gavenčiak, Král', and Oum [ICALP'12], who proved that the case of the problem with $|T| = 2$ is fixed-parameter tractable (FPT) when parameterized by the length of the circuit. We extend the result of Gavenčiak, Král', and Oum by showing that for regular matroids

- the Minimum Spanning Circuit problem, deciding whether there is a circuit with at most ℓ elements containing T, is FPT parameterized by $k = \ell - |T|$;
- the Spanning Circuit problem, deciding whether there is a circuit containing T, is FPT parameterized by $|T|$.

We note that extending our algorithmic findings to binary matroids, a superclass of regular matroids, is highly unlikely: Minimum Spanning Circuit parameterized by ℓ is W[1]-hard on binary matroids even when $|T| = 1$. We also show a limit to how far our results can be strengthened by considering a smaller parameter. More precisely, we prove that Minimum Spanning Circuit parameterized by $|T|$ is W[1]-hard even on cographic matroids, a proper subclass of regular matroids.

3.10 Parameterized Traveling Salesman Problem: Beating the Average

Gregory Z. Gutin (Royal Holloway University of London, GB)

Joint work of Gregory Z. Gutin, Viresh Patel
Main reference G. Gutin, V. Patel, "Parameterized Traveling Salesman Problem: Beating the Average", SIAM J. Discrete Math., 30(1):220–238, SIAM, 2016.
URL http://dx.doi.org/10.1137/140980946

In the traveling salesman problem (TSP), we are given a complete graph K_n together with an integer weighting w on the edges of K_n, and we are asked to find a Hamilton cycle of K_n of minimum weight. Let $h(w)$ denote the average weight of a Hamilton cycle of K_n for the weighting w. Vizing in 1973 asked whether there is a polynomial-time algorithm which always finds a Hamilton cycle of weight at most $h(w)$. He answered this question in the affirmative and subsequently Rublineckii, also in 1973, and others described several other TSP heuristics satisfying this property. We prove a considerable generalization of Vizing's result: for each fixed k, we give an algorithm that decides whether, for any input edge weighting w of K_n, there is a Hamilton cycle of K_n of weight at most $h(w) - k$ (and constructs such a cycle if it exists). For k fixed, the running time of the algorithm is polynomial in n, where the degree of the polynomial does not depend on k (i.e., the generalized Vizing problem is fixed-parameter tractable with respect to the parameter k).

3.11 How proofs are prepared at Camelot

Petteri Kaski (Aalto University, FI)

Joint work of Andreas Björklund, Petteri Kaski
Main reference A. Björklund, P. Kaski, "How Proofs are Prepared at Camelot: Extended Abstract", in Proc. of the 2016 ACM Symposium on Principles of Distributed Computing (PODC 2016), pp. 391–400, ACM, 2016.
URL http://dx.doi.org/10.1145/2933057.2933101

We study a design framework for robust, independently verifiable, and workload-balanced distributed algorithms working on a common input. The framework builds on recent noninteractive Merlin–Arthur proofs of batch evaluation of Williams [31st IEEE Colloquium on Computational Complexity (CCC'16, May 29-June 1, 2016, Tokyo), 2:117] with the basic observation that Merlin's magic is not needed for batch evaluation: mere Knights can prepare the independently verifiable proof, in parallel, and with intrinsic error-correction.

As our main technical result, we show that the k-cliques in an n-vertex graph can be counted and verified in per-node $O(n(\omega+\epsilon)\frac{k}{6})$ time and space on $O(n(\omega+\epsilon)\frac{k}{6})$ compute nodes, for any constant $\epsilon > 0$ and positive integer k divisible by 6, where $2 \leq \omega < 2.3728639$ is the exponent of square matrix multiplication over the integers. This matches in total running time the best known sequential algorithm, due to Nešetřil and Poljak [Comment. Math. Univ. Carolin. 26 (1985) 415–419], and considerably improves its space usage and parallelizability. Further results include novel algorithms for counting triangles in sparse graphs, computing the chromatic polynomial of a graph, and computing the Tutte polynomial of a graph.

3.12 Improved algebraic algorithms for out-branchings problems

Yiannis Koutis (University of Puerto Rico – Rio Piedras, PR)

We present an $O^*(2^k)$ algorithm for deciding if a directed graph contains an out-branching with at least k internal nodes. We also present an algorithm for detecting out-branchings with at least k leaves and at most s internal nodes with out-degree greater than 1. The algorithm runs in time $O^*(2^{k+s})$, and for certain values of s it improves upon the previous upper bounds for the k-leaf problem. The algorithms are algebraic and work via reductions to two non-standard problems concerning monomial detection in multivariate polynomials.

3.13 Improving TSP tours using dynamic programming over tree decomposition

Łukasz Kowalik (University of Warsaw, PL)

Joint work of Marek Cygan, Łukasz Kowalik, Arkadiusz Socała

Given a traveling salesman problem (TSP) tour H in graph G a k-move is an operation which removes k edges from H, and adds k edges of G so that a new tour H' is formed. The popular k-OPT heuristics for TSP finds a local optimum by starting from an arbitrary tour H and then improving it by a sequence of k-moves.

Until 2016, the only known algorithm to find an improving k-move for a given tour was the naive solution in time $O(n^k)$. At ICALP'16 de Berg, Buchin, Jansen and Woeginger showed an $O(n^{\lfloor \frac{2}{3k} \rfloor + 1})$-time algorithm.

We show an algorithm which runs in $O(n^{(\frac{1}{4}+\epsilon_k)k})$ time, where $\lim \epsilon_k = 0$. We are able to show that it improves over the state of the art for every $k = 5, \ldots, 10$. For the most practically relevant case $k = 5$ we provide a slightly refined algorithm running in $O(n^{3.4})$ time. We also show that for the $k = 4$ case, improving over the $O(n^3)$-time algorithm of de Berg et al. would be a major breakthrough: an $O(n^{3-\epsilon})$-time algorithm for any $\epsilon > 0$ would imply an $O(n^{3-\delta})$-time algorithm for the APSP problem, for some $\delta > 0$.

3.14 A Randomized Polynomial Kernelization for Vertex Cover with a Smaller Parameter

Stefan Kratsch (Universität Bonn, DE)

Main reference S. Kratsch, "A Randomized Polynomial Kernelization for Vertex Cover with a Smaller Parameter", in Proc. of the 24th Annual European Symposium on Algorithms (ESA 2016), LIPIcs, Vol. 57, pp. 59:1-59:17, Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2016.
URL http://dx.doi.org/10.4230/LIPIcs.ESA.2016.59

In the Vertex Cover problem we are given a graph $G = (V, E)$ and an integer k and have to determine whether there is a set $X \subseteq V$ of size at most k such that each edge in E has at least one endpoint in X. The problem can be easily solved in time $O^*(2^k)$, making it fixed-parameter tractable (FPT) with respect to k. While the fastest known algorithm takes only time $O^*(1.2738^k)$, much stronger improvements have been obtained by studying *parameters that are smaller than k*. Apart from treewidth-related results, the arguably best algorithm for Vertex Cover runs in time $O^*(2.3146^p)$, where $p = k - LP(G)$ is only the excess of the solution size k over the best fractional vertex cover (Lokshtanov et al. TALG 2014). Since $p \leq k$ but k cannot be bounded in terms of p alone, this strictly increases the range of tractable instances.

Recently, Garg and Philip (SODA 2016) greatly contributed to understanding the parameterized complexity of the Vertex Cover problem. They prove that $2LP(G) - MM(G)$ is a lower bound for the vertex cover size of G, where $MM(G)$ is the size of a largest matching of G, and proceed to study parameter $\ell = k - (2LP(G) - MM(G))$. They give an algorithm of running time $O^*(3^\ell)$, proving that Vertex Cover is FPT in ℓ. It can be easily observed that $\ell \leq p$ whereas p cannot be bounded in terms of ℓ alone. We complement the work of Garg and Philip by proving that Vertex Cover admits a randomized polynomial kernelization in terms of ℓ, i.e., an efficient preprocessing to size polynomial in ℓ. This improves over parameter $p = k - LP(G)$ for which this was previously known (Kratsch and Wahlström FOCS 2012).

3.15 Gap Amplification using Bipartite Random Graphs

Bingkai Lin (National Institute of Informatics – Tokyo, JP)

Gap amplification transformation plays an important role in proving hardness of approximation results. This talk presents a new method to construct gap amplification reduction for parameterized optimization problems. First, I will review the threshold phenomenon of random graphs $G(n, p)$ containing a bipartite complete subgraph . Then I will show its application on ruling out super-polynomial time algorithms for approximating Maximum k-Set Intersection and Minimum Set Cover to some ratios.

3.16 Lossy Kernelization I

M. S. Ramanujan (TU Wien, AT)

Joint work of Daniel Lokshtanov, Fahad Panolan, M. S. Ramanujan, Saket Saurabh
Main reference D. Lokshtanov, F. Panolan, M. S. Ramanujan, S. Saurabh, "Lossy Kernelization", arXiv:1604.04111v2 [cs.DS], 2016.
URL https://arxiv.org/abs/1604.04111v2

Introductory talk on a new framework for analyzing the performance of preprocessing algorithms. This framework builds on the notion of kernelization from parameterized complexity. However, as opposed to the original notion of kernelization, this framework combines very well with approximation algorithms and heuristics.

3.17 Lossy Kernelization II: Cycle Packing

Fahad Panolan (University of Bergen, NO)

Joint work of Daniel Lokshtanov, Fahad Panolan, M. S. Ramanujan, Saket Saurabh
Main reference D. Lokshtanov, F. Panolan, M. S. Ramanujan, S. Saurabh, "Lossy Kernelization", arXiv:1604.04111v2 [cs.DS], 2016.
URL https://arxiv.org/abs/1604.04111v2

In this talk we see an example of Lossy Kernelization – Disjoint Factors. Disjoint Factors problem is closely related to Cycle Packing. We prove that Disjoint Factors admits a Polynomial Sized Approximate Kernelization Scheme (PSAKS).

3.18 Lossy Kernelization, III: Lower Bounds

Daniel Lokshtanov (University of Bergen, NO)

Joint work of Daniel Lokshtanov, Fahad Panolan, M. S. Ramanujan, Saket Saurabh
Main reference D. Lokshtanov, F. Panolan, M. S. Ramanujan, S. Saurabh, "Lossy Kernelization", arXiv:1604.04111v2 [cs.DS], 2016.
URL https://arxiv.org/abs/1604.04111v2

We show how to combine the tecniques for showing kernelization lower bounds with the methods for showing hardness of approximation to rule out approximate kernels of polynomial size for concrete problems. We outline proofs that the longest path problem parameterized by solution size, and the set cover problem parameterized by the size of the universe do not admit constant factor approximate kernels of polynomial size.

3.19 Exponential Time Paradigms Through the Polynomial Time Lens

Jesper Nederlof (TU Eindhoven, NL)

Joint work of Andy Drucker, Jesper Nederlof, Rahul Santhanam
Main reference A. Drucker, J. Nederlof, R. Santhanam, "Exponential Time Paradigms Through the Polynomial Time Lens", in Proc. of the 24th Annual European Symposium on Algorithms (ESA 2016), LIPIcs, Vol. 57, pp. 36:1-36:14, Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2016.
URL http://dx.doi.org/10.4230/LIPIcs.ESA.2016.36

We propose a general approach to modelling algorithmic paradigms for the exact solution of NP-hard problems. Our approach is based on polynomial time reductions to succinct versions of problems solvable in polynomial time. We use this viewpoint to explore and compare the power of paradigms such as branching and dynamic programming, and to shed light on the true complexity of various problems.

In this talk I will mainly talk about lower bounds for OPP algorithms. For example, if there is a polynomial time algorithm that, given a planar graph, outputs a maximum independent set of n vertices with probability $\mathsf{exp}(-n^{1-\epsilon})$ for some $\epsilon > 0$, then $\mathsf{NP} \subseteq \mathsf{coNP/poly}$. I will also outline connections with "AND-compositions" from kernelization theory.

3.20 Faster Space-Efficient Algorithms for Subset Sum, k-Sum and Related Problems

Jesper Nederlof (TU Eindhoven, NL)

Joint work of Nikhil Bansal, Shashwat Garg, Jesper Nederlof, Nikhil Vyas
Main reference N. Bansal, S. Garg, J. Nederlof, N. Vyas, "Faster Space-Efficient Algorithms for Subset Sum, k-Sum and Related Problems", arXiv:1612.02788v1 [cs.DS], 2016.
URL https://arxiv.org/abs/1612.02788v1

We present a randomized Monte Carlo algorithm that solves a given instance of Subset Sum on n integers using $O^*(2^{0.86n})$ time and $O^*(1)$ space, where $O^*()$ suppresses factors polynomial in the input size. The algorithm assumes random access to the random bits used. The same result can be obtained for Knapsack on n items, and the same methods also have consequences for the k-Sum problem.

3.21 Subexponential Parameterized Algorithms for Planar Graphs, Apex-Minor-Free Graphs and Graphs of Polynomial Growth via Low Treewidth Pattern Covering

Marcin Pilipczuk (University of Warsaw, PL) and Dániel Marx (Hungarian Academy of Sciences – Budapest, HU)

Joint work of Fedor V. Fomin, Daniel Lokshtanov, Dániel Marx, Marcin Pilipczuk, Michal Pilipczuk, Saket Saurabh
Main reference F. V. Fomin, D. Lokshtanov, D. Marx, Ma. Pilipczuk, Mi. Pilipczuk, S. Saurabh, "Subexponential Parameterized Algorithms for Planar and Apex-Minor-Free Graphs via Low Treewidth Pattern Covering", in Proc. of the 57th Annual Symposium on Foundations of Computer Science (FOCS 2016), pp. 515–524, IEEE, 2016.
URL http://dx.doi.org/10.1109/FOCS.2016.62

We prove the following theorem. Given a planar graph G and an integer k, it is possible in polynomial time to randomly sample a subset A of vertices of G with the following properties:

- A induces a subgraph of G of treewidth $O(\sqrt{k}\log k)$, and
- for every connected subgraph H of G on at most k vertices, the probability that A covers the whole vertex set of H is at least $(2^{O(\sqrt{k}\log^2 k)} \cdot n^{O(1)})^{-1}$, where n is the number of vertices of G.

Together with standard dynamic programming techniques for graphs of bounded treewidth, this result gives a versatile technique for obtaining (randomized) subexponential parameterized algorithms for problems on planar graphs, usually with running time bound $2^{O(\sqrt{k}\log^2 k)}n^{O(1)}$. The technique can be applied to problems expressible as searching for a small, connected pattern with a prescribed property in a large host graph; examples of such problems include Directed k-Path, Weighted k-Path, Vertex Cover Local Search, and Subgraph Isomorphism, among others. Up to this point, it was open whether these problems can be solved in subexponential parameterized time on planar graphs, because they are not amenable to the classic technique of bidimensionality. Furthermore, all our results hold in fact on any class of graphs that exclude a fixed apex graph as a minor, in particular on graphs embeddable in any fixed surface. We also provide a similar statement for graph classes of polynomial growth.

3.22 Exact Algorithms via Monotone Local Search

Saket Saurabh (The Institute of Mathematical Sciences, India, IN)

Joint work of Daniel Lokshtanov, Serge Gaspers, Fedor Fomin, Saket Saurabh

In a vertex subset problem we are given as input a universe U of size n, and a family F of subsets of the universe defined implicitly from the input. The task is to find a subset S in F of smallest possible size. For an example the Vertex Cover problem is a subset problem where input is a graph G, the universe is the vertex set of G, and the family F is the family of all vertex covers of G. Here a vertex set S is a vertex cover of G if every edge of G has at least one endpoint in S. Many problems, such as Vertex Cover, Feedback Vertex Set, Hitting Set and Minimum Weight Satisfiability can be formalized as vertex subset problems. The trivial algorithm for such problems runs in time 2^n. We show that (essentially) any vertex subset problem that admits an FPT algorithm with running time $c^{k}n^{O(1)}$, where c is a constant and

k is the size of the optimal solution, also admits an algorithm with running time $(2 - \frac{1}{c})^n$. In one stroke this theorem improves the best known exact exponential time algorithms for a number of problems, and gives tighter combinatorial bounds for several well-studied objects. The most natural variant of our algorithm is randomized, we also show how to get a deterministic algorithm with the same running time bounds, up to a sub-exponential factor in the running time. Our de-randomization relies on a new pseudo-random construction that may be of independent interest.

3.23 Backdoors for Constraint Satisfaction

Stefan Szeider (TU Wien, AT)

Joint work of Robert Ganian, Serge Gaspers, Neeldhara Misra, Sebastian Ordyniak, M. S Ramanujan, Stefan Szeider, Stanislav Živný
Main reference S. Gaspers, S. Ordyniak, S. Szeider, "Backdoor Sets for CSP", in The Constraint Satisfaction Problem: Complexity and Approximability, Dagstuhl Follow-Ups, Vol. 7, pp. 137–157, Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2017.
URL http://dx.doi.org/10.4230/DFU.Vol7.15301.137

We will review some recent parameterised complexity results for the Constraint Satisfaction Problem (CSP), considering parameters that arise from strong backdoor sets into CSP classes defined by tractable constraint languages. The language restrictions have recently stepped into the spotlight because of the recently claimed solution of the long-standing Dichotomy Conjecture. One of the results we will present is based on a novel combination of backdoor sets and treewidth.

3.24 Parameterized Algorithms for Matrix Factorization Problems

David P. Woodruff (IBM Almaden Center – San Jose, US)

I will give a survey on parameterized algorithms for matrix factorization problems, focusing on non-negative matrix factorization, ℓ_1 low rank factorization, tensor factorization, and weighted low rank approximation.

3.25 k-Path of Algorithms

Meirav Zehavi (University of Bergen, NO)

An overview of several algorithms for the k-Path problem and the tools employed to de-randomize them, including a presentation of a simple algorithm for the Longest Cycle problem.

4 Open problems

4.1 FPT-approximation of bandwidth

Daniel Lokshtanov (University of Bergen, NO)

Bandwidth
Input: An undirected graph $G = (V, E)$, integer k.
Question: Is there an ordering (injective function) $\pi : V \to \{1, \ldots, |V|\}$, such that $\max_{uv \in E} |\pi(u) - \pi(v)| \leq k$.

▶ **Open Problem 1.** *Is there an FPT-approximation on general graphs parameterized by k? In particular none of the following is known:*
- *Is there $(1 + \epsilon)$-approximation in FPT time?*
- *Is there constant approximation in FPT time?*
- *Is there $f(k)$-approximation in FPT time?*

Relevant reference: In [11] a polynomial time $k^{O(k)}$-approximation is shown for trees and graphs of bounded treelength.

4.2 Time and space complexity of k-LCS

Michał Pilipczuk (University of Warsaw, PL)

k-Longest Common Subsequence (k-LCS)
Input: alphabet Σ, strings $s_1, \ldots, s_k \in \Sigma^*$.
Question: what is the longest common subsequence of all the strings s_i.

The standard dynamic programming has running time and space complexity $O(n^k)$. By Savitch's theorem we can reduce the space complexity to $\mathsf{poly}(k, n)$ at the cost of increasing the running time to $n^{O(k \log n)}$.

▶ **Open Problem 2.** *Is k-LCS solvable in $n^{f(k)}$ time and FPT space?*

Relevant reference: in [21] a connection is proved between this open problem and the question of space efficient algorithms for bounded treewidth graphs. Other relevant reference: [12].

4.3 Fine-grained complexity of k-LCS

Karl Bringmann (MPI für Informatik – Saarbrücken, DE)

We can solve k-LCS (defined above) in time $O(n^k)$, but under SETH there is no $O(n^{k-\epsilon})$ time algorithm $|\Sigma| = \Omega(k)$ [1]. On the other hand we know for $|\Sigma| = O(1)$ the problem is W[1]-hard and there is no $n^{o(k)}$ time algorithm [19].

▶ **Open Problem 3.** *Is there an $O(n^{(1-\epsilon_\Sigma)k})$ time algorithm?*

4.4 Fine-grained complexity of Hitting Set w.r.t. VC dimension

Karl Bringmann (MPI für Informatik – Saarbrücken, DE)

Hitting Set
Input: a set family $\mathcal{F} \subseteq 2^U$, integer k.
Question: Is there a set $X \subseteq U$ of size at most k, such that X interesects each set in $\mathcal{F}$.

We know that Hitting Set can be solved in time $n^{k+o(1)}$ (for $k \geq 7$), and under the Strong Exponential Time Hypothesis (SETH) no $O(n^{k-\epsilon})$ time algorithm exists [18].

▶ **Definition 1.** We say that a set $X \subseteq U$ is *shattered* by a set family $\mathcal{F} \subseteq 2^U$ if the family $\{X \cap S : S \in \mathcal{F}\}$ contains all the subsets of X. The VC dimension of $\mathcal{F}$ is the largest cardinality of a set X, such that X is shattered by $\mathcal{F}$.

It is known that for VC = 1 the Hitting Set problem is polynomial time solvable, while for VC = 2 the problem becomes W[1]-hard and does not admit $n^{o(\frac{k}{\log k})}$ time algorithm [6].

▶ **Open Problem 4.** *Is there $O(n^{(1-\epsilon_{VC})k})$ time algorithm for the Hitting Set problem?*

4.5 FPT-approximation of VC dimension

Bingkai Lin (National Institute of Informatics – Tokyo, JP)

▶ **Open Problem 5.** *Is there a constant-factor FPT-time approximation algorithm for VC dimension (defined above)?*

4.6 Better approximation of Dominating Set

Bingkai Lin (National Institute of Informatics – Tokyo, JP)

Dominating Set
Input: an undirected graph G, an integer k.
Question: is there a set $X \subseteq V(G)$ of size at most k, such that each vertex of G is in X or has a neighbour in X?

It is well known that Dominating Set admits polynomial time $\ln(n)$-approximation algorithm as well as $n^{O(k)}$ time exact algorithm.

▶ **Open Problem 6.** *Is there an $o(\ln n)$-approximation algorithm for the Dominating Set problem running in time $n^{k-\epsilon}$?*

4.7 Orthogonal Vectors for Subset Sum

Jesper Nederlof (TU Eindhoven, NL)

Orthogonal Vectors for Subset Sum (OVSS)
Input: $\mathcal{A}, \mathcal{B} \subseteq \binom{[d]}{d/4}$.
Question: is there $A \in \mathcal{A}$, $B \in \mathcal{B}$ such that $A \cap B = \emptyset$?

We are satisfied with any algorithm with constant error probability. For an integer d, denote $[d] = \{1, \ldots, d\}$ and $\binom{[d]}{d/4}$ for the set of all subsets of $[d]$ of size $d/4$. Let $h(\cdot)$ denote the binary entropy function and $\tilde{O}$ omit factors polynomial in d.

▶ **Open Problem 7.** *Solve OVSS in time $\tilde{O}\left((|\mathcal{A}| + \mathcal{B}|) \cdot \frac{2^{(1-\epsilon)d}}{\binom{d}{d/4}}\right)$ for $\epsilon > 0$.*

Observations: Let $\alpha = 1 - h(1/4) \approx 0.1888$. Note that $2^{\alpha d} = 2^d / \binom{d}{d/4}$.

- There is an $\tilde{O}((|\mathcal{A}| + |\mathcal{B}|)2^{\alpha d})$ time algorithm based on representative sets (see [16] for an extended version of this open problem statement outlining the algorithm).
- If $|\mathcal{A}| \leq 2^{\alpha' d}$ or $|\mathcal{B}| \leq 2^{\alpha' d}$ for $\alpha' < \alpha$, then trivial enumeration works. Moreover, by directly using the improvements over this trivial enumeration from [2, 7, 13], we may in fact assume $|\mathcal{A}|, |\mathcal{B}| \geq 2^{(\alpha+\delta)d}$ for some $\delta > 0$.
- If $|\mathcal{A}| > 2^{\beta d}$, where $\beta > h(1/4) - (1 - h(1/4)) \approx 0.6223$, an algorithm of Björklund et al. [5] works: it runs in time $\tilde{O}((|\downarrow \mathcal{A}| + |\downarrow \mathcal{B}|)) \leq \tilde{O}(2^{h(1/4)d})$, where for a set family $\mathcal{F}$, $\downarrow \mathcal{F}$ denotes the sets of subsets of elements of $\mathcal{F}$.
- In fact, $|\downarrow \mathcal{A}|$ can be upper bounded by $\tilde{O}(\max_\lambda \min\{|\mathcal{A}|\binom{d/4}{\lambda d}, \binom{d}{\lambda d}\})$. After a small calculation, this gives that the algorithm from [5] is fast enough whenever $\beta > 0.525$.

▶ **Open Problem 8.** *Does there exist for some constant $c > 0$ an algorithm that, given $z = 2^{cd}$ instances $(\mathcal{A}_1, \mathcal{B}_1), \ldots, (\mathcal{A}_z, \mathcal{B}_z)$ of OVSS, detects whether any instance is a YES-instance in time $(\sum_{i=1}^{z}(|\mathcal{A}_i| + |\mathcal{B}_i|))2^{(\alpha-\epsilon)d}$, for $\epsilon > 0$?*

Note Open Problem 8 relaxes Open Problem 7 as it asks whether exponentially many instances of OVSS can be solved fast in an amortized sense.

Motivation: Following the approach of [4], a positive answer would imply an $\tilde{O}(2^{(.5-\epsilon)n})$ time algorithm for n-integer subset sum for some $\epsilon > 0$.

4.8 Fixed parameter tractability of Weighted Low Rank Approximation

David P. Woodruff (IBM Almaden Center – San Jose, US)

Weighted Low Rank Approximation
Input: $n \times n$ matrix A over reals, rank bound $r = O(1)$, weight matrix $W \in \mathbb{R}^{n \times n}$
Goal: find a rank r matrix B such that the weighted Frobenius norm of the difference $|W \circ (A - B)|_F = \sum (W_{i,j} \cdot (A_{i,j} - B_{i,j})^2)$ is small, i.e., at most $1.01 \cdot OPT$

We assume the entries of A and W are integers in the range $\{-M, -M+1, \ldots, M\}$ for an integer $M \leq 2^{poly(n)}$, i.e., that the entries of A and W can be specified using $poly(n)$ bits.

► **Open Problem 9.** *Is there an FPT algorithm for this problem when parameterized by the rank of the weight matrix W?*

It is known [22] that there is an $n^{O(k)}$ upper bound and conditional $2^{\Omega(k)}$ lower bound.

4.9 Short resolution refutations for SAT when parameterized by treewidth

Stefan Szeider (TU Wien, AT)

We consider propositional formulas in conjunctive normal form (CNF), given as a set of clauses, where each clause is a set of literals, e.g., $F = \{\{x, y\}, \{x, \bar{y}, z\}, \{\bar{x}, y\}, \{\bar{x}, \bar{y}\}, \{\bar{z}\}\}$.

► **Definition 1.** A clause C is the *resolvent* of clauses C_1 and C_2 if there is exactly one variable x such that $x \in C_1$, $\bar{x} \in C_2$, and $C = (C_1 \setminus \{x\}) \cup (C_2 \setminus \{\bar{x}\})$.

A *resolution refutation* of a formula F is a vertex-labeled dag with exactly one sink where each vertex has in-degree 0 or 2. Each node is labeled with a clause as follows: (i) each source is labeled with a clause from F, (ii) each non-source is labeled with the resolvent of the clauses labeling its predecessors, and (iii) the clause which labels the sink is empty.

The *size* of a resulution refutation is the number of its vertices.

It is known that a formula is unsatisfiable if and only if it has a resolution refutation.

► **Definition 2.** The *primal graph* $P(F)$ of a formula F is the graph whose vertices are the variables of F, where two vertices are connected by an edge iff the corresponding variables appear together (negated or unnegated) in some clause.

The *incidence graph* $I(F)$ is the bipartite graph between variables and clauses where two vertices are connected by an edge iff the corresponding variable appears (negated or unnegated) in the corresponding clause.

It is known that for any formula F the treewidth of its incidence graph is at most the treewidth of its primal graph plus one:

$$\mathsf{tw}(I(F)) \leq tw(P(F)) + 1\,.$$

Also, it is known that #SAT is FPT when parameterized by $\mathsf{tw}(I(F))$ and $\mathsf{tw}(P(F))$. Further, it is known that every unsatisfiable formula F has a resolution refutation of FPT size when parameterized by $\mathsf{tw}(P(F))$.

▶ **Open Problem 10.** *Is there always a resolution refutation of FPT size when parameterized by* $\mathsf{tw}(I(F))$*?*

4.10 Small universal Steiner tree covers

Marcin Pilipczuk (University of Warsaw, PL)

Let G be a graph embedded on the plane in such a manner that the outerface of G, denoted henceforth ∂G, is a simple cycle of length k. For a set $T \subseteq V(\partial G)$ and $A \subseteq V(G)$, we say that A *covers* an optimal Steiner tree for T if there exists an optimum Steiner tree in G with terminals T, such that every vertex of degree at least three in this tree lies in A. A set A is a *universal Steiner tree cover* in G if A covers an optimal Steiner tree for every $T \subseteq V(\partial G)$.

In [20] we have shown an existence of a universal Steiner tree cover of size bounded polynomially in k, but the degree of the bound is above 100. On the other hand, we do not know any example that is significantly worse than a grid of perimeter k.

▶ **Open Problem 11.** *Prove or disprove the following statement: for every such G, there exists a universal Steiner tree cover of size $\widetilde{O}(k^2)$.*

4.11 Even Set

Dániel Marx (Hungarian Academy of Sciences – Budapest, HU)

Even Set
Input: Set system $\mathcal{S}$ over a universe U, integer k.
Find: 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.
- Minimum distance in a linear code over a binary alphabet.

▶ **Open Problem 12.** *What is the parameterized complexity of Even Set? Is it fixed parameter tractable?*

4.12 FPT-approximation of Maximum Clique and Minimum Dominating Set

Dániel Marx (Hungarian Academy of Sciences – Budapest, HU)

► **Open Problem 13.** *Can Maximum Clique (Minimum Dominating Set) be approximated in FPT time? I.e., is there an algorithm running in time $f(k) \cdot n^{O(1)}$ that, given a graph G and an integer k, finds a $g(k)$-clique (dominating set of size $g(k)$) for some unbounded nondecreasing function g or correctly states that there is no k-clique (dominating set of size k) in G?*

4.13 Polynomial (Turing) Kernels

Dániel Marx (Hungarian Academy of Sciences – Budapest, HU)

► **Open Problem 14.** *Do the following problems have polynomial kernels?*

- *Directed Feedback Vertex Set*
- *Multiway Cut (with arbitrary number t of terminals)*
- *Planar Vertex Deletion*

Does k-Path have a polynomial Turing kernel?

4.14 Directed Odd Cycle Traversal

Dániel Marx (Hungarian Academy of Sciences – Budapest, HU)

Directed Odd Cycle Traversal
Input: Directed graph G, integer k.
Find: A set $X \subseteq U$ of at most k vertices such that $G - X$ has no directed cycle of odd length.

This problem generalizes

- Directed Feedback Vertex Set [9]
- Odd Cycle Transversal [23]
- Directed S-Cycle Transversal [10]

► **Open Problem 15.** *What is the parameterized complexity of Directed Odd Cycle Traversal?*

4.15 Square root phenomenon

Dániel Marx (Hungarian Academy of Sciences – Budapest, HU)

▶ **Open Problem 16.** *Are there* $2^{O(\sqrt{k}\cdot\text{polylog}(k))}n^{O(1)}$ *time FPT algorithms for planar problems?*

Natural targets are

- Steiner Tree
- Directed Steiner Tree
- Directed Subset TSP

What about counting problems?

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

4.16 Disjoint paths / minor testing

Dániel Marx (Hungarian Academy of Sciences – Budapest, HU)

The best known parameter dependence for the k-disjoint paths problem and H-minor testing seems to be triple exponential [15] using [8]. For planar graphs [3] gave an $2^{2^{\text{poly}(k)}}n^{O(1)}$ algorithm.

▶ **Open Problem 17.** *Are there* $2^{\text{poly}(k)}n^{O(1)}$ *time algorithms for planar or general graphs?*

References

1 Amir Abboud, Arturs Backurs, and Virginia Vassilevska Williams. Tight hardness results for LCS and other sequence similarity measures. In Venkatesan Guruswami, editor, *IEEE 56th Annual Symposium on Foundations of Computer Science, FOCS 2015, Berkeley, CA, USA, 17-20 October, 2015*, pages 59–78. IEEE Computer Society, 2015. URL: http://ieeexplore.ieee.org/xpl/mostRecentIssue.jsp?punumber=7352273, doi:10.1109/FOCS.2015.14.

2 Amir Abboud, Richard Ryan Williams, and Huacheng Yu. More applications of the polynomial method to algorithm design. In Indyk [14], pages 218–230. doi:10.1137/1.9781611973730.17.

3 Isolde Adler, Stavros G Kolliopoulos, Philipp Klaus Krause, Daniel Lokshtanov, Saket Saurabh, and Dimitrios Thilikos. Tight bounds for linkages in planar graphs. In *International Colloquium on Automata, Languages, and Programming*, pages 110–121. Springer, 2011.

4 Per Austrin, Petteri Kaski, Mikko Koivisto, and Jesper Nederlof. Dense subset sum may be the hardest. In Ollinger and Vollmer [17], pages 13:1–13:14. doi:10.4230/LIPIcs.STACS.2016.13.

5 Andreas Björklund, Thore Husfeldt, Petteri Kaski, and Mikko Koivisto. Counting paths and packings in halves. In Amos Fiat and Peter Sanders, editors, *Algorithms – ESA 2009, 17th Annual European Symposium, Copenhagen, Denmark, September 7-9, 2009. Proceedings*, volume 5757 of *Lecture Notes in Computer Science*, pages 578–586. Springer, 2009. doi:10.1007/978-3-642-04128-0_52.

6 Karl Bringmann, László Kozma, Shay Moran, and N. S. Narayanaswamy. Hitting set for hypergraphs of low vc-dimension. In Piotr Sankowski and Christos D. Zaroliagis, editors, *24th Annual European Symposium on Algorithms, ESA 2016, August 22-24, 2016, Aarhus, Denmark*, volume 57 of *LIPIcs*, pages 23:1–23:18. Schloss Dagstuhl – Leibniz-Zentrum fuer Informatik, 2016. URL: http://www.dagstuhl.de/dagpub/978-3-95977-015-6, doi:10.4230/LIPIcs.ESA.2016.23.

7 Timothy M. Chan. Speeding up the four russians algorithm by about one more logarithmic factor. In Indyk [14], pages 212–217. doi:10.1137/1.9781611973730.16.

8 Chandra Chekuri and Julia Chuzhoy. Degree-3 treewidth sparsifiers. In *Proceedings of the Twenty-Sixth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 242–255. Society for Industrial and Applied Mathematics, 2015.

9 Jianer Chen, Yang Liu, Songjian Lu, Barry O'sullivan, and Igor Razgon. A fixed-parameter algorithm for the directed feedback vertex set problem. *Journal of the ACM (JACM)*, 55(5):21, 2008.

10 Rajesh Chitnis, Marek Cygan, Taghi Hajiaghayi, Mohammad, Marcin Pilipczuk, and Michal Pilipczuk. Designing fpt algorithms for cut problems using randomized contractions. In *Foundations of Computer Science (FOCS), 2012 IEEE 53rd Annual Symposium on*, pages 460–469. IEEE, 2012.

11 Markus Sortland Dregi and Daniel Lokshtanov. Parameterized complexity of bandwidth on trees. In Javier Esparza, Pierre Fraigniaud, Thore Husfeldt, and Elias Koutsoupias, editors, *Automata, Languages, and Programming – 41st International Colloquium, ICALP 2014, Copenhagen, Denmark, July 8-11, 2014, Proceedings, Part I*, volume 8572 of *Lecture Notes in Computer Science*, pages 405–416. Springer, 2014. doi:10.1007/978-3-662-43948-7_34.

12 Michael Elberfeld, Christoph Stockhusen, and Till Tantau. On the space and circuit complexity of parameterized problems: Classes and completeness. *Algorithmica*, 71(3):661–701, 2015. doi:10.1007/s00453-014-9944-y.

13 Russell Impagliazzo, Shachar Lovett, Ramamohan Paturi, and Stefan Schneider. 0-1 integer linear programming with a linear number of constraints. *CoRR*, abs/1401.5512, 2014. URL: http://arxiv.org/abs/1401.5512.

14 Piotr Indyk, editor. *Proceedings of the Twenty-Sixth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2015, San Diego, CA, USA, January 4-6, 2015*. SIAM, 2015. doi:10.1137/1.9781611973730.

15 Ken-ichi Kawarabayashi and Paul Wollan. A shorter proof of the graph minor algorithm: the unique linkage theorem. In *Proceedings of the forty-second ACM symposium on Theory of computing*, pages 687–694. ACM, 2010.

16 Jesper Nederlof. Faster subset sum via improved orthogonal vectors? http://www.win.tue.nl/ jnederlo/problem.pdf.

17 Nicolas Ollinger and Heribert Vollmer, editors. *33rd Symposium on Theoretical Aspects of Computer Science, STACS 2016, February 17-20, 2016, Orléans, France*, volume 47 of *LIPIcs*. Schloss Dagstuhl – Leibniz-Zentrum fuer Informatik, 2016.

18 Mihai Patrascu and Ryan Williams. On the possibility of faster SAT algorithms. In Moses Charikar, editor, *Proceedings of the Twenty-First Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2010, Austin, Texas, USA, January 17-19, 2010*, pages 1065–1075. SIAM, 2010. doi:10.1137/1.9781611973075.86.

19 Krzysztof Pietrzak. On the parameterized complexity of the fixed alphabet shortest common supersequence and longest common subsequence problems. *J. Comput. Syst. Sci.*, 67(4):757–771, 2003. doi:10.1016/S0022-0000(03)00078-3.

20 Marcin Pilipczuk, Michal Pilipczuk, Piotr Sankowski, and Erik Jan van Leeuwen. Network sparsification for steiner problems on planar and bounded-genus graphs. In *55th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2014, Philadelphia, PA, USA, October 18-21, 2014*, pages 276–285. IEEE Computer Society, 2014. URL: http://ieeexplore.ieee.org/xpl/mostRecentIssue.jsp?punumber=6975722, doi:10.1109/FOCS.2014.37.

21 Michal Pilipczuk and Marcin Wrochna. On space efficiency of algorithms working on structural decompositions of graphs. In Ollinger and Vollmer [17], pages 57:1–57:15. doi:10.4230/LIPIcs.STACS.2016.57.

22 Ilya P. Razenshteyn, Zhao Song, and David P. Woodruff. Weighted low rank approximations with provable guarantees. In Daniel Wichs and Yishay Mansour, editors, *Proceedings of the 48th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2016, Cambridge, MA, USA, June 18-21, 2016*, pages 250–263. ACM, 2016. URL: http://dl.acm.org/citation.cfm?id=2897518, doi:10.1145/2897518.2897639.

23 Bruce Reed, Kaleigh Smith, and Adrian Vetta. Finding odd cycle transversals. *Operations Research Letters*, 32(4):299–301, 2004.

Participants

- Amir Abboud
Stanford University, US
- Arturs Backurs
MIT – Cambridge, US
- Andreas Björklund
Lund University, SE
- Édouard Bonnet
Middlesex University, GB
- Cornelius Brand
Universität des Saarlandes, DE
- Karl Bringmann
MPI für Informatik – Saarbrücken, DE
- Yixin Cao
Hong Kong Polytechnic University, CN
- Radu Curticapean
Hungarian Academy of Sciences – Budapest, HU
- Marek Cygan
University of Warsaw, PL
- Holger Dell
Universität des Saarlandes, DE
- Fedor V. Fomin
University of Bergen, NO
- Tobias Friedrich
Hasso-Plattner-Institut – Potsdam, DE
- Petr A. Golovach
University of Bergen, NO
- Gregory Z. Gutin
Royal Holloway University of London, GB
- Danny Hermelin
Ben Gurion University – Beer Sheva, IL
- Petr Hlineny
Masaryk University – Brno, CZ
- Petteri Kaski
Aalto University, FI
- Eun Jung Kim
University Paris-Dauphine, FR
- Yiannis Koutis
University of Puerto Rico – Rio Piedras, PR
- Łukasz Kowalik
University of Warsaw, PL
- Stefan Kratsch
Universität Bonn, DE
- Bingkai Lin
National Institute of Informatics – Tokyo, JP
- Daniel Lokshtanov
University of Bergen, NO
- Dániel Marx
Hungarian Academy of Sciences – Budapest, HU
- Jesper Nederlof
TU Eindhoven, NL
- Fahad Panolan
University of Bergen, NO
- Christophe Paul
CNRS – Montpellier, FR
- Geevarghese Philip
Chennai Mathematical Institute, IN
- Marcin Pilipczuk
University of Warsaw, PL
- Michal Pilipczuk
University of Warsaw, PL
- M. S. Ramanujan
TU Wien, AT
- Peter Rossmanith
RWTH Aachen, DE
- Marc Roth
Universität des Saarlandes, DE
- Saket Saurabh
The Institute of Mathematical Sciences, India, IN
- Ildiko Schlotter
Budapest University of Technology & Economics, HU
- Stefan Szeider
TU Wien, AT
- Dimitrios M. Thilikos
University of Athens, GR
- Magnus Wahlström
Royal Holloway University of London, GB
- Gerhard J. Woeginger
RWTH Aachen, DE
- David P. Woodruff
IBM Almaden Center – San Jose, US
- Meirav Zehavi
University of Bergen, NO