DAGSTUHL
REPORTS

Volume 14, Issue 10, October 2024

Dagstuhl Reports, Vol. 14, Issue 10 ISSN 2192-5283

ISSN 2192-5283

Published online and open access by
Schloss Dagstuhl – Leibniz-Zentrum für Informatik GmbH, Dagstuhl Publishing, Saarbrücken/Wadern, Germany. Online available at
https://www.dagstuhl.de/dagpub/2192-5283

Publication date
April, 2025

Bibliographic information published by the Deutsche Nationalbibliothek
The Deutsche Nationalbibliothek lists this publication in the Deutsche Nationalbibliografie; detailed bibliographic data are available in the Internet at `https://dnb.d-nb.de`.

License
This work is licensed under a Creative Commons Attribution 4.0 International license (CC BY 4.0).
In brief, this license authorizes each and everybody to share (to copy, distribute and transmit) the work under the following conditions, without impairing or restricting the authors' moral rights:
- Attribution: The work must be attributed to its authors.

The copyright is retained by the corresponding authors.

Digital Object Identifier: 10.4230/DagRep.14.10.i

Aims and Scope
The periodical *Dagstuhl Reports* documents the program and the results of Dagstuhl Seminars and Dagstuhl Perspectives Workshops.
In principal, for each Dagstuhl Seminar or Dagstuhl Perspectives Workshop a report is published that contains the following:
- an executive summary of the seminar program and the fundamental results,
- an overview of the talks given during the seminar (summarized as talk abstracts), and
- summaries from working groups (if applicable).

This basic framework can be extended by suitable contributions that are related to the program of the seminar, e. g. summaries from panel discussions or open problem sessions.

Editorial Board
- Elisabeth André
- Franz Baader
- Goetz Graefe
- Reiner Hähnle
- Barbara Hammer
- Lynda Hardman
- Steve Kremer
- Rupak Majumdar
- Heiko Mantel
- Lennart Martens
- Albrecht Schmidt
- Wolfgang Schröder-Preikschat
- Raimund Seidel (*Editor-in-Chief*)
- Heike Wehrheim
- Verena Wolf
- Martina Zitterbart

Editorial Office
Michael Wagner *(Managing Editor)*
Michael Didas *(Managing Editor)*
Jutka Gasiorowski *(Editorial Assistance)*
Dagmar Glaser *(Editorial Assistance)*
Thomas Schillo *(Technical Assistance)*

Contact
Schloss Dagstuhl – Leibniz-Zentrum für Informatik
Dagstuhl Reports, Editorial Office
Oktavie-Allee, 66687 Wadern, Germany
reports@dagstuhl.de
https://www.dagstuhl.de/dagrep

Report from Dagstuhl Seminar 24411

New Tools in Parameterized Complexity: Paths, Cuts, and Decomposition

Fedor V. Fomin[*1], **Dániel Marx**[*2], **Saket Saurabh**[*3], **Roohani Sharma**[*4], **and Madhumita Kundu**[†5]

1 **University of Bergen, NO.** fomin@ii.uib.no
2 **CISPA – Saarbrücken, DE.** marx@cispa.de
3 **The Institute of Mathematical Sciences – Chennai, IN.** saket@imsc.res.in
4 **MPI für Informatik – Saarbrücken, DE.** roohani.sharma90@gmail.com
5 **University of Bergen, NO.** Madhumita.Kundu@uib.no

Abstract

The Dagstuhl Seminar concentrated on the development of new tools arising from the parameterized complexity of cuts, paths, and decompositions in graphs. The last 2 years were very exciting for the area, with a number of breakthroughs.

In FOCS 2021, Korhonen introduced a new method for approximating tree decompositions in graphs. His method, which was deeply rooted in classical graph theory, appeared to be a very handy tool for decomposing graphs, and several STOC/FOCS papers developed this method in various settings. In parallel, a novel perspective on graph decompositions was proposed by Bonnet *et al.* in FOCS 2020. The new theory of twin-width had many exciting consequences, and we were still at the beginning of understanding the real impact of the new decompositions on graph algorithms.

In a series of papers (SODA 2021, STOC 2022, SODA 2023), Kim *et al.* developed beautiful algorithmic methods for handling separators in (undirected, weighted, or directed) graphs by the addition of arcs. The new algorithmic tool was used to resolve a number of long-standing open problems in the area, and it also seemed to pave the road to many more new discoveries.

Reis and Rothvoss (Arxiv 2023) announced a $((\log n)^{O(n)})$ time randomized algorithm to solve integer programs in n variables. This breakthrough had an impact on many problems in parameterized complexity, especially on problems concerning cuts in graphs. Finally, by employing algebraic methods (both new and old), significant progress was made on several problems related to paths, including the classical (k)-disjoint path problems.

This seminar brought together people from the parameterized complexity community, specialists in cuts, flows, and connectivity, and those who had been at the forefront of these new developments. In doing so, it consolidated the results achieved in recent years, discussed future research directions, and further explored the potential applications of the methods and techniques described above.

Seminar October 6–11, 2024 – https://www.dagstuhl.de/24411
2012 ACM Subject Classification Theory of computation → Computational complexity and cryptography; Theory of computation → Design and analysis of algorithms
Keywords and phrases fixed-parameter tractability, intractability, parameterized complexity
Digital Object Identifier 10.4230/DagRep.14.10.1

* Editor / Organizer
† Editorial Assistant / Collector

Except where otherwise noted, content of this report is licensed under a Creative Commons BY 4.0 International license
New Tools in Parameterized Complexity: Paths, Cuts, and Decomposition, *Dagstuhl Reports*, Vol. 14, Issue 10, pp. 1–21
Editors: Fedor V. Fomin, Dániel Marx, Saket Saurabh, and Roohani Sharma
Dagstuhl Reports
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

1 Executive Summary

Fedor V. Fomin (University of Bergen, NO)
Dániel Marx (CISPA – Saarbrücken, DE)
Saket Saurabh (The Institute of Mathematical Sciences – Chennai, IN)
Roohani Sharma (MPI für Informatik – Saarbrücken, DE)

License Creative Commons BY 4.0 International license
© Fedor V. Fomin, Dániel Marx, Saket Saurabh, and Roohani Sharma

Description of the Seminar: Topics and goals

Parameterized Complexity is an alternative approach of handling computational intractability. The main idea of the approach taken by the Parameterized Complexity is to analyze the complexity in finer detail by considering additional problem parameters beyond the input size and expresses the efficiency of the algorithms in terms of these parameters. In this framework, many NP-hard problems have been shown to be (fixed-parameter) tractable (FPT) when certain structural parameters of the inputs are bounded.

In the last three decades, there has been tremendous progress in understanding which problems are fixed-parameter tractable and which problems are not (under standard complexity assumptions). For all these years the central vision of the Parameterized Complexity has been to provide the algorithmic and complexity-theoretic toolkit for studying multivariate algorithmics in different disciplines and subfields of Computer Science. To achieve this vision several algorithmic and complexity theoretic tools such as *polynomial time preprocessing, aka, kernelization*, *color-coding*, *graph-decompositions*, *parameterized integer programming*, *iterative compression*, or *lower bounds methods based on assumptions stronger than* P≠NP have been developed. These tools are *universal* as they not only helped in the development of the core of Parameterized Complexity but also led to its success in other subfields of Computer Science such as Approximation Algorithms, Computational Social Choice, Computational Geometry, problems solvable in P (polynomial time) to name a few.

In the last few years several decade old open problems in Parameterized Complexity have been resolved. These have resulted in several new algorithmic tools for the core of Parameterized Complexity. These include tools such as iterative and local improvement methods for graph decomposition, methods arising from extremal combinatorics and graph theory, flow augmentation, faster algorithms for solving integer programs on n variables, and new algebraic methods. A natural question is to extend these tools in different directions and explore the limits and applicability of these new tools. Thus,

> *the main objective of the Dagstuhl Seminar was to initiate the discussion on extension, limits and applicability of newly developed tools arising from paths, cuts, and decomposition.*

One of the seminar's central goals was to facilitate a fruitful dialogue between researchers working at the core of Parameterized Complexity and those from Mathematical Programming, Computational Linear Algebra, Graph Theory, and Combinatorics, who had contributed to recent advances in parameterized algorithms. The Dagstuhl event enabled participants to explore possibilities for developing new tools and techniques that emerged from this collaboration.

Next, the seminar presented a few concrete examples of newly developed tools and techniques from various domains of Parameterized Complexity, which formed the focal points of discussion.

- ***Width Parameters: Treewidth and Twinwidth.***
- ***Tools Based on Extremal Combinatorics: Paths and Rainbow Matching.***
- ***Cut Based Tool: Flow Augmentation.***
- ***Mathematical Programming.***
- ***Algebraic Methods.***

Related Seminars

This Dagstuhl Seminar could be considered as a continuation of the Dagstuhl Seminar series on parameterized algorithms and complexity. The previous seminars in this series are New Horizons in Parameterized Complexity (seminar 19041), Randomization in Parameterized Complexity (seminar 17041), Optimality and Tight Results in Parameterized Complexity (seminar 14451), Data Reduction and Problem Kernels (seminar 12241), Parameterized complexity and approximation algorithms (seminar 09511), Structure Theory and FPT Algorithmics for Graphs, Digraphs and Hypergraphs (seminar 07281), Exact Algorithms and Fixed-Parameter Tractability (seminar 05301), Fixed Parameter Algorithms (seminar 03311), and Parameterized Complexity (seminar 01311).

Organization of the Seminar

During the five-day seminar, 48 researchers from theoretical computer science, mathematical optimization, and operations research convened. Attendees ranged from senior scientists to postdoctoral scholars and advanced doctoral candidates, creating a rich environment for both mentorship and innovation.

The seminar featured 22 presentations of varying lengths. Six keynote speakers–Dániel Marx, Michał Pilipczuk, Euiwoong Lee, Tuukka Korhonen, Jie Xue, and Daniel Lokshtanov–delivered 60-minute talks that provided overviews of state-of-the-art methods and showcased recent breakthroughs in their respective areas. The remaining slots were filled with shorter, 30-minute talks covering various topics.

At the beginning of the week, open problem sessions encouraged participants to share challenges and spark collaborative research. The schedule also included ample free time, which attendees used for productive discussions and joint work, fostering new ideas and potential research partnerships.

Outcome

Organizers and participants regarded the seminar as a great success. It successfully brought together the relevant research communities, facilitated the sharing of state-of-the-art results, and enabled discussions on the major challenges in the field. The talks were not only excellent but also highly stimulating, prompting active engagement in working groups during afternoons and evenings. Particularly noteworthy was the participation of younger researchers (postdocs and PhD students), who integrated seamlessly and contributed to the seminar's collegial and productive atmosphere.

2 Table of Contents

3 Overview of Talks

3.1 Approximating Small Sparse Cuts

Aditya Anand (University of Michigan – Ann Arbor, US), Thatchaphol Saranurak (University of Michigan – Ann Arbor, US)

License Creative Commons BY 4.0 International license
© Aditya Anand and Thatchaphol Saranurak
Joint work of Aditya Anand, Jason Li, Thatchaphol Saranurak
Main reference Aditya Anand, Euiwoong Lee, Jason Li, Thatchaphol Saranurak: "Approximating Small Sparse Cuts", CoRR, Vol. abs/2403.08983, 2024.
URL https://doi.org/10.48550/ARXIV.2403.08983

We study polynomial-time approximation algorithms for (edge/vertex) Sparsest Cut and Small Set Expansion in terms of k, the number of edges or vertices cut in the optimal solution. Our main results are $O(poly \log k)$-approximation algorithms for various versions in this setting.

Our techniques involve an extension of the notion of sample sets (Feige and Mahdian STOC'06), originally developed for small balanced cuts, to sparse cuts in general. We then show how to combine this notion of sample sets with two algorithms, one based on an existing framework of LP rounding and another new algorithm based on the cut-matching game, to get such approximation algorithms. Our cut-matching game algorithm can be viewed as a local version of the cut-matching game by Khandekar, Khot, Orecchia and Vishnoi and certifies an expansion of every vertex set of size s in $O(\log s)$ rounds. These techniques may be of independent interest.

As corollaries of our results, we also obtain an O(logopt)-approximation for min-max graph partitioning, where opt is the min-max value of the optimal cut, and improve the bound on the size of multicut mimicking networks computable in polynomial time.

3.2 Unbreakable Decomposition in Close-to-Linear Time

Aditya Anand (University of Michigan – Ann Arbor, US), Euiwoong Lee (University of Michigan – Ann Arbor, US), Thatchaphol Saranurak (University of Michigan – Ann Arbor, US)

License Creative Commons BY 4.0 International license
© Aditya Anand, Euiwoong Lee, and Thatchaphol Saranurak
Joint work of Aditya Anand, Euiwoong Lee, Jason Li, Yaowei Long, Thatchaphol Saranurak
Main reference Aditya Anand, Euiwoong Lee, Jason Li, Yaowei Long, Thatchaphol Saranurak: "Unbreakable Decomposition in Close-to-Linear Time", in Proc. of the 2025 Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2025, New Orleans, LA, USA, January 12-15, 2025, pp. 1464–1493, SIAM, 2025.
URL https://doi.org/10.1137/1.9781611978322.46

Unbreakable decomposition, introduced by Cygan et al. (SICOMP'19) and Cygan et al. (TALG'20), has proven to be one of the most powerful tools for parameterized graph cut problems in recent years. Unfortunately, all known constructions require at least $\Omega_k(mn^2)$ time, given an undirected graph with n vertices, m edges, and cut-size parameter k. In this work, we show the first close-to-linear time parameterized algorithm that computes an unbreakable decomposition. More precisely, for any $0 < \epsilon \leq 1$, our algorithm runs in time $2^{O(\frac{k}{\epsilon}\log(k/\epsilon))}m^{1+\epsilon}$ and computes a $(O(k/\epsilon), k)$ unbreakable tree decomposition of the input graph, where each bag has adhesion at most $O(k/\epsilon)$. This immediately opens up possibilities for obtaining close-to-linear time algorithms for numerous problems whose only known solution is based on unbreakable decomposition.

3.3 Planar Min-Sum Disjoint Paths in Subexponential FPT time

Matthias Bentert (University of Bergen, NO), Fedor V. Fomin (University of Bergen, NO), Petr A. Golovach (University of Bergen, NO)

License Creative Commons BY 4.0 International license
© Matthias Bentert, Fedor V. Fomin, and Petr A. Golovach

In the Min-Sum Disjoint Paths problem, we are given an edge-weighted n-vertex graph and k terminal pairs. The task is to connect all terminal pairs by pairwise internally vertex-disjoint paths of minimum total length (or decide that there is no set of pairwise disjoint paths). We show that Min-Sum Disjoint Paths on (directed or undirected) planar input graphs can be solved in $2^{O(\sqrt{\ell}\log^{O(1)}(\ell))}n^{O(1)}$ time, where ℓ is the number of edges in an optimal solution. We complement our main result with an ETH-based lower bound excluding $2^{o(\sqrt{n})}$-time algorithms for undirected and unweighted planar graphs even in the special case where we ask whether all terminal pairs can be connected by shortest paths between the respective endpoints.

3.4 Twin-width – Part 2

Édouard Bonnet (ENS – Lyon, FR)

License Creative Commons BY 4.0 International license
© Édouard Bonnet

We survey some algorithmic applications of twin-width.

3.5 Algorithms for 2-connected network design and flexible Steiner trees with a constant number of terminals

Joseph Cheriyan (University of Waterloo, CA)

License Creative Commons BY 4.0 International license
© Joseph Cheriyan
Joint work of Joseph Cheriyan, Ishan Bansal, Logan Grout, Sharat Ibrahimpur
Main reference Ishan Bansal, Joe Cheriyan, Logan Grout, Sharat Ibrahimpur: "Algorithms for 2-Connected Network Design and Flexible Steiner Trees with a Constant Number of Terminals", in Proc. of the Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques, APPROX/RANDOM 2023, September 11-13, 2023, Atlanta, Georgia, USA, LIPIcs, Vol. 275, pp. 14:1–14:14, Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2023.
URL https://doi.org/10.4230/LIPICS.APPROX/RANDOM.2023.14

The k-Steiner-2NCS problem is as follows: Given a constant k, and an undirected connected graph $G = (V, E)$, non-negative costs c on the edges, and a partition of V into a set of terminals, T, and a set of non-terminals (or, Steiner nodes), $V - T$, where $|T| = k$, our algorithmic goal is to find a min-cost two-node connected subgraph that contains the terminals.

We present a randomized polynomial-time algorithm for the unweighted problem, and a randomized FPTAS for the weighted problem.

We obtain similar results for the k-Steiner-2ECS problem, where the input is the same, and the algorithmic goal is to find a min-cost two-edge connected subgraph that contains the terminals.

Our methods build on results by Bjorklund, Husfeldt, and Taslaman (SODA 2012) that give a randomized polynomial-time algorithm for the unweighted k-Steiner-cycle problem; this problem has the same inputs as the unweighted k-Steiner-2NCS problem, and the algorithmic goal is to find a min-cost simple cycle that contains the terminals.

3.6 Topological methods for graph algorithms: (multi-)cuts on surface-embedded graphs

Éric Colin de Verdière (Gustave Eiffel University – Marne-la-Vallée, FR)

License Creative Commons BY 4.0 International license
© Éric Colin de Verdière

Minimum cut and multicut problems for general graphs are well-studied. In this talk, we survey algorithms and lower bounds for these problems when restricting to graphs that are either planar or embeddable on a fixed surface. For such classes of graphs, topological techniques, developed originally in computational geometry/topology with very different motivations in mind, turn out to be very useful.

We first aim to provide an executive summary of the computational topology routines that are the most useful for cut problems for surface-embedded graphs. We then discuss near-linear time algorithms for minimum cut on a fixed surface, which outperform the existing generic (polynomial-time) combinatorial algorithms. We then turn to the multicut problem, presenting an algorithm that is fixed-parameter tractable parameterized by the genus and the number of terminals, and an almost matching lower bound. There is actually a refined complexity analysis depending also on the demand pattern. Finally, we present an algorithm for the multicut problem that returns an $(1+\varepsilon)$-approximation and that is fixed parameter tractable in the genus, the number of terminals, and ε.

All these results rely on topological methods: The subgraph of the dual of the input graph, made of the edges dual to a multicut, has nice properties, which can be exploited using classical tools from algebraic topology such as homotopy, homology, and covering spaces.

3.7 Efficient Approximation of Fractional Hypertree Width

Daniel Lokshtanov (University of California – Santa Barbara, US), Saket Saurabh (The Institute of Mathematical Sciences – Chennai, IN), Vaishali Surianarayanan (University of California – Santa Barbara, US), Jie Xue (New York University – Shanghai, CN)

License Creative Commons BY 4.0 International license
© Daniel Lokshtanov, Saket Saurabh, Vaishali Surianarayanan, and Jie Xue
Joint work of Daniel Lokshtanov, Viktoriia Korchemna, Saket Saurabh, Vaishali Surianarayanan, Jie Xue
Main reference Viktoriia Korchemna, Daniel Lokshtanov, Saket Saurabh, Vaishali Surianarayanan, Jie Xue: "Efficient Approximation of Fractional Hypertree Width", CoRR, Vol. abs/2409.20172, 2024.
URL https://doi.org/10.48550/ARXIV.2409.20172

We give two new approximation algorithms to compute the *fractional hypertree width* of an input hypergraph. The first algorithm takes as input n-vertex m-edge hypergraph H of fractional hypertree width at most ω, runs in polynomial time and produces a tree decomposition of H of fractional hypertree width $\mathcal{O}(\omega \log n \log \omega)$, i.e., it is an $\mathcal{O}(\log n \log \omega)$-approximation algorithm. As an immediate corollary this yields polynomial time $\mathcal{O}(\log^2 n \log \omega)$-approximation algorithms for (generalized) hypertree width as well. To the best of our knowledge our algorithm is the first non-trivial polynomial-time approximation algorithm for fractional hypertree width and (generalized) hypertree width, as opposed to algorithms that run in polynomial time only when ω is considered a constant. For hypergraphs with the *bounded intersection property* (i.e. hypergraphs where every pair of hyperedges have at most η vertices in common) the algorithm outputs a hypertree decomposition with fractional hypertree width $\mathcal{O}(\eta\omega^2 \log \omega)$ and generalized hypertree width $\mathcal{O}(\eta\omega^2 \log \omega(\log \eta + \log \omega))$. This ratio is comparable with the recent algorithm of Lanzinger and Razgon [STACS 2024], which produces a hypertree decomposition with generalized hypertree width $\mathcal{O}(\omega^2(\omega + \eta))$, but uses time (at least) exponential in η and ω.

The second algorithm runs in time $n^\omega m^{\mathcal{O}(1)}$ and produces a tree decomposition of H of fractional hypertree width $\mathcal{O}(\omega \log^2 \omega)$. This significantly improves over the $(n+m)^{\mathcal{O}(\omega^3)}$ time algorithm of Marx [ACM TALG 2010], which produces a tree decomposition of fractional hypertree width $\mathcal{O}(\omega^3)$, both in terms of running time and the approximation ratio.

Our main technical contribution, and the key insight behind both algorithms, is a variant of the classic Menger's Theorem for clique separators in graphs: For every graph G, vertex sets A and B, family $\mathcal{F}$ of cliques in G, and positive rational f, either there exists a sub-family of $\mathcal{O}(f \cdot \log^2 n)$ cliques in $\mathcal{F}$ whose union separates A from B, or there exist $f \cdot \log |\mathcal{F}|$ paths from A to B such that no clique in $\mathcal{F}$ intersects more than $\log |\mathcal{F}|$ paths.

3.8 Twin-width I

Eun Jung Kim (KAIST – Daejeon, KR & CNRS – Paris, FR)

License 🅭 Creative Commons BY 4.0 International license
© Eun Jung Kim

Twin-width is a relatively new notion introduced in 2020 by Bonnet, Kim, Thomassé and Watrigant. Since then, it is prove to demonstrate rich structure and be a useful tool for understanding graph classes and logic, designing algorithms, etc. In this talk, we present the notion, grid theorem for twin-width and how to use it for proving certain graph classes have bounded twin-width as well as other use cases of twin-width.

3.9 Longest Path parameterized by Maximum Independent Set

Fedor V. Fomin (University of Bergen, NO)

License 🅭 Creative Commons BY 4.0 International license
© Fedor V. Fomin

Question 1. What is the complexity of LONGESTED PATH parameterized by $\alpha(G)$ (the largest size of an indpendent set). We know that the problem is **FPT** in undirected graphs and either returns a path of length at least k or establishes that $\alpha(G) \geq k$. However, what is the complexity of the problem in directed graphs? Is it even in **XP**?

Additionally, what about the **promise version**, where an independent set of size $\alpha(G)$ is provided as input?

Question 2. (Gallai-Milgram Theorem) Let G be a directed graph. The vertices of G can be covered by at most $\alpha(G)$ disjoint paths.

Additionally, there exist at least $k - t$ edge-disjoint paths if the graph satisfies certain conditions.

3.10 Steiner Tree Parameterized by Multiway Cut and Even Less

Bart Jansen (TU Eindhoven, NL)

License Creative Commons BY 4.0 International license
© Bart Jansen
Joint work of Bart M. P. Jansen, Celine Swennenhuis
Main reference Bart M. P. Jansen, Céline M. F. Swennenhuis: "Steiner Tree Parameterized by Multiway Cut and Even Less", in Proc. of the 32nd Annual European Symposium on Algorithms, ESA 2024, September 2-4, 2024, Royal Holloway, London, United Kingdom, LIPIcs, Vol. 308, pp. 76:1–76:16, Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2024.
URL https://doi.org/10.4230/LIPICS.ESA.2024.76

In the Steiner Tree problem we are given an undirected edge-weighted graph as input, along with a set K of vertices called *terminals*. The task is to output a minimum-weight connected subgraph that spans all the terminals. The famous Dreyfus-Wagner algorithm running in $3^{|K|}\mathsf{poly}(n)$ time shows that the problem is fixed-parameter tractable parameterized by the number of terminals. We present fixed-parameter tractable algorithms for Steiner Tree using structurally smaller parameterizations.

Our first result concerns the parameterization by a multiway cut S of the terminals, which is a vertex set S (possibly containing terminals) such that each connected component of $G - S$ contains at most one terminal. We show that Steiner Tree can be solved in $2^{O(|S| \log |S|)}\mathsf{poly}(n)$ time and polynomial space, where S is a minimum multiway cut for K. The algorithm is based on the insight that, after guessing how an optimal Steiner tree interacts with a multiway cut S, computing a minimum-cost solution of this type can be formulated as minimum-cost bipartite matching.

Our second result concerns a new hybrid parameterization called *K-free treewidth* that simultaneously refines the number of terminals $|K|$ and the treewidth of the input graph. By utilizing recent work on $\mathcal{H}$-Treewidth in order to find a corresponding decomposition of the graph, we give an algorithm that solves Steiner Tree in time $2^{O(k)}\mathsf{poly}(n)$, where k denotes the K-free treewidth of the input graph. To obtain this running time, we show how the *rank-based* approach for solving Steiner Tree parameterized by treewidth can be extended to work in the setting of K-free treewidth, by exploiting existing algorithms parameterized by $|K|$ to compute the table entries of leaf bags of a tree K-free decomposition.

3.11 A Subexponential Time Algorithm for Makespan Scheduling of Unit Jobs with Precedence Constraints

Jesper Nederlof (Utrecht University, NL)

License Creative Commons BY 4.0 International license
© Jesper Nederlof
Joint work of Jesper Nederlof, Céline M. F. Swennenhuis, Karol Wegrzycki
Main reference Jesper Nederlof, Céline M. F. Swennenhuis, Karol Wegrzycki: "A Subexponential Time Algorithm for Makespan Scheduling of Unit Jobs with Precedence Constraints", CoRR, Vol. abs/2312.03495, 2023.
URL https://doi.org/10.48550/ARXIV.2312.03495

In a classical scheduling problem, we are given a set of n jobs of unit length with precedence constraints, and the goal is to find a schedule of these jobs on m identical machines that minimizes the makespan. In standard 3-field notation, it is denoted as $Pm|\text{prec}, p_j = 1|C_{\max}$.

For $m = 2$ machines, the problem can be solved in polynomial time. Settling the complexity for any constant $m \geq 3$ is a longstanding open question in the field, asked by Lenstra and Rinnooy Kan [OR 1978] in the late 70s and prominently featured in the textbook of Garey and Johnson. Since then, the problem has been thoroughly investigated, but nontrivial solutions had been found only in special cases or relaxed settings. For example, despite the possibility of the problem being polynomially solvable in the exact setting, just the existence of an approximation-scheme is widely regarded as a major open problem (see the survey of Bansal [MAPS 2017]), but so far, only superpolynomial approximations are known.

In this paper, we make the first progress on the exact complexity of $Pm|\text{prec}, p_j = 1|C_{\max}$. We present an algorithm that runs in $2^{O(\sqrt{n}\log n)}$ time for $m = O(1)$. Before our work, only a $2^{O(n)}$ time exact algorithm was known by Held and Karp [ACM 1961].

3.12 The Parameter Report: An Orientation Guide for Data-Driven Parameterization

Christian Komusiewicz (Friedrich-Schiller-Universität Jena, DE)

License Ⓒ Creative Commons BY 4.0 International license
© Christian Komusiewicz
Joint work of Christian Komusiewicz, Nils Morawietz, Frank Sommer, Luca Staus

A strength of parameterized algorithmics is that each problem can be parameterized by an essentially inexhaustible set of parameters. Usually, the choice of the considered parameter is informed by the theoretical relations between parameters with the general goal of achieving FPT-algorithms for smaller and smaller parameters. However, the FPT-algorithms for smaller parameters usually have higher running times and it is not clear whether the decrease in the parameter value or the increase in the running time bound dominates in real-world data. Any answer to this question requires knowledge on typical parameter values. To provide a data-driven guideline for parameterized complexity studies of graph problems, we present the first comprehensive comparison of parameter values for a set of benchmark graphs originating from real-world applications. Our study covers degree-related parameters, such as maximum degree or degeneracy, neighborhood-based parameters such as neighborhood diversity and modular-width, modulator-based parameters such as vertex cover number and feedback vertex set number, and the treewidth of the graphs. Our implementation and full experimental data are openly available.

3.13 Linear-Time Algorithms for k-Edge-Connected Components, k-Lean Tree Decompositions, and More

Tuukka Korhonen (University of Copenhagen, DK)

License Ⓒ Creative Commons BY 4.0 International license
© Tuukka Korhonen

We present $k^{O(k^2)}m$ time algorithms for various problems about decomposing a given graph by edge cuts or vertex separators of size less than k into parts that are "well-connected" with respect to cuts or separators of size less than k; here, m is the total number of vertices and edges of the graph. As an application of our results, we obtain for every fixed k a linear-time algorithm for computing the k-edge-connected components of a given graph, solving a long-standing open problem. More generally, we obtain a $k^{O(k^2)}m$ time algorithm for computing a k-Gomory-Hu tree of a given graph, which is a structure representing all pairwise minimum cuts of size less than k.

Our main technical result, from which the other results follow, is a $k^{O(k^2)}m$ time algorithm for computing a k-lean tree decomposition of a given graph. This is a tree decomposition with adhesion size less than k that captures the existence of separators of size less than k between subsets of its bags. A k-lean tree decomposition is also an unbreakable tree decomposition with optimal unbreakability parameters for the adhesion size bound k.

As further applications, we obtain $k^{O(k^2)}m$ time algorithms for k-vertex connectivity and for k-Gomory-Hu tree for element connectivity. All of our algorithms are deterministic.

Our techniques are inspired by the tenth paper of the Graph Minors series of Robertson and Seymour and by Bodlaender's parameterized linear-time algorithm for computing treewidth.

3.14 Parameterized Inapproximability Hypothesis under Exponential Time Hypothesis

Bingkai Lin (Nanjing University, CN)

License Creative Commons BY 4.0 International license
© Bingkai Lin
Joint work of Bingkai Lin, Venkatesan Guruswami, Xuandi Ren, Yican Sun, Kewen Wu
Main reference Venkatesan Guruswami, Bingkai Lin, Xuandi Ren, Yican Sun, Kewen Wu: "Parameterized Inapproximability Hypothesis under Exponential Time Hypothesis", in Proc. of the 56th Annual ACM Symposium on Theory of Computing, STOC 2024, Vancouver, BC, Canada, June 24-28, 2024, pp. 24–35, ACM, 2024.
URL https://doi.org/10.1145/3618260.3649771

The Parameterized Inapproximability Hypothesis (PIH) asserts that no fixed parameter tractable (FPT) algorithm can distinguish a satisfiable CSP instance, parameterized by the number of variables, from one where every assignment fails to satisfy an ε fraction of constraints for some absolute constant $\varepsilon > 0$. PIH plays the role of the PCP theorem in parameterized complexity. However, PIH has only been established under Gap-ETH, a very strong assumption with an inherent gap.

In this work, we prove PIH under the Exponential Time Hypothesis (ETH). This is the first proof of PIH from a gap-free assumption. Our proof is self-contained and elementary. We identify an ETH-hard CSP whose variables take vector values, and constraints are either linear or of a special parallel structure. Both kinds of constraints can be checked with constant soundness via a "parallel PCP of proximity" based on the Walsh-Hadamard code.

3.15 Feedback Vertex Set on Planar Graphs

Daniel Lokshtanov (University of California – Santa Barbara, US)

License Creative Commons BY 4.0 International license
© Daniel Lokshtanov

Given an undirected graph $G = (V, E)$, the **Feedback Vertex Set (FVS)** problem asks for a minimum-size subset $S \subseteq V$ such that the subgraph induced by $V \setminus S$ is acyclic. The unweighted FEEDBACK VERTEX SET problem is known to admit an EPTAS on planar graphs, whereas the weighted version only has a PTAS. A natural question is whether the problem admits an EPTAS even in the weighted case.

3.16 Cuts, Paths, and Decompositions

Dániel Marx (CISPA – Saarbrücken, DE)

License Creative Commons BY 4.0 International license
© Dániel Marx

We overview the many different types of cut, path, and decomposition problems in the field of parameterized algorithms and sketch some connections between the different topics.

3.17 Minimum Isolating Cuts for Fast Graph Algorithms: A tutorial

Thatchaphol Saranurak (University of Michigan – Ann Arbor, US)

License Creative Commons BY 4.0 International license
© Thatchaphol Saranurak

I give a gentle tutorial on using the minimum isolating cuts for fast graph algorithms, as well as, give a survey of its applications.

3.18 Parameterized Streaming

Ramanujan Sridharan (University of Warwick – Coventry, GB)

License Creative Commons BY 4.0 International license
© Ramanujan Sridharan

Joint work of Ramanujan Sridharan, Daniel Lokshtanov, Pranabendu Misra, Fahad Panolan, M. S. Ramanujan, Saket Saurabh, Meirav Zehavi
Main reference Daniel Lokshtanov, Pranabendu Misra, Fahad Panolan, M. S. Ramanujan, Saket Saurabh, Meirav Zehavi: "Meta-theorems for Parameterized Streaming Algorithms‡", in Proc. of the 2024 ACM-SIAM Symposium on Discrete Algorithms, SODA 2024, Alexandria, VA, USA, January 7-10, 2024, pp. 712–739, SIAM, 2024.
URL https://doi.org/10.1137/1.9781611977912.28

The streaming model has been studied from the point of parameterized complexity in the last decade by several researchers. However, the applicability of the streaming model to central problems in parameterized complexity has remained somewhat limited due to simple $\Omega(n)$-space lower bounds for many problems. In other words, the $k^{O(1)}(\log n)^{O(1)}$ space requirement of the parameterized streaming model is too restrictive. This has motivated the study of parameterized semi-streaming algorithms.

In this talk, we will discuss some recent developments in this direction.

3.19 Planar Disjoint Shortest Paths is Fixed-Parameter Tractable

Giannos Stamoulis (University of Warsaw, PL)

License Creative Commons BY 4.0 International license
© Giannos Stamoulis

Joint work of Michał Pilipczuk, Giannos Stamoulis, Michał Włodarczyk

In the Disjoint Shortest Paths problem one is given a graph G and a set $T = \{(s1_,t_1), ..., (s_k, t_k)\}$ of k vertex pairs. The question is whether there exist vertex-disjoint paths $P_1, ..., P_k$ in G so that each P_i is a shortest path between s_i and t_i. While the problem is known to be W[1]-hard in general, we show that it is fixed-parameter tractable on planar graphs with positive edge weights. Specifically, we propose an algorithm for Planar Disjoint Shortest Paths with running time $2^{O(k \log k)} n^{O(1)}$. Notably, our parameter dependency is better than state-of-the-art $2^{O(k^2)}$ for the Planar Disjoint Paths problem, where the sought paths are not required to be shortest paths.

3.20 Parameterized Approximation for Capacitated d-Hitting Set with Hard Capacities

Vaishali Surianarayanan (University of California – Santa Barbara, US), Daniel Lokshtanov (University of California – Santa Barbara, US), Saket Saurabh (The Institute of Mathematical Sciences – Chennai, IN), Jie Xue (New York University – Shanghai, CN)

License Creative Commons BY 4.0 International license
© Vaishali Surianarayanan, Daniel Lokshtanov, Saket Saurabh, and Jie Xue
Joint work of Vaishali Surianarayanan, Abhishek Sahu, Daniel Lokshtanov, Saket Saurabh, Jie Xue

In the Capacitated d-Hitting Set problem input is a universe U equipped with a capacity function $\mathsf{cap} : U \to \mathbb{N}$, and a collection $\mathcal{A}$ of subsets of U, each of size at most d. The task is to find a minimum size subset S of U and an assignment $\phi : \mathcal{A} \to S$ such that, for every set $A \in \mathcal{A}$ we have $\phi(A) \in A$ and for every $x \in U$ we have $|\phi^{-1}(x)| \leq \mathsf{cap}(x)$. When $d = 2$ the problem is known under the name Capacitated Vertex Cover. In Weighted Capacitated d-Hitting Set each element of U has a positive integer weight and the goal is to find a capacitated hitting set of minimum weight.

In this paper we initiate the study of parameterized (approximation) algorithms for Capacitated d-Hitting Set. Capacitated d-Hitting Set is a well studied problem and is known to admit a d-approximation algorithm and no $(d - \epsilon)$-approximation under UGC for any $\epsilon > 0$. Further, unweighted Capacitated d-Hitting Set for $d \geq 3$ is W[1]-hard parameterized by solution size. Our main result is a parameterized approximation algorithm that runs in time $\left(\frac{k}{\epsilon}\right)^k 2^{k^{O(kd)}} (|U| + |\mathcal{A}|)^{O(1)}$ and either concludes that there is no solution of size at most k or outputs a solution S of size at most $4/3 \cdot k$ and weight at most $2 + \epsilon$ times the minimum weight of a solution whose size is at most k. We also complement our algorithmic results with hardness results.

3.21 Parameterized Complexity in Explainable AI

Stefan Szeider (TU Wien, AT)

License Creative Commons BY 4.0 International license
© Stefan Szeider

As machine learning (ML) models become increasingly complex, interpretability and explainability in ML-based decisions, referred to as eXplainable AI (XAI) has become an important objective. In this talk, we introduce two computational questions that arise in the context of XAI and discuss possible parameterizations.

1. Finding a small symbolic ML model that best represents given data.
2. Generating a concise explanation for a prediction from an existing symbolic model.

We discuss recent results [1, 2, 3, 4, 5, 6, 7, 8] and examine these questions for various model types, including decision trees (DT), decision sets (DS), decision lists (DL), and binary decision diagrams (BDD), highlighting possible avenues for future research.

References

1 K. Dabrowski, E. Eiben, S. Ordyniak, G. Paesani, and S. Szeider. Learning small decision trees for data of low rank-width. In J. Dy and S. Natarajan, editors, *AAAI'24*, pages 10476–10483. AAAI Press, 2024.

2 E. Eiben, S. Ordyniak, G. Paesani, and S. Szeider. Learning small decision trees with large domain. In E. Elkind, editor, *IJCAI'23*, pages 3184–3192, 2023.

3 H. Gahlawat and M. Zehavi. Learning small decision trees with few outliers: A parameterized perspective. In M. J. Wooldridge, J. G. Dy, and S. Natarajan, editors, *AAAI'24*, pages 12100–12108. AAAI Press, 2024.

4 C. Komusiewicz, P. Kunz, F. Sommer, and M. Sorge. On computing optimal tree ensembles. In A. Krause, E. Brunskill, K. Cho, B. Engelhardt, S. Sabato, and J. Scarlett, editors, *ICML'23*, pages 17364–17374. PMLR, 2023.

5 S. Ordyniak, G. Paesani, M. Rychlicki, and S. Szeider. Explaining decisions in ML models: a parameterized complexity analysis. In *KR'24*, 2024. To appear.

6 S. Ordyniak, G. Paesani, M. Rychlicki, and S. Szeider. A general theoretical framework for learning smallest interpretable models. In J. Dy and S. Natarajan, editors, *AAAI'24*, pages 10662–10669. AAAI Press, 2024.

7 S. Ordyniak, G. Paesani, and S. Szeider. The parameterized complexity of finding concise local explanations. In *IJCAI'23*, pages 3312–3320. ijcai.org, 2023.

8 S. Ordyniak and S. Szeider. Parameterized complexity of small decision tree learning. In *AAAI'21*, pages 6454–6462. AAAI Press, 2021.

3.22 Complexity Framework for Subgraph-Free Graphs & Beyond

Erik Jan van Leeuwen (Utrecht University, NL)

License Creative Commons BY 4.0 International license
© Erik Jan van Leeuwen
Joint work of Hans Bodlaender, Matthew Johnson, Barnaby Martin, Jelle J. Oostveen, Sukanya Pandey, Daniël Paulusma, Siani Smith, Erik Jan van Leeuwen
Main reference Matthew Johnson, Barnaby Martin, Jelle J. Oostveen, Sukanya Pandey, Daniël Paulusma, Siani Smith, Erik Jan van Leeuwen: "Complexity Framework For Forbidden Subgraphs", CoRR, Vol. abs/2211.12887, 2022.
URL https://doi.org/10.48550/ARXIV.2211.12887

For any finite set $\mathcal{H} = \{H_1, \ldots, H_p\}$ of graphs, a graph is $\mathcal{H}$-subgraph-free if it does not contain any of $H_1, \ldots, H_p$ as a subgraph. We give a new framework that precisely classifies whether problems are "efficiently solvable" or "computationally hard" for $\mathcal{H}$-subgraph-free graphs, depending on $\mathcal{H}$. To illustrate the broad applicability of our framework, we study partitioning, covering and packing problems, network design problems, and width parameter problems. We apply the framework to obtain a dichotomy (depending on $\mathcal{H}$) between polynomial-time solvability and NP-completeness of those problems. For other problems, we obtain a dichotomy between almost-linear-time solvability and having no subquadratic-time algorithm (conditioned on some hardness hypotheses). Along the way, we unify and strengthen known results from the literature. We also discuss recent insights into the complexity on $\mathcal{H}$-subgraph-free graphs of problems that do not fall within the framework.

3.23 Planar Disjoint Paths, Treewidth, and Kernels

Michał Włodarczyk (University of Warsaw, PL)

License Creative Commons BY 4.0 International license
© Michał Włodarczyk
Joint work of Michał Włodarczyk, Meirav Zehavi
Main reference Michał Włodarczyk, Meirav Zehavi: "Planar Disjoint Paths, Treewidth, and Kernels", in Proc. of the 64th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2023, Santa Cruz, CA, USA, November 6-9, 2023, pp. 649–662, IEEE, 2023.
URL https://doi.org/10.1109/FOCS57990.2023.00044

In the Planar Disjoint Paths problem, one is given an undirected planar graph with a set of k vertex pairs (s_i, t_i) and the task is to find k pairwise vertex-disjoint paths such that the i-th path connects s_i to t_i. We study the problem through the lens of kernelization, aiming at efficiently reducing the input size in terms of a parameter. We show that Planar Disjoint Paths does not admit a polynomial kernel when parameterized by k unless coNP $\subseteq$ NP/poly, resolving an open problem by [Bodlaender, Thomassé, Yeo, ESA'09]. Moreover, we rule out the existence of a polynomial Turing kernel unless the WK-hierarchy collapses. Our reduction carries over to the setting of edge-disjoint paths, where the kernelization status remained open even in general graphs.

On the positive side, we present a polynomial kernel for Planar Disjoint Paths parameterized by $k+tw$, where tw denotes the treewidth of the input graph. As a consequence of both our results, we rule out the possibility of a polynomial-time (Turing) treewidth reduction to $tw = k^{O(1)}$ under the same assumptions. To the best of our knowledge, this is the first hardness result of this kind. Finally, combining our kernel with the known techniques [Adler, Kolliopoulos, Krause, Lokshtanov, Saurabh, Thilikos, JCTB'17; Schrijver, SICOMP'94] yields an alternative (and arguably simpler) proof that Planar Disjoint Paths can be solved in time $2^{O(k^2)} \cdot n^{O(1)}$, matching the result of [Lokshtanov, Misra, Pilipczuk, Saurabh, Zehavi, STOC'20].

4 Open problems

4.1 Extending 1-Planar Drawings by a Few Vertices

Robert Ganian (TU Wien, AT)

License Creative Commons BY 4.0 International license
© Robert Ganian

A drawing of a graph G is *1-planar* (or 1-*plane*) if each edge has at most one crossing. We restrict our attention to simple drawings – in particular, the curves representing edges are not allowed to "touch" or cross through vertices which are not their endpoints. A drawing α *extends* a drawing β if $\beta \subseteq \alpha$. The following question was left open in an ICALP paper of Eiben, Ganian, Hamm, Klute and Nöllenburg [3].

Is the following problem FPT w.r.t. k?

Given a graph G, a vertex subset $X \subseteq V(G)$ of size k and a 1-planar drawing $\mathcal{H}$ of the graph $H = G - X$, does there exist a 1-planar drawing of G which extends $\mathcal{H}$?

Essentially, we are given an "almost complete" partial 1-planar drawing of a graph G and want to extend it to a full drawing, where the parameter tells us how many vertices are still missing from the partial drawing. Note that the statement in the paper is slightly more general as it also allows individual edges to be missing from H, but the real difficult/interesting case is the one above.

The problem was shown to be in XP in a follow-up MFCS paper [3], and the edge-deletion version of the problem – i.e., where we parameterize by how many edges are missing – is known to be FPT [2] (by using decomposition techniques). Several follow-ups to the latter result are known by now [4, 5, 1], but the vertex-deletion case has not been cracked so far.

References

1 S. Bhore, R. Ganian, L. Khazaliya, F. Montecchiani, and M. Nöllenburg. Extending orthogonal planar graph drawings is fixed-parameter tractable. In E. W. Chambers and J. Gudmundsson, editors, *39th International Symposium on Computational Geometry, SoCG 2023, June 12-15, 2023, Dallas, Texas, USA*, volume 258 of *LIPIcs*, pages 18:1–18:16. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2023.

2 E. Eiben, R. Ganian, T. Hamm, F. Klute, and M. Nöllenburg. Extending nearly complete 1-planar drawings in polynomial time. In J. Esparza and D. Král', editors, *45th International Symposium on Mathematical Foundations of Computer Science, MFCS 2020, August 24-28, 2020, Prague, Czech Republic*, volume 170 of *LIPIcs*, pages 31:1–31:16. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2020.

3 E. Eiben, R. Ganian, T. Hamm, F. Klute, and M. Nöllenburg. Extending partial 1-planar drawings. In A. Czumaj, A. Dawar, and E. Merelli, editors, *47th International Colloquium on Automata, Languages, and Programming, ICALP 2020, July 8-11, 2020, Saarbrücken, Germany (Virtual Conference)*, volume 168 of *LIPIcs*, pages 43:1–43:19. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2020.

4 R. Ganian, T. Hamm, F. Klute, I. Parada, and B. Vogtenhuber. Crossing-optimal extension of simple drawings. In N. Bansal, E. Merelli, and J. Worrell, editors, *48th International Colloquium on Automata, Languages, and Programming, ICALP 2021, July 12-16, 2021, Glasgow, Scotland (Virtual Conference)*, volume 198 of *LIPIcs*, pages 72:1–72:17. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2021.

5 T. Hamm and P. Hliněný. Parameterised partially-predrawn crossing number. In X. Goaoc and M. Kerber, editors, *38th International Symposium on Computational Geometry, SoCG 2022, June 7-10, 2022, Berlin, Germany*, volume 224 of *LIPIcs*, pages 46:1–46:15. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2022.

4.2 Disjoint Paths Reconfiguration in Planar Graphs

Yusuke Kobayashi (Kyoto University, JP)

License Creative Commons BY 4.0 International license
© Yusuke Kobayashi

Let $G = (V, E)$ be a graph with distinct terminals $s_1, \dots, s_k, t_1, \dots, t_k$. A tuple $\mathcal{P} = (P_1, \dots, P_k)$ of paths is called a *linkage* if they are vertex-disjoint and each P_i connects s_i and t_i for $i = 1, \dots, k$. For two linkages $\mathcal{P} = (P_1, \dots, P_k)$ and $\mathcal{Q} = (Q_1, \dots, Q_k)$, we say that $\mathcal{P}$ is *adjacent* to $\mathcal{Q}$ if there exists $i \in \{1, \dots, k\}$ such that $P_j = Q_j$ for $j \in \{1, \dots, k\} \setminus \{i\}$ and $P_i \neq Q_i$. We say that a sequence $\langle \mathcal{P}_1, \mathcal{P}_2, \dots, \mathcal{P}_\ell \rangle$ of linkages is a *reconfiguration sequence* from $\mathcal{P}_1$ to $\mathcal{P}_\ell$ if $\mathcal{P}_i$ and $\mathcal{P}_{i+1}$ are adjacent for $i = 1, \dots, \ell - 1$. If such a sequence exists, we say that $\mathcal{P}_1$ is *reconfigurable* to $\mathcal{P}_\ell$.

In Disjoint Paths Reconfiguration, we are given a graph $G = (V, E)$, distinct terminals $s_1, \dots, s_k, t_1, \dots, t_k$, and two linkages $\mathcal{P}$ and $\mathcal{Q}$. The objective is to determine whether $\mathcal{P}$ is reconfigurable to $\mathcal{Q}$ or not. It is shown in [1] that Disjoint Paths Reconfiguration is PSPACE-complete for $k = 2$. The polynomial solvability for the planar case is open.

For fixed k, is there a polynomial-time algorithm for DISJOINT PATHS RECONFIGURATION in planar graphs? The problem is open even for $k = 2$.

References

1 T. Ito, Y. Iwamasa, N. Kakimura, Y. Kobayashi, S.-i. Maezawa, Y. Nozaki, Y. Okamoto, and K. Ozeki. Rerouting Planar Curves and Disjoint Paths. In K. Etessami, U. Feige, and G. Puppis, editors, *50th International Colloquium on Automata, Languages, and Programming (ICALP 2023)*, volume 261 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 81:1–81:19, Dagstuhl, Germany, 2023. Schloss Dagstuhl – Leibniz-Zentrum für Informatik.

4.3 Three-Sets Cut-Uncut

Tuukka Korhonen (University of Copenhagen, DK)

License Ⓒ Creative Commons BY 4.0 International license
© Tuukka Korhonen

The Three-Sets Cut-Uncut Problem. Given a graph G and three sets of terminal vertices T_1, T_2, and T_3, the goal is to remove a minimum number of edges from G such that in the resulting graph:

1. For each $i \in \{1, 2, 3\}$, every pair of vertices $u, v \in T_i$ remains connected by a path.
2. For every pair of distinct indices $i \neq j$, there is no path between any vertex $u \in T_i$ and any vertex $v \in T_j$.

(Note that it is possible that no solution exists.)

We ask: What is the parameterized complexity of the Three-Sets Cut-Uncut problem on planar graphs when parameterized by $k = |T_1| + |T_2| + |T_3|$? In particular, it remains open whether the problem is polynomial-time solvable even for the case $|T_1| = |T_2| = 1 \text{and} |T_3| = 2$.

For related literature, please see the discussion and references in [1].

References

1 M. Bentert, P. G. Drange, F. V. Fomin, P. A. Golovach, and T. Korhonen. Two-sets cut-uncut on planar graphs. In K. Bringmann, M. Grohe, G. Puppis, and O. Svensson, editors, *51st International Colloquium on Automata, Languages, and Programming, ICALP 2024, July 8-12, 2024, Tallinn, Estonia*, volume 297 of *LIPIcs*, pages 22:1–22:18. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2024.

4.4 Beating $n!$ for Permutation CSPs

Marcin Pilipczuk (University of Warsaw, PL)

License Ⓒ Creative Commons BY 4.0 International license
© Marcin Pilipczuk

We consider a problem where we ask for an existence of a permutation $\pi : [n] \to [n]$ satisfying a number of *constraints*. Each constraint is of the form $(\pi(a) < \pi(b)) \vee (\pi(c) < \pi(d))$ for some $a, b, c, d \in [n]$.

Clearly, this problem can be solved in time roughly $n!$ by brute-force enumeration. Leif Eriksson in his master thesis [1] observed that this can be improved to roughly $((n/2)!)^2$. Does there exist an $2^{o(n \log n)}$-time algorithm?

When the constraints are of the form $\pi(a) < \pi(b)$, a simple dynamic programming algorithm solves the problem in $2^{\mathcal{O}(n)}$ time. To the best of my knowledge, the problem remains open for constraints being an alternative of three comparisons. A simple reduction from $k \times k$ CLIQUE [2] gives an ETH $2^{\Omega(n \log n)}$ lower bound for constraints being an alternative of four comparisons.

References

1 L. Eriksson. Solving temporal CSPs via enumeration and SAT compilation, 2019. Master thesis.

2 D. Lokshtanov, D. Marx, and S. Saurabh. Slightly superexponential parameterized problems. *SIAM J. Comput.*, 47(3):675–702, 2018.

4.5 Better bounds for directed flow-augmentation

Marcin Pilipczuk (University of Warsaw, PL)

License Ⓒ Creative Commons BY 4.0 International license
© Marcin Pilipczuk

Let G be a directed (multi)graph with distinguished vertices $s, t \in V(G)$ and let k be an integer parameter. For an inclusion-wise minimal st-cut $Z \subseteq E(G)$, a set $A \subseteq V(G) \times V(G)$ is *compatible* with Z if Z remains an st-cut in $G + A := (V(G), E(G) \uplus A)$.

The flow-augmentation technique, in its basic form, can be stated as follows: given G, s, t, k as above, one can in FPT time find a family $\mathcal{A}$ of subsets of $V(G) \times V(G)$ such that for every inclusion-wise minimal st-cut Z of size at most k, there exists $A \in \mathcal{A}$ that is compatible with Z and such that, furthermore, Z becomes a *minimum* st-cut in $G + A$. The work [1] provides $\mathcal{A}$ of size bounded by

$$2^{\mathcal{O}(k^4 \log k)} \cdot (\log n)^{\mathcal{O}(k^3)} \leq 2^{\mathcal{O}(k^4 \log k)} n^{o(1)}.$$

The only known lower bound known is an observation that every important separator Z requires a different set A in the family $\mathcal{A}$. As the number of important separators of size at most k can be as large as k-th Catalan number (i.e., $\Omega(4^k k^{-3/2})$), this is also a lower bound on the size of $\mathcal{A}$.

Please provide a better upper or lower bound.

References

1 E. J. Kim, S. Kratsch, M. Pilipczuk, and M. Wahlström. Directed flow-augmentation. In S. Leonardi and A. Gupta, editors, *STOC '22: 54th Annual ACM SIGACT Symposium on Theory of Computing, Rome, Italy, June 20 – 24, 2022*, pages 938–947. ACM, 2022.

4.6 Space efficient Min Cut

Michał Pilipczuk (University of Warsaw, PL)

License Creative Commons BY 4.0 International license
© Michał Pilipczuk

The following question seems to be fundamental for very low parameterized space complexity classes.

Given an undirected n-vertex graph G, a pair of vertices s and t, and an integer k, is it possible to decide in deterministic space $f(k) + \mathcal{O}(\log n)$, for a computable f, the following question: is there a vertex subset $X \subseteq V(G) \setminus \{s, t\}$ with $|X| \leq k$ such that every s-t path intersects X.

I would conjecture that the answer should be negative, and that establishing it could provide some technique for lower bounds on parameterized space complexity.

4.7 Bisection in Planar Graphs

Saket Saurabh (The Institute of Mathematical Sciences – Chennai, IN)

License Creative Commons BY 4.0 International license
© Saket Saurabh

In the classic **Minimum Bisection** problem, we are given as input a graph G and an integer k. The task is to determine whether there is a partition of $V(G)$ into two parts A and B such that $||A| - |B|| \leq 1$ and there are at most k edges with one endpoint in A and the other in B.

1. Is the problem polynomial time solvable on planar graphs?
2. Does the problem admit a quasi-polynomial time algorithm on planar graphs?
3. Does the problem admit subexponential (in k) time FPT algorithm?

4.8 Skew Multicut on Planar DAGs

Michał Wlodarczyk (University of Warsaw, PL)

License Creative Commons BY 4.0 International license
© Michał Wlodarczyk

In Skew Multicut one is given a (directed) graph G, a set $T \subseteq V(G)$ of terminals, an ordering σ over T, and a budget k. We ask whether there is a set $X \subseteq V(G) \setminus T, |X| \leq k$, such that for each pair $s, t \in T$ with $s <_\sigma t$ there is no (s, t)-path in $G - X$.

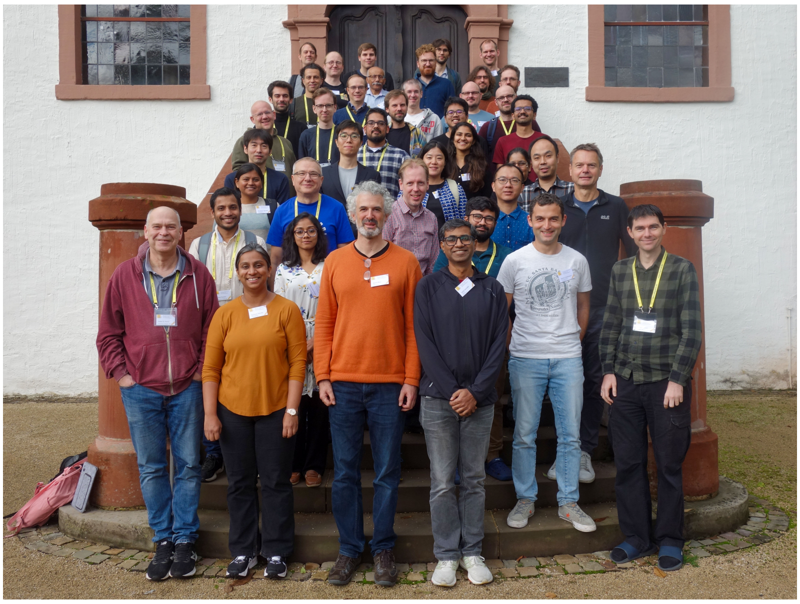
Does Skew Multicut admit a polynomial kernel on planar DAGs when parameterized by $k + |T|$?

If yes, can you make the kernel oblivious to σ? That is, the ordering σ is revealed after the compressed instance is returned.

Such an oblivious kernel would be helpful for kernelization of planar DFVS.

Participants

- Akanksha Agrawal
Indian Institute of Techology Madras, IN
- Aditya Anand
University of Michigan – Ann Arbor, US
- Matthias Bentert
University of Bergen, NO
- Édouard Bonnet
ENS – Lyon, FR
- Joseph Cheriyan
University of Waterloo, CA
- Éric Colin de Verdière
Gustave Eiffel University – Marne-la-Vallée, FR
- Eduard Eiben
Royal Holloway, University of London, GB
- Fedor V. Fomin
University of Bergen, NO
- Robert Ganian
TU Wien, AT
- Petr A. Golovach
University of Bergen, NO
- Bart Jansen
TU Eindhoven, NL
- Eun Jung Kim
KAIST – Daejeon, KR & CNRS – Paris, FR
- Yusuke Kobayashi
Kyoto University, JP
- Christian Komusiewicz
Friedrich-Schiller-Universität Jena, DE
- Tuukka Korhonen
University of Copenhagen, DK
- Stefan Kratsch
HU Berlin, DE
- Madhumita Kundu
University of Bergen, NO
- Euiwoong Lee
University of Michigan – Ann Arbor, US
- Bingkai Lin
Nanjing University, CN
- William Lochet
CNRS – Montpellier, FR
- Daniel Lokshtanov
University of California – Santa Barbara, US
- Dániel Marx
CISPA – Saarbrücken, DE
- Neeldhara Misra
Indian Institute of Techology – Madras, IN
- Pranabendu Misra
Chennai Mathematical Institute, IN
- Jesper Nederlof
Utrecht University, NL
- Daniel Neuen
MPI für Informatik – Saarbrücken, DE
- Fahad Panolan
University of Leeds, GB
- Marcin Pilipczuk
University of Warsaw, PL
- Michał Pilipczuk
University of Warsaw, PL
- Thatchaphol Saranurak
University of Michigan – Ann Arbor, US
- Ignasi Sau Valls
LIRMM, Université de Montpellier, CNRS – Montpellier, FR
- Saket Saurabh
The Institute of Mathematical Sciences – Chennai, IN
- Roohani Sharma
MPI für Informatik – Saarbrücken, DE
- Ramanujan Sridharan
University of Warwick – Coventry, GB
- Giannos Stamoulis
University of Warsaw, PL
- Vaishali Surianarayanan
University of California – Santa Barbara, US
- Stefan Szeider
TU Wien, AT
- Erik Jan van Leeuwen
Utrecht University, NL
- Magnus Wahlström
Royal Holloway, University of London, GB
- Michał Wlodarczyk
University of Warsaw, PL
- Jie Xue
New York University – Shanghai, CN



Report from Dagstuhl Seminar 24421

SAT and Interactions

Olaf Beyersdorff*[1], Laura Kovács*[2], Meena Mahajan*[3], Martina Seidl*[4], and Kaspar Kasche†[5]

1 **Friedrich-Schiller-Universität Jena, DE.** olaf.beyersdorff@uni-jena.de
2 **TU Wien, AT.** laura.kovacs@tuwien.ac.at
3 **The Institute of Mathematical Sciences (a CI of Homi Bhabha National Institute) – Chennai, IN.** meena@imsc.res.in
4 **Johannes Kepler Universität Linz, AT.** martina.seidl@jku.at
5 **Friedrich-Schiller-Universität Jena, DE.** kaspar.kasche@uni-jena.de

Abstract

This report documents the program and the outcomes of Dagstuhl Seminar "SAT and Interactions" (24421). The seminar brought together theoreticians and practitioners from the areas of proof complexity, SAT and QBF solving, and first-order theorem proving, who discussed recent developments in their fields and embarked on an interdisciplinary exchange of ideas and techniques between these neighbouring subfields of SAT.

Seminar October 13–18, 2024 – https://www.dagstuhl.de/24421

2012 ACM Subject Classification Theory of computation → Proof complexity; Theory of computation → Proof theory; Theory of computation → Automated reasoning; Theory of computation → Complexity theory and logic

Keywords and phrases SAT, QBF, proof complexity, solving, first-order logic, automated theorem proving

Digital Object Identifier 10.4230/DagRep.14.10.22

1 Executive Summary

Olaf Beyersdorff (Friedrich-Schiller-Universität Jena, DE, olaf.beyersdorff@uni-jena.de)
Laura Kovács (TU Wien, AT, lkovacs@forsyte.at)
Meena Mahajan (The Institute of Mathematical Sciences – Chennai, IN, meena@imsc.res.in)
Martina Seidl (Johannes Kepler Universität Linz, AT, martina.seidl@jku.at)

License Creative Commons BY 4.0 International license
© Olaf Beyersdorff, Laura Kovács, Meena Mahajan, and Martina Seidl

The problem of deciding whether a propositional formula is satisfiable (SAT) is one of the most fundamental problems in computer science. Its theoretical significance derives from the Cook-Levin Theorem, identifying SAT as the first NP-complete problem. Since then SAT has become a reference for an enormous variety of complexity statements, among them the celebrated P vs NP problem.

There are many generalisations of SAT to logics such as quantified Boolean formulas (QBF), modal and first-order (FO) logics. Often these logics present harder satisfiability problems (e.g. PSPACE-complete for QBF), but can express many practically relevant problems more succinctly, thus applying to more real-world problems.

* Editor / Organizer
† Editorial Assistant / Collector

Except where otherwise noted, content of this report is licensed under a Creative Commons BY 4.0 International license

Due to its practical implications, intensive research has been performed to solve SAT problems in an automated fashion. The last decades have seen the development of practically efficient algorithms for SAT, QBF, and further logics, and their implementation as solvers, which successfully solve huge industrial instances.

As the fourth in its series, the Dagstuhl Seminar took a broad perspective on the theory of SAT, encompassing propositional logic, QBF, and first-order theorem proving. Its main aim was to bring together researchers from different areas of activity in SAT and first-order logic, including computational complexity, proof complexity, proof theory, theorem proving, and solving, so that they can communicate state-of-the-art advances and embark on a systematic interdisciplinary interaction.

The Dagstuhl Seminar placed particular emphasis on the three following fields: propositional logic (complexity, proof complexity, solving), QBF (proof complexity and solving), and FO theorem proving. A particularly novel feature was the interaction of the communities active in proof complexity and solving of SAT/QBF (the propositional logics) with the first-order theorem proving community. There appeared to be overall consensus among the participants that this interchange of ideas between SAT solving techniques and first-order theorem proving was very stimulating and might particularly prove useful towards further efficient implementations of first-order proof rules.

To facilitate interactions between participants from the different fields, the seminar included a number of survey talks to introduce neighbouring communities to the main notions, results, and challenges of the represented areas. The following survey talks were given towards the beginning of the seminar:

- Marc Vinyals: SAT Solving and Proof Complexity;
- Friedrich Slivovsky: QBF Solving and Proof Complexity;
- Cesare Tinelli: An introduction to Satisfiability Modulo Theories;
- Stephan Schulz: First-order Theorem Proving.

Each of these surveys was accompanied by one or more sessions with contributed talks dedicated to recent specific results of the field.

The seminar also included an open problem session where participants discussed open research directions and specific problems. The following topics were discussed:

- Stephan Schulz: Can we achieve good engineering and long-term viability of systems? Software projects often get abandoned over time, especially if the original authors leave. We are looking for technical and organisational solutions to mitigate this.
- Sophie Tourret: Beyond critical.
  There are established techniques to compare the hardness of propositional formulas. Can we find analogous techniques for first-order logic and SMT? Particularly interesting are cases outside the decidable fragments of these theories.
- Neil Thapen: Where is symmetry breaking in TFNP?
  Symmetry breaking techniques can be understood as the optimisation problem of finding a lexically minimal assignment, which is in TFNP. We know it is in PLS, it might be in CLS, but can we at least show that it is not in FP? This has implications on the strength of Extended Frege.
- Adrian Rebola-Pardo: How should we design future proof formats?
  There is a variety of practical proof formats that need to be suitable for verification and querying. Is it possible to design good universal proof formats? Important considerations include binary encodings, non-clausal representations, specific addition and deletion rules, the needs of incremental SAT solvers, and parallel proof checking.

- Florent Capelli: Properties of a hypergraph measure. The β-hyperorderwidth is a purely graph-theoretic measure on hypergraphs. How does it compare to established hypergraph measures? For example, there is a hypergraph that has β-hyperorderwidth 1, but its incidence graph has treewidth n. Is this possible the other way around? Can we generalize the definition of β-hyperorderwidth?

The seminar included ample time for discussions and informal interactions, a feature that appeared to be largely welcomed and productively used. On Wednesday afternoon we organised a traditional well-attended hike. On Thursday evening, we had a joyful music night with contributions from Sophie Tourret, Ilario Bonacina, Dominik Scheder, Florent Capelli, and Kaspar Kasche. They played music by Marin Marais, Georg Philipp Telemann, Wolfgang Amadeus Mozart, Francis Poulenc, and George Gershwin.

The organisers believe that the seminar fulfilled their original high goals: the talks were well received and triggered many discussions. Many participants reported about the inspiring seminar atmosphere, fruitful interactions, and a generally positive experience. The organisers and participants wish to thank the staff and the management of Schloss Dagstuhl for their assistance and excellent support in the arrangement of a very successful and productive event.

2 Table of Contents

3 Overview of Talks

3.1 Generalized Satisfiability Problems via Operator Assignments

Albert Atserias (UPC Barcelona Tech, ES)

License Creative Commons BY 4.0 International license
© Albert Atserias
Joint work of Albert Atserias, Phokion G. Kolaitis, Simone Severini
Main reference Albert Atserias, Phokion G. Kolaitis, Simone Severini: "Generalized satisfiability problems via operator assignments", J. Comput. Syst. Sci., Vol. 105, pp. 171–198, 2019.
URL https://doi.org/10.1016/J.JCSS.2019.05.003

Schaefer introduced a framework for generalized satisfiability problems on the Boolean domain and characterized the computational complexity of such problems. We investigate an algebraization of Schaefer's framework in which the Fourier transform is used to represent constraints by multilinear polynomials in a unique way. The polynomial representation of constraints gives rise to a relaxation of the notion of satisfiability in which the values to variables are linear operators on some Hilbert space. For the case of constraints given by a system of linear equations over the two-element field, this relaxation has received considerable attention in the foundations of quantum mechanics, where such constructions as the Mermin-Peres magic square show that there are systems that have no solutions in the Boolean domain, but have solutions via operator assignments on some finite-dimensional Hilbert space. We obtain a complete characterization of the classes of Boolean relations for which there is a gap between satisfiability in the Boolean domain and the relaxation of satisfiability via operator assignments. To establish our main result, we adapt the notion of primitive-positive definability (pp-definability) to our setting, a notion that has been used extensively in the study of constraint satisfaction problems. Here, we show that pp-definability gives rise to gadget reductions that preserve satisfiability gaps. We also present several additional applications of this method. In particular and perhaps surprisingly, we show that the relaxed notion of pp-definability in which the quantified variables are allowed to range over operator assignments gives no additional expressive power in defining Boolean relations.

3.2 A new hypergraph measure for #SAT

Florent Capelli (University of Artois/CNRS – Lens, FR)

License Creative Commons BY 4.0 International license
© Florent Capelli
Joint work of Florent Capelli, Oliver Irwin
Main reference Florent Capelli, Oliver Irwin: "Direct Access for Conjunctive Queries with Negations", in Proc. of the 27th International Conference on Database Theory, ICDT 2024, March 25-28, 2024, Paestum, Italy, LIPIcs, Vol. 290, pp. 13:1–13:20, Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2024.
URL https://doi.org/10.4230/LIPICS.ICDT.2024.13

The problem #SAT of counting the number of satisfying assignments of a CNF formula is a notoriously hard combinatorial problem, even for very restricted classes of CNF formulas, such as monotone 2-CNF. Islands of tractability have been however discovered for the problem by restricting the way clauses and variables interact inside the formula. CNF formulas whose hypergraph is beta-acyclic have been shown tractable for #SAT and knowledge compilation. This result gave hope toward proving tractability for a larger family of formulas, those having bounded beta-hypertreewidth. However, the definition of beta-hypertreewidth is not based on graph decomposition and hence are hard to deal with algorithmically. The hardness of #SAT on such formulas remains open. In this talk, I will present a new hypergraph parameter, called beta-hyperorder width, for which #SAT and knowledge compilation are tractable. This measure naturally generalizes primal treewidth and beta-acyclicity while being defined via elimination orders which can be leveraged into an algorithm.

The results presented in this talk are adapted from recent work with Oliver Irwin, "Direct Access for Conjunctive Queries with Negations" which appeared at ICDT 2025, which present a more database flavor version of the results.

3.3 Propositional Proofs for PSPACE problems (including #SAT)

Leroy Nicholas Chew (TU Wien, AT)

License Creative Commons BY 4.0 International license
© Leroy Nicholas Chew
Joint work of Leroy Nicholas Chew, Sravanthi Chede, Anil Shukla

We show how to use the correctness of a polynomial space algorithm that decides some language L as a basis for an L-proof system. The first example is a proof system for #SAT.

The proofs consist of a non-deterministically guessed multi circuit. The circuit takes in a binary integer to represent the time and outputs the memory of the algorithm at said timepoint. In #SAT we can simplify this to calculating the cumulative function (or running count) of models counted up to a point. Checking the circuit for correctness is a CoNP problem so we only need a propositional proof that proves the circuit is in fact correct.

Our proof system is called Circuit Linear Induction Proposition (CLIP), CLIP for #SAT simulates all previous #SAT proof systems. This gives us a new opportunity to create strong proof systems for PSPACE languages that do not rely on a direct translation to Quantified Boolean Formulas (QBF). In this paper we explore some proof complexity results of systems of this form and study the connection to QBF proof complexity.

This talk adapts both the FSTTCS paper "Circuits Proofs and Propositional Model Counting" (to appear) by Sravanthi Chede, Leroy Chew and Anil Shukla and an upcoming paper "Propositional Proofs for PSPACE Problems" by Leroy Chew.

3.4 Pudlák-Buss games for (non)deterministic branching programs

Anupam Das (University of Birmingham, GB)

License Creative Commons BY 4.0 International license
© Anupam Das
Joint work of Anupam Das, Avgerinos Delkos

A natural nonuniform version of (N)L is given by (non)deterministic branching programs. These may be naturally given a proof theoretic treatment via a system for (non)deterministic decision trees with extension to represent dagness. Such systems, eL(N)DT, were proposed by Buss, Das and Knop in '20, who also established their basic proof complexity results.

In this talk I will speak about recent work with Avgerinos Delkos recasting those systems as Prover-Adversary games, à la Pudlák & Buss. Our main result is a correspondence between strategies and proofs. Along the way, we establish a proof complexity theoretic version of the Immerman-Szelepcsényi theorem that NL=coNL. One novelty here is that our proof exploits the ability to count at the proof level rather than the formula level, significantly simplifying our construction.

3.5 Truly Supercritical Trade-offs for Resolution, Cutting Planes, and Monotone Circuits

Susanna de Rezende (Lund University, SE)

License Creative Commons BY 4.0 International license
© Susanna de Rezende
Joint work of Susanna F. de Rezende, Noah Fleming, Duri Andrea Janett, Jakob Nordström, Shuo Pang
Main reference Susanna F. de Rezende, Noah Fleming, Duri Andrea Janett, Jakob Nordström, Shuo Pang: "Truly Supercritical Trade-offs for Resolution, Cutting Planes, Monotone Circuits, and Weisfeiler-Leman", Electron. Colloquium Comput. Complex., Vol. TR24-185, 2024.
URL https://eccc.weizmann.ac.il/report/2024/185

We present supercritical trade-off for monotone circuits, showing that there are functions computable by small circuits for which any circuit must have depth super-linear or even super-polynomial in the number of variables, far exceeding the linear worst-case upper bound. We obtain similar trade-offs in proof complexity, where we establish the first size-depth trade-offs for cutting planes and resolution that are truly supercritical, i.e., in terms of formula size rather than number of variables, and we also show supercritical trade-offs between width and size for treelike resolution. Our results build on a new supercritical width-depth trade-off for resolution, obtained by refining and strengthening the compression scheme for the Cop-Robber game in [1], which we will have heard about in the previous talk. The supercritical size-depth trade-offs for monotone circuits, cutting planes and resolution, and the supercritical size-width trade-off for tree-like resolution follow from improved lifting theorems that might be of independent interest. This is joint work with Noah Fleming, Duri Andrea Janett, Jakob Nordström, and Shuo Pang.

References

1 Martin Grohe, Moritz Lichter, Daniel Neuen, and Pascal Schweitzer. Compressing CFI graphs and lower bounds for the Weisfeiler-Leman refinements. In *Proceedings of the 64th Annual IEEE Symposium on Foundations of Computer Science (FOCS '23)*, pages 798–809, November 2023.

3.6 SAT modulo IPASIR-UP

Katalin Fazekas (TU Wien, AT)

License Creative Commons BY 4.0 International license
© Katalin Fazekas
Joint work of Katalin Fazekas, Aina Niemetz, Mathias Preiner, Markus Kirchweger, Stefan Szeider, Armin Biere, Airna Niemetz, Mathias Fleury, Florian Pollitt
Main reference Katalin Fazekas, Aina Niemetz, Mathias Preiner, Markus Kirchweger, Stefan Szeider, Armin Biere: "IPASIR-UP: User Propagators for CDCL", in Proc. of the 26th International Conference on Theory and Applications of Satisfiability Testing, SAT 2023, July 4-8, 2023, Alghero, Italy, LIPIcs, Vol. 271, pp. 8:1–8:13, Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2023.
URL https://doi.org/10.4230/LIPICS.SAT.2023.8

Modern SAT solvers are often integrated as sub-reasoning engines into more complex tools to address problems beyond Boolean satisfiability. Consider, for example, solvers for Satisfiability Modulo Theories (SMT), combinatorial optimization, model enumeration, and model counting. In our work, we have proposed a general interface for CDCL SAT solvers to capture the essential functionalities necessary to simplify and improve use cases that require more fine-grained interaction with the SAT solver than provided by the standard IPASIR interface.

In this talk I will briefly describe the interface and highlight the main changes made since its introduction. I will also present our ongoing work and the challenges and future work we are currently considering. The aim of the talk is to initiate further discussions with the other participants of the seminar to get a better understanding of the required features and typical use cases.

3.7 First-order theorem proving for operator statements

Clemens Hofstadler (Universität Kassel, DE)

License Creative Commons BY 4.0 International license
© Clemens Hofstadler
Joint work of Clemens Hofstadler, Clemens G. Raab, Georg Regensburger
Main reference Clemens Hofstadler, Clemens G. Raab, Georg Regensburger: "Universal truth of operator statements via ideal membership",CoRR, Vol. abs/2212.11662 2024.
URL https://doi.org/10.48550/arXiv.2212.11662

Algebraic statements involving matrices or linear operators appear in various branches of mathematics and related disciplines. In linear algebra and geometry, we study matrices and their properties as fundamental objects, while in functional analysis, linear operators help to analyse function spaces. Their applications range from the study of integral and differential equations to different tasks in the field of signal processing. In quantum mechanics, linear operators describe the evolution of quantum systems through the Schrödinger equation.

In this talk, we present a recently developed framework for proving first-order statements about identities of matrices or linear operators by performing algebraic computations. Our main result is a semi-decision procedure that allows to prove any true operator statement based on a single algebraic computation. We also discuss our software package `operator_gb` [2], which offers functionality for automating such computations, and we present the results of a recent case study [1].

References

1 K. Bernauer, C. Hofstadler, and G. Regensburger. *How to Automatise Proofs of Operator Statements: Moore–Penrose Inverse; A Case Study.* International Workshop on Computer Algebra in Scientific Computing, pp. 39–68, 2023.
2 C. Hofstadler. *Noncommutative Gröbner bases and automated proofs of operator statements.* PhD thesis. Johannes Kepler University Linz, Austria, 2023.

3.8 Proof Complexity for Model Counting

Kaspar Kasche (Friedrich-Schiller-Universität Jena, DE)

License Creative Commons BY 4.0 International license
© Kaspar Kasche
Joint work of Olaf Beyersdorff, Johannes Klaus Fichte, Joachim Giesen, Andreas Goral, Markus Hecher, Tim Hoffmann, Kaspar Kasche, Christoph Staudt
Main reference Olaf Beyersdorff, Johannes Klaus Fichte, Markus Hecher, Tim Hoffmann, Kaspar Kasche: "The Relative Strength of #SAT Proof Systems", in Proc. of the 27th International Conference on Theory and Applications of Satisfiability Testing, SAT 2024, August 21-24, 2024, Pune, India, LIPIcs, Vol. 305, pp. 5:1–5:19, Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2024.
URL https://doi.org/10.4230/LIPICS.SAT.2024.5

The propositional model counting problem #SAT asks to compute the number of satisfying assignments for a given propositional formula. Recently, three #SAT proof systems kcps[1] (knowledge compilation proof system), MICE[2] (model counting induction by claim extension), and CPOG[3] (certified partitioned-operation graphs) have been introduced with the aim to model #SAT solving and enable proof logging for solvers. Prior to this paper, the relations between these proof systems have been unclear and very few proof complexity results are known. We completely determine the simulation order of the three systems, establishing that CPOG simulates both MICE and kcps, while MICE and kcps are exponentially incomparable. This implies that CPOG is strictly stronger than the other two systems.

References

1 Florent Capelli: Knowledge Compilation Languages as Proof Systems. SAT 2019: 90-99
2 Johannes Klaus Fichte, Markus Hecher, Valentin Roland: Proofs for Propositional Model Counting. SAT 2022: 30:1-30:24
3 Randal E. Bryant, Wojciech Nawrocki, Jeremy Avigad, Marijn J. H. Heule: Certified Knowledge Compilation with Application to Verified Model Counting. 6:1-6:20

3.9 The packing chromatic number of the infinite square lattice

Barnaby Martin (Durham University, GB)

License Ⓒ Creative Commons BY 4.0 International license
© Barnaby Martin

The packing chromatic number of a graph is the minimum n so that the graph may be vertex-coloured with n colours so that vertices coloured $i < n + 1$ never appear at distance i or less from one another (thus colour 1 behaves as in a usual proper colouring). We survey results about the packing chromatic number of various infinite lattices and present bounds for the infinite square lattice that have been found by various methods, including SAT-solving, which were recently perfected to the answer 15.

3.10 Supercritical and Robust Trade-offs for Resolution Depth Versus Width and Weisfeiler-Leman

Jakob Nordström (University of Copenhagen, DK & Lund University, SE)

License Ⓒ Creative Commons BY 4.0 International license
© Jakob Nordström
Joint work of Duri Andrea Janett, Jakob Nordström, Shuo Pang

We prove robust supercritical trade-off results for depth versus width in resolution and for the Weisfeiler-Leman algorithm, where optimizing one complexity measure even approximately causes worse than brute-force worst-case behaviour for the other measure. These are the first trade-offs in these settings that are truly supercritical measured not only in the number of variables but in the size of the input.

3.11 AVATAR: 10 years on

Michael Rawson (TU Wien, AT)

License Creative Commons BY 4.0 International license
© Michael Rawson
Joint work of Michael Rawson, Martin Suda

AVATAR is a powerful but poorly-understood component of the Vampire theorem prover, first introduced by Andrei Voronkov circa 2014 [1]. In essence, AVATAR allows a SAT solver to manage the propositional structure of splitting decisions during proof search. I introduce AVATAR, chart its development over the last ten years (such as practical experiments [2] and AVATAR modulo theories [3]), and give some future directions [4].

References

1 Andrei Voronkov. *AVATAR: the architecture for first-order theorem provers.* CAV 2014, Springer.
2 Giles Reger Martin Suda Andrei Voronkov. *Playing with AVATAR.* CADE-25, Springer.
3 Nikolaj Bjorner Giles Reger Martin Suda Andrei Voronkov. *AVATAR modulo theories.* GCAI 2016.
4 Sólrún Halla Einarsdóttir. *Lemma Discovery and Strategies for Automated Induction.* IJCAR 2024, Springer.

3.12 Open problems in interference and proofs for SAT solving

Adrian Rebola-Pardo (TU Wien, AT & Johannes Kepler Universität Linz, AT)

License Creative Commons BY 4.0 International license
© Adrian Rebola-Pardo
Joint work of Adrian Rebola-Pardo, Georg Weissenbacher
Main reference Adrián Rebola-Pardo: "Even Shorter Proofs Without New Variables", in Proc. of the 26th International Conference on Theory and Applications of Satisfiability Testing, SAT 2023, July 4-8, 2023, Alghero, Italy, LIPIcs, Vol. 271, pp. 22:1–22:20, Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2023.
URL https://doi.org/10.4230/LIPICS.SAT.2023.22

Enormous progress has been made over the last decade in proofs for SAT solving. The combination of satisfiability-preserving inferences, deletion instructions and clausal proofs allowed the development of effective and compact proof formats, such as DRAT. Further advancements include extended inference power, featured in DPR and (W)SR proofs, as well as hinted proofs that can be checked with verified tools, like LRAT and FRAT.

This fast evolution has left some problems unsolved, and some paths unexplored. In this talk I will argue that these gaps in our understanding and development of proofs for SAT are relevant for certification, and for SAT solving itself. In my talk I will focus on (at most, depending on time restrictions) four of these problems.

First, the proliferation of proof systems has created a cornucopia of different checkers and formats with slightly different properties, capabilities and even semantics. Some approaches are to every extent superior, yet their implementation is still spotty; some approaches have been accepted at face value without considering drawbacks or alternatives. We will review some design decisions in proof systems that may be worth revisiting.

Second, deletion instructions, which first appeared in DRUP as a performance-related feature, have quietly become more relevant, and more problematic. CDCL-based SAT proofs are dominated by one specific flavor of deletion, which I call linear deletion. However, this is not the case in adjacent fields. Both VeriPB and WSR depart from this idea, and in doing so they enable new paths in proofs and in reasoning.

Third, satisfiability-preserving inferences have recently come to be understood as reasoning without loss of generality. This enabled very powerful proof systems, but in practice these features are underutilized. In the last couple of years, however, the interest on these techniques has rekindled. I will review what is new, and what theoretical and practical problems still remain to be solved.

Finally, I will explain how well (or not) these advancements extend to other solving paradigms, including non-CNF SAT solving, (D)QBF and model checking, and what roadblocks exist that need further research.

3.13 PPSZ is better than you think

Dominik Alban Scheder (TU Chemnitz, DE)

License Ⓒ Creative Commons BY 4.0 International license
© Dominik Alban Scheder
Main reference Dominik Scheder: "PPSZ is better than you think", TheoretiCS, Vol. 3, 2024.
URL https://doi.org/10.46298/THEORETICS.24.5

PPSZ, for long time the fastest known algorithm for k-SAT, works by going through the variables of the input formula in random order; each variable is then set randomly to 0 or 1, unless the correct value can be inferred by an efficiently implementable rule (like small-width resolution; or being implied by a small set of clauses).

We show that PPSZ performs exponentially better than previously known, for all $k \geq 3$. We achieve this through an improved analysis and without any change to the algorithm itself. The core idea is to pretend that PPSZ does not process the variables in uniformly random order, but according to a carefully designed distribution. We write "pretend" since this can be done while running the original algorithm, which does use a uniformly random order.

3.14 Tutorial: First-Order Automated Theorem Proving

Stephan Schulz (Duale Hochschule Baden-Württemberg – Stuttgart, DE)

License Ⓒ Creative Commons BY 4.0 International license
© Stephan Schulz

I provide a short overview of modern first-order theorem proving, discussing the overall structure of a prover, including clausification and refutation core. For the latter part, we discuss saturation up to redundancy in the superposition setting. We also discuss implementation and search control.

3.15 QBF Solving and Proof Complexity

Friedrich Slivovsky (University of Liverpool, GB)

License Creative Commons BY 4.0 International license
© Friedrich Slivovsky

Quantified Boolean Formulas (QBF) extend propositional formulas with quantifiers ranging over truth values. Satisfiability testing of QBFs is PSPACE-complete, and they can succinctly encode many problems arising in verification and synthesis. This talk provides an introduction to the main paradigms in QBF solving, Quantified CDCL and Expansion, and their corresponding proof systems. It also gives an overview of QBF proof complexity, highlighting the unique role of strategy extraction.

3.16 Polynomial Calculus for QBF

Luc Spachmann (Friedrich-Schiller-Universität Jena, DE)

License Creative Commons BY 4.0 International license
© Luc Spachmann

Joint work of Olaf Beyersdorff, Kasper Kasche, Luc Nicolas Spachmann
Main reference Olaf Beyersdorff, Tim Hoffmann, Kaspar Kasche, Luc Nicolas Spachmann: "Polynomial Calculus for Quantified Boolean Logic: Lower Bounds Through Circuits and Degree", in Proc. of the 49th International Symposium on Mathematical Foundations of Computer Science, MFCS 2024, August 26-30, 2024, Bratislava, Slovakia, LIPIcs, Vol. 306, pp. 27:1–27:15, Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2024.
URL https://doi.org/10.4230/LIPICS.MFCS.2024.27

We initiate an in-depth proof-complexity analysis of polynomial calculus (Q-PC) for Quantified Boolean Formulas (QBF). In the course of this we establish a tight proof-size characterisation of Q-PC in terms of a suitable circuit model (polynomial decision lists). Using this correspondence we show a size-degree relation for Q-PC, similar in spirit, yet different from the classic size-degree formula for propositional PC by Impagliazzo, Pudlák and Sgall (1999). We use the circuit characterisation together with the size-degree relation to obtain various new lower bounds on proof size in Q-PC. This leads to incomparability results for Q-PC systems over different fields.

3.17 First-Order Finite Model Finding via SAT

Martin Suda (Czech Technical University – Prague, CZ)

License Creative Commons BY 4.0 International license
© Martin Suda

Main reference Koen Claessen: "New Techniques that Improve MACE-style Finite Model Finding", 2007.
URL https://api.semanticscholar.org/CorpusID:15694927

I will present how the search for finite models in first-order logic is typically translated into a sequence of SAT formulas, what kinds of symmetry breaking can be used to make these formulas easier to tackle and what extra challenges the multi-sorted setting brings.

3.18 SAT modulo Symmetries – an update

Stefan Szeider (TU Wien, AT) and Tomáš Peitl (TU Wien, AT)

License Creative Commons BY 4.0 International license
© Stefan Szeider and Tomáš Peitl
Joint work of Mikoláš Janota, Markus Kirchweger, Tomáš Peitl, Stefan Szeider
Main reference Mikoláš Janota, Markus Kirchweger, Tomáš Peitl, Stefan Szeider: "Breaking Symmetries in Quantified Graph Search: A Comparative Study", 2025.
URL https://arxiv.org/abs/2502.15078

SAT modulo Symmetries (SMS) is a framework for the exhaustive isomorph-free generation of combinatorial objects with a prescribed property. SMS relies on the tight integration of a CDCL SAT solver with a custom dynamic symmetry-breaking algorithm that iteratively refines an ordered partition of the generated object's elements. This talk will discuss the basic concepts of SMS, some applications, and recent extensions to graph properties specified as general quantified Boolean formulas (QBF). In the second part of the talk, Tomáš Peitl will give a live demo of SMS and its QBF extension.

3.19 SLIM for MaxSAT

Stefan Szeider (TU Wien, AT)

License Creative Commons BY 4.0 International license
© Stefan Szeider
Joint work of André Schidler, Stefan Szeider
Main reference André Schidler, Stefan Szeider: "Structure-Guided Local Improvement for Maximum Satisfiability", in Proc. of the 30th International Conference on Principles and Practice of Constraint Programming, CP 2024, September 2-6, 2024, Girona, Spain, LIPIcs, Vol. 307, pp. 26:1–26:23, Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2024.
URL https://doi.org/10.4230/LIPICS.CP.2024.26

The enhanced performance of today's MaxSAT solvers has elevated their appeal for many large-scale applications, notably in software analysis and computer-aided design. Our research delves into refining anytime MaxSAT solving by repeatedly identifying and solving with an exact solver smaller subinstances that are chosen based on the graphical structure of the instance. We investigate various strategies to pinpoint these subinstances. This structure-guided selection of subinstances provides an exact solver with a high potential for improving the current solution. Our exhaustive experimental analyses contrast our methodology as instantiated in our tool MaxSLIM with previous studies and benchmark it against leading-edge MaxSAT solvers.

3.20 Resolution height and a candidate formula hard for CDCL without restarts

Neil Thapen (The Czech Academy of Sciences – Prague, CZ)

License Creative Commons BY 4.0 International license
© Neil Thapen
Joint work of Sam Buss, Neil Thapen
Main reference Sam Buss, Neil Thapen: "A Simple Supercritical Tradeoff between Size and Height in Resolution", Electron. Colloquium Comput. Complex., Vol. TR24-001, 2024.
URL https://eccc.weizmann.ac.il/report/2024/001

We describe a family of CNFs in n variables which have small resolution refutations but are such that any small refutation must have height larger than n (even exponential in n), where the height of a refutation is the length of the longest path in it (a similar result appeared in [1]). Small refutations of our formulas are thus highly irregular, containing paths querying the same variable many times. This makes it a plausible candidate to separate resolution from pool resolution, which amounts to separating CDCL with restarts from CDCL without. We are not able to show this, but in the other direction we show that a simpler version of our formula, with a similar irregularity property, does have polynomial size pool resolution refutations.

References

1 Noah Fleming, Toniann Pitassi, Robert Robere: Extremely Deep Proofs. ITCS 2022: 70:1-70:23

3.21 An introduction to Satisfiability Modulo Theories

Cesare Tinelli (University of Iowa – Iowa City, US)

License Creative Commons BY 4.0 International license
© Cesare Tinelli

The talk provides an overview of Satisfiability Modulo Theories (SMT), a subfield of automated reasoning that combines uniform methods to reason about formulas in first-order logic with specialized methods to reason about various data types of interest in computer science such as integer and real numbers, bit vectors, strings, finite sets, lists, and so on. The talk is in two parts: Part I focuses on motivation, functionality and applications. Part II describes at an abstract level, in terms of satisfiability proof systems, major approaches to build SMT solvers by combining SAT solvers with specialized solvers for individual theories of various data types.

3.22 Mechanizing the Splitting Framework

Sophie Tourret (INRIA Nancy – Grand Est, FR)

License Creative Commons BY 4.0 International license
© Sophie Tourret
Joint work of Sophie Tourret, Ghilaien Bergeron, Florent Krasnopol

In this talk, I presented the current state of the Isabelle/HOL mechanization efforts that I am leading on the "splitting framework", that makes it possible to use propositional models and a SAT solver to guide the inferences of a saturation-based calculus, as is done in AVATAR. The Isabelle/HOL results do not cover AVATAR yet, but a simpler form of splitting. In ongoing work, we cover "splitting without backtracking" over resolution in FOL. The work was still ongoing at the time of the seminar and I explained where we stood then and why.

3.23 SAT and Proof Complexity

Marc Vinyals (University of Auckland, NZ)

License Creative Commons BY 4.0 International license
© Marc Vinyals

We discuss connections between SAT solvers and proof complexity: how we can analyse the CDCL algorithm thanks to resolution, and how stronger proof systems can guide new developments.

3.24 The Complexity of Enumerating Satisfying Assignments

Heribert Vollmer (Leibniz Universität Hannover, DE)

License Creative Commons BY 4.0 International license
© Heribert Vollmer
Joint work of Nadia Creignou, Arnaud Durand, Heribert Vollmer, Markus Kröll, Reinhard Pichler, Sebastian Skritek
Main reference Nadia Creignou, Arnaud Durand, Heribert Vollmer: "Enumeration Classes Defined by Circuits", in Proc. of the 47th International Symposium on Mathematical Foundations of Computer Science, MFCS 2022, August 22-26, 2022, Vienna, Austria, LIPIcs, Vol. 241, pp. 38:1–38:14, Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2022.
URL https://doi.org/10.4230/LIPICS.MFCS.2022.38
Main reference Nadia Creignou, Markus Kröll, Reinhard Pichler, Sebastian Skritek, Heribert Vollmer: "A complexity theory for hard enumeration problems", Discret. Appl. Math., Vol. 268, pp. 191–209, 2019.
URL https://doi.org/10.1016/J.DAM.2019.02.025

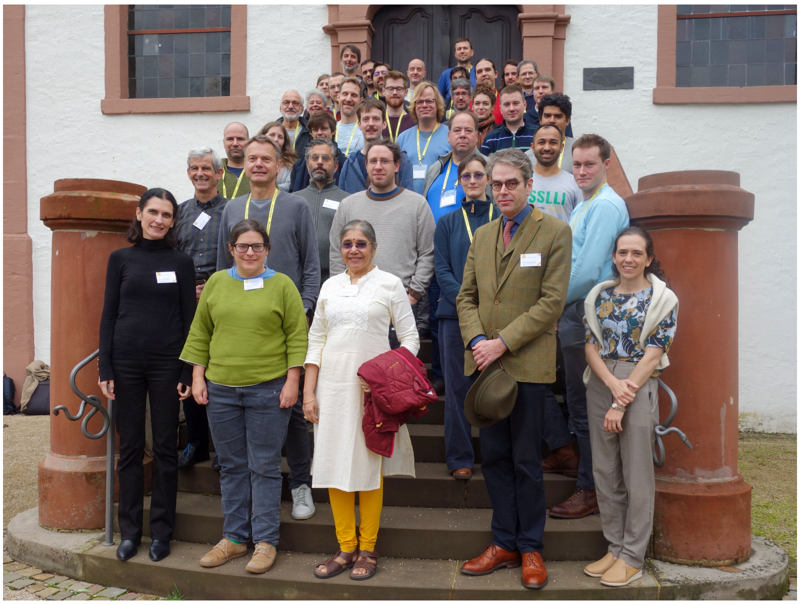
We study the algorithmic problem to enumerate all satisfying assignments of a given propositional formula. For formula classes with a polynomial-time decision problem, this can be done within the class DelayP, introduced by Johnson, Papadimitriou and Yannakakis in 1988 and regarded since then as a reasonable notion of "efficient" enumeration. We will present two results:

1. We generalize the class DelayP to a hierarchy analogous to the polynomial-time hierarchy of decision problems and show that Enum-SAT is complete in the Σ_1-level of this hierarchy under some form of enumeration Turing reductions.
2. We define a hierarchy of very efficiently enumerable problems within DelayP, based in the Boolean circuit class AC_0, and place the enumeration problems for monotone, IHS, Krom, and affine formulas in lower levels of this hierarchy.

Open remains an exact classification of the problem Enum-Horn-SAT.

Participants

- Albert Atserias
UPC Barcelona Tech, ES
- Olaf Beyersdorff
Friedrich-Schiller-Universität Jena, DE
- Ilario Bonacina
UPC Barcelona Tech, ES
- Florent Capelli
University of Artois/CNRS – Lens, FR
- Leroy Nicholas Chew
TU Wien, AT
- Anupam Das
University of Birmingham, GB
- Susanna de Rezende
Lund University, SE
- Katalin Fazekas
TU Wien, AT
- Mathias Fleury
Universität Freiburg, DE
- Pascal Fontaine
University of Liège, BE
- Marlene Gründel
Friedrich-Schiller-Universität Jena, DE
- Clemens Hofstadler
Universität Kassel, DE
- Kaspar Kasche
Friedrich-Schiller-Universität Jena, DE
- Phokion G. Kolaitis
University of California – Santa Cruz, US
- Wietze Koops
Lund University, SE & University of Copenhagen, DK
- Konstantin Korovin
University of Manchester, GB
- Laura Kovács
TU Wien, AT
- Massimo Lauria
Sapienza University of Rome, IT
- Meena Mahajan
The Institute of Mathematical Sciences – Chennai, IN
- Barnaby Martin
Durham University, GB
- Stefan Mengel
CNRS, CRIL – Lens, FR
- Claudia Nalon
University of Brasília, BR
- Jakob Nordström
University of Copenhagen, DK & Lund University, SE
- Tomáš Peitl
TU Wien, AT
- Florian Pollitt
Universität Freiburg, DE
- Michael Rawson
TU Wien, AT
- Adrian Rebola-Pardo
TU Wien, AT & Johannes Kepler Universität Linz, AT
- Rahul Santhanam
University of Oxford, GB
- Dominik Alban Scheder
TU Chemnitz, DE
- Tanja Schindler
Universität Basel, CH
- Stephan Schulz
Duale Hochschule Baden-Württemberg – Stuttgart, DE
- Martina Seidl
Johannes Kepler Universität Linz, AT
- Friedrich Slivovsky
University of Liverpool, GB
- Luc Spachmann
Friedrich-Schiller-Universität Jena, DE
- Martin Suda
Czech Technical University – Prague, CZ
- Stefan Szeider
TU Wien, AT
- Neil Thapen
The Czech Academy of Sciences – Prague, CZ
- Cesare Tinelli
University of Iowa – Iowa City, US
- Jacobo Torán
Universität Ulm, DE
- Sophie Tourret
INRIA Nancy – Grand Est, FR
- Marc Vinyals
University of Auckland, NZ
- Heribert Vollmer
Leibniz Universität Hannover, DE
- Andrei Voronkov
University of Manchester, GB



Report from Dagstuhl Seminar 24431

Automated Programming and Program Repair

Claire Le Goues*[1], Michael Pradel*[2], Abhik Roychoudhury*[3], and Shin Hwei Tan*[4]

1 **Carnegie Mellon University – Pittsburgh, US.** clegoues@cs.cmu.edu
2 **Universität Stuttgart, DE.** michael@binaervarianz.de
3 **National University of Singapore, SG.** abhik.roychoudhury@gmail.com
4 **Concordia University – Montreal, CA.** shinhwei.tan@concordia.ca

Abstract

The Dagstuhl Seminar 24431 on "Automated Programming and Program Repair" brought together 33 researchers from academia and industry to explore the intersection of automated code generation and program repair. Over five days (October 21–25, 2024), participants discussed advances in large language models (LLMs) for code generation, the role of automated program repair in improving generated code, and challenges in deploying these technologies in real-world software development. The seminar featured over 20 talks and three panel discussions on topics such as benchmarks for LLM-generated code, trust in automated programming, and the broader applications of LLMs beyond coding assistance. Key outcomes included identifying critical challenges in benchmarking, evaluation criteria, and developer adoption of automated repair techniques, fostering future collaborations and actionable research directions in the field.

Seminar October 20–25, 2024 – https://www.dagstuhl.de/24431
2012 ACM Subject Classification Software and its engineering → Automatic programming; Software and its engineering → Software testing and debugging
Keywords and phrases Auto-coding, Large Language Models, Automated Program Repair, Program Synthesis, Trustworthy Software
Digital Object Identifier 10.4230/DagRep.14.10.39

1 Executive Summary

Shin Hwei Tan (Concordia University – Montreal, CA)

License Creative Commons BY 4.0 International license
© Shin Hwei Tan

Automated tools that generate and improve code promise to fundamentally change software development. For example, there is a recent trend towards automated code generation from large language models, as evidenced by the capabilities of Codex/Copilot, ChatGPT, and GPT-4. These models, and other techniques, such as search-based and semantic analysis-based techniques, have the potential to automate significant parts of today's software development process. In particular, there are promising techniques for automated programming and automated program repair. Automated programming refers to techniques that suggest newly written code, e.g., in the form of code completion tools. The capabilities of such tools have increased from moderately successful single-token predictions just a few years ago to predicting entire functions with relatively high accuracy. Techniques for automated programming include large language models that predict code based on natural language specifications of the intended behavior.

* Editor / Organizer

Except where otherwise noted, content of this report is licensed
under a Creative Commons BY 4.0 International license
Automated Programming and Program Repair, *Dagstuhl Reports*, Vol. 14, Issue 10, pp. 39–57
Editors: Claire Le Goues, Michael Pradel, Abhik Roychoudhury, and Shin Hwei Tan
DAGSTUHL REPORTS Dagstuhl Reports
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

Automated program repair refers to a suite of techniques for automated rectification of errors or vulnerabilities in programs. Automated program repair technologies were originally developed for reducing the debugging effort for manually written code. In other words, automated program repair technologies are meant to boost developer productivity in locating and fixing errors in manually written code. Automated programming and automated program repair strongly overlap in terms of their goals and the techniques used to achieve these goals. Both streams of research aim at generating correct source code while having to cope with limited knowledge of the behavior this source code is meant to have. Since formal specifications of correct program behavior are typically not available, both techniques try to infer specifications from various program artifacts, such as large code corpora, past program versions, natural language documentation, or various executions of the program. To address the challenge of predicting likely correct code, both streams of research combine techniques from machine learning, search-based approaches, and semantic code analysis.

Despite these similarities, the subcommunities working on automated programming and automated program repair are only partially aware of each other's most recent techniques. This seminar set out to explore the intersection of these two fields in order to foster collaborations between them. For example, to discuss recent work and potential future work in the following directions: (1) Apply program repair to fix code generated by code completion models. The code generated by large language models often leaves significant room for improvement in terms of correctness, thereby raising the question whether automated program repair techniques can be used for last-mile improvement of code that was automatically generated by large language models. (2) Apply the generate-and-validate paradigm from program repair to the code completion problem. For example, such techniques can repeatedly generate code completion candidates and validate them by running test suites. (3) Apply language model-based code generators to the program repair problem. Once the location of a bug has been (heuristically) determined, large language models can predict candidate code snippets for replacing the incorrect code. (4) Use the ability of large language models to infer the intended behavior of code from natural language information embedding in the code. For example, we plan to discuss techniques that specify the intended behavior in the form of assertions or test cases, which can then guide automated program repair. (5) In addition to predicting (fixed) code, generate evidence that the final code is trustworthy. Such evidence may take the form of tests generated along with the code, or other certificates obtained from formal reasoning.

Thus, to discuss topics at the intersection of automated programming and program repair, we had Dagstuhl Seminar 24431. In a five-day seminar with 33 participants from both academia and industries (e.g., Microsoft and Google), we held a series of talks and three panel discussions. The seminar concluded with more than 20 talks and three panel discussions. Overall, the seminar stimulated quite a few discussions where researchers initiated some future research directions and potential international collaborations.

Before the seminar, all participants had received an invitation to give a talk of a flexible duration (i.e., lightning update that is around 5 minutes, short talk that is around 10 minutes or long talk that is around 25 minutes). More than 20 participants replied positively to the invitation, resulting in a great variety of talks given by many participants. The first day of the seminar (i.e., October 21, 2025) started with an introduction by the organizers and a self-introduction by all the participants. Then, a few short talks and longer talks (more than 25 minutes) were given by participants. The first day ended with a panel discussion on "Benchmarks for LLM Code Generation". On October 22, 2025, there were several talks followed by a panel discussion on "LLM-beyond just coding-assistance". On October 23, 2025, several talks took place in the morning, followed by an excursion to Mettlach and Villa Borg after lunch. On October 24, 2025, a few inspiring talks took place, followed by a panel discussion on "Obstacles for deploying program repair techniques".

Overall, the seminar has received very positive feedback from the participants both personally and formally (via email). Notably, one participant sent an email to one of the organizers saying that " It was my best Dagstuhl Seminar last October, and I really appreciate your organizing of the seminar once again", demonstrating that the seminar has been quite successful in leaving a good impression in comparison to other Dagstuhl Seminars that the participants have attended. Meanwhile, a few participants have complimented Dagstuhl on the diversity of the social events held (e.g., excursion, the treetop walk, and sauna), and the babysitting services provided for participants attending the seminar with young children.

In terms of collaborations, there are a few actionable topics for collaborations that have been discussed. An opinion piece of AI Software Engineer titled "AI Software Engineer: Programming with Trust" is now available.[1] Another potential collaboration is a critical review on benchmarks crafted by AI communities (i.e., SWE-Bench). Meanwhile, AutoCodeRover (presented by one of the organizers in Dagstuhl), which was an NUS spinoff, has on February 19, 2025, been officially acquired by SonarSource, a leader in code quality via its static analysis solutions.

The seminar focused on the following key themes:

- Topics at the intersection of automated programming and automated program repair, analyzing progress in both fields.
- Understanding common mistakes in automatically generated code.
- Discussing the theme of **"Trusted Automated Programming"**, which focuses on:
  - How automatically generated code can be made more trustworthy.
  - How to generate evidence that improvements to auto-generated code maintain trustworthiness.
  - How to decide, based on such evidence, when to incorporate automatically generated code into an existing software project with a stable code-base.
- Important challenges in automated program repair and automated programming in general.
- Using large language models (LLMs) beyond just coding assistance.
- Obstacles in deploying program repair techniques in real-world settings.

The seminar also identified several critical challenges:

1. The problem of curating widely accepted benchmarks for code generation that serve both the Software Engineering and AI communities.
2. The problem of designing evaluation criteria that effectively assess the quality and reliability of auto-generated code.
3. The challenges and opportunities in applying LLM-based techniques beyond traditional automated program repair (APR).
4. The obstacles in training developers to effectively use program repair techniques in real-world software development.

[1] https://arxiv.org/abs/2502.13767

2 Table of Contents

3 Overview of Talks

3.1 Automatic Semantic Augmentation of Language Model Prompts

Earl T. Barr (University College London, GB)

License Creative Commons BY 4.0 International license
© Earl T. Barr
Joint work of Toufique Ahmed, Kunal Suresh Pai, Premkumar T. Devanbu, Earl T. Barr
Main reference Toufique Ahmed, Kunal Suresh Pai, Premkumar T. Devanbu, Earl T. Barr: "Automatic Semantic Augmentation of Language Model Prompts (for Code Summarization)", in Proc. of the 46th IEEE/ACM International Conference on Software Engineering, ICSE 2024, Lisbon, Portugal, April 14-20, 2024, pp. 220:1–220:13, ACM, 2024.
URL https://doi.org/10.1145/3597503.3639183

Researchers are still learning how to best "program" LLMs via prompt engineering. We start with the intuition that developers tend to consciously and unconsciously collect semantics facts, from code, while working. Most are shallow, simple facts arising from a quick read. One might assume that LLMs are implicitly capable of performing simple "code analysis" and extracting such information: but are they, really? If not, could explicitly adding this information help? Our goal here is to investigate this question and evaluate whether automatically augmenting an LLM's prompt with semantic facts explicitly, actually helps. We find that adding semantic facts to the prompt actually does help! This approach improves performance on the code summarization and completion tasks in several different settings suggested by prior work, including for three different Large Language Models. In most cases, we see improvements, as measured by a range of commonly-used metrics.

3.2 RepairAgent: An Autonomous, LLM-Based Agent for Program Repair

Islem Bouzenia (Universität Stuttgart, DE), Premkumar T. Devanbu (University of California – Davis, US), Michael Pradel (Universität Stuttgart, DE)

License Creative Commons BY 4.0 International license
© Islem Bouzenia, Premkumar T. Devanbu, and Michael Pradel

Automated program repair has emerged as a powerful technique to mitigate the impact of software bugs on system reliability and user experience. This paper introduces RepairAgent, the first work to address the program repair challenge through an autonomous agent based on a large language model (LLM). Unlike existing deep learning-based approaches, which prompt a model with a fixed prompt or in a fixed feedback loop, our work treats the LLM as an agent capable of autonomously planning and executing actions to fix bugs by invoking suitable tools. RepairAgent freely interleaves gathering information about the bug, gathering repair ingredients, and validating fixes, while deciding which tools to invoke based on the gathered information and feedback from previous fix attempts. Key contributions that enable RepairAgent include a set of tools that are useful for program repair, a dynamically updated prompt format that allows the LLM to interact with these tools, and a finite state machine that guides the agent in invoking the tools. Our evaluation on the popular Defects4J dataset demonstrates RepairAgent's effectiveness in autonomously repairing 164 bugs, including 39 bugs not fixed by prior techniques. Interacting with the LLM imposes an average cost of 270,000 tokens per bug, which, under the current pricing of OpenAI's GPT-3.5 model, translates to 14 cents per bug. To the best of our knowledge, this work is the first to present an autonomous, LLM-based agent for program repair, paving the way for future agent-based techniques in software engineering.

3.3 Ya'll are APRing the wrong thing

Yuriy Brun (University of Massachusetts Amherst, US)

License Creative Commons BY 4.0 International license
© Yuriy Brun
Joint work of Yuriy Brun, Emily First, Arjun Guha, Zhanna Kaufman, Alex Sanchez-Stern, Abhishek Varghese, Saketh Kasibatla, Arpan Agrawal, Tom Reichel, Shizhuo Zhang, Timothy Zhou, Dylan Zhang, Sorin Lerner, Talia Ringer
Main reference Alex Sanchez-Stern, Abhishek Varghese, Zhanna Kaufman, Dylan Zhang, Talia Ringer, Yuriy Brun: "QEDCartographer: Automating Formal Verification Using Reward-Free Reinforcement Learning", in Proc. of the 47th International Conference on Software Engineering (ICSE), pp. 405–418, 2025.
URL https://doi.org/10.1109/ICSE55347.2025.00033

Automated program repair (APR) has advanced program synthesis technology but is limited by the problem of weak specifications. The domain of proof synthesis for formal verification has many of the same challenges as APR but has access to a strong oracle – the theorem prover – to determine proof correctness. This oracle helps overcome the overfitting problem for proof synthesis. We have developed several tools for synthesizing proofs from scratch using natural language processing, including TacTok, Passport, Diva, Baldur, and Cobblestone.

References

1. First, Emily and Brun, Yuriy and Guha, Arjun, TacTok: Semantics-aware proof synthesis, OOPSLA 2020
2. Sanchez-Stern, Alex and First, Emily and Zhou, Timothy and Kaufman, Zhanna and Brun, Yuriy and Ringer, Talia, Passport: Improving automated formal verification using identifiers, TOPLAS23
3. Sanchez-Stern, Alex and Varghese, Abhishek and Kaufman, Zhanna and Zhang, Dylan and Ringer, Talia and Brun, Yuriy, QEDCartographer: Automating formal verification using reward-free reinforcement learning, ICSE 25, To Appear
4. First, Emily and Brun, Yuriy, Diversity-driven automated formal verification, ICSE 2022
5. First, Emily and Rabe, Markus N and Ringer, Talia and Brun, Yuriy, Baldur: Whole-proof generation and repair with large language models, FSE 2023
6. First, Emily and Brun, Yuriy, Diversity-driven automated formal verification, ICSE 2022
7. Brun, Yuriy, Demo for Proofster: Automated Formal Verification, https://www.youtube.com/watch?v=xQAi66lRfwI

3.4 Reflections on Automated Programming / Input Repair: Can LLMs help?

Cristian Cadar (Imperial College London, GB)

License Creative Commons BY 4.0 International license
© Cristian Cadar
Joint work of Tomasz Kuchta, Cristian Cadar, Miguel Castro, Manuel Costa
Main reference Tomasz Kuchta, Cristian Cadar, Miguel Castro, Manuel Costa: "Docovery: toward generic automatic document recovery", in Proc. of the ACM/IEEE International Conference on Automated Software Engineering, ASE '14, Vasteras, Sweden – September 15 – 19, 2014, pp. 563–574, ACM, 2014.
URL https://doi.org/10.1145/2642937.2643004

In my talk, I first reflect on the challenges of integrating automated programming into software projects maintained by humans. I then introduce the problem of input repair, particularly in the context of complex documents and document processors, with the goal of discussing the applicability of modern LLM-based code repair techniques.

3.5 Fault Localization (LLM + Heuristics): initial results

Celso G. Camilo-Junior (Federal University of Goiás, BR)

License Creative Commons BY 4.0 International license
© Celso G. Camilo-Junior

The task of Fault Localization is an important part of the software debugging process, as it requires a lot of effort and investment. Additionally, it directly impacts the quality of the produced code.

With the emergence of LLMs and their potential, there is an opportunity to apply them to Fault Localization as well. However, discovering the best way to integrate known techniques with LLMs is a challenge.

Thus, this presentation shows some initial results on how the use of LLMs for Fault Localization, utilizing a few facts, can be reasonable, and which scenarios still face challenges for applying more complex methods. Moreover, an architecture was presented that aims to enrich the necessary inputs, such as better problem descriptions, for the optimal performance of these tools.

The initial results show that, for some simpler problems, using a generic LLM (GPT-3.5 and Llama 3) with a good human description or an artificial description yields reasonable results. However, in more complex scenarios, the combination of LLM with heuristics (spectrum-based) shows more promising outcomes.

3.6 Reinventing ourselves

Satish Chandra (Google – Mountain View, US)

License Creative Commons BY 4.0 International license
© Satish Chandra

In this talk, I will cover a few related themes. First, I will share my views on how modern LLMs are upending the field of software engineering with capabilities that only a few years ago required creative solutions. Next, I'll describe how at Google we have been working on weaving AI capabilities in developer workflows, how we collect developer data, and how we prioritize our work (based on the a recent blog post https://research.google/blog/ai-in-software-engineering-at-google-progress-and-the-path-ahead/.). I will then change tracks a bit and discuss the importance of evals for software engineering tasks; the more real the better. Finally, I will segue into our preliminary investigation into understanding our own bug database at Google, and how it may compare in distribution to SWEBench. I'll share our preliminary experience with agentic ways of resolving these internal bugs.

3.7 Automated Software A11Y Repair

Chunyang Chen (TU München – Heilbronn, DE)

License Creative Commons BY 4.0 International license
© Chunyang Chen
Main reference Zhe Liu, Chunyang Chen, Junjie Wang, Mengzhuo Chen, Boyu Wu, Xing Che, Dandan Wang, Qing Wang: "Make LLM a Testing Expert: Bringing Human-like Interaction to Mobile GUI Testing via Functionality-aware Decisions", in Proc. of the 46th IEEE/ACM International Conference on Software Engineering, ICSE 2024, Lisbon, Portugal, April 14-20, 2024, pp. 100:1–100:13, ACM, 2024.
URL https://doi.org/10.1145/3597503.3639180

As one of the major infrastructures of society, software must be accessible to everyone, regardless of their physical abilities or socioeconomic status. However, the stark reality is that much software remains inaccessible or challenging to use for individuals with disabilities, creating a significant digital divide. To repair those accessibility issues, I will present our works using machine learning to automatically repair those issues i.e., generating missing labels for interactive icons in the GUI based on computer vision, and generating meaningful hint-text for text blanks in mobile applications with LLM.

3.8 Enhancing Fault Localization with LLM Agents and Self-Reflection

Tse-Hsun Chen (Concordia University – Montreal, CA)

License Creative Commons BY 4.0 International license
© Tse-Hsun Chen
Joint work of Md Nakhla Rafi, Dong Jae Kim, Tse-Hsun Chen, Shaowei Wang
Main reference Md Nakhla Rafi, Dong Jae Kim, Tse-Hsun Chen, Shaowei Wang: "Enhancing Fault Localization Through Ordered Code Analysis with LLM Agents and Self-Reflection", CoRR, Vol. abs/2409.13642, 2024.
URL https://arxiv.org/abs/2409.13642

Locating and fixing software faults is time-consuming and resource-intensive. Traditional methods like Spectrum-Based Fault Localization (SBFL) suffer from low accuracy, while learning-based approaches require large datasets and are computationally expensive. Recent advancements in Large Language Models (LLMs) show promise in improving fault localization, but they still face challenges such as token limits and difficulties with large projects. To address this, we introduce LLM4FL, an LLM-agent-based approach that combines SBFL with a divide-and-conquer strategy. By using multiple LLM agents and prompt chaining, LLM4FL navigates codebases and localizes faults more effectively. In tests using Defects4J, LLM4FL outperformed AutoFL by 19.27% in Top-1 accuracy and surpassed state-of-the-art methods like DeepFL and Grace, without task-specific training. Coverage splitting and method ordering further boosted accuracy by up to 22%.

3.9 Code from LLMs: Use, modify, or discard?

Premkumar T. Devanbu (University of California – Davis, US)

License Creative Commons BY 4.0 International license
© Premkumar T. Devanbu
Joint work of Claudio Spiess, David Gros, Kunal Suresh Pai, Michael Pradel, Md Rafiqul Islam Rabin, Amin Alipour, Susmit Jha, Premkumar T. Devanbu, Toufique Ahmed

Code generated by language models is often flawed. And yet, LLMs can produce a lot of useful code. How is a developer to decide whether a given generated fragrment of code should be used or not? We consider this to be a joint human-AI decision problem, where it vitally important for the AI (LLM) to reveal to the AI some trustworthy indication of the expected likelihood of the code being correct.

References

1 Spiess, Claudio and Gros, David and Pai, Kunal Suresh and Pradel, Michael and Rabin, Md Rafiqul Islam and Alipour, Amin and Jha, Susmit and Devanbu, Prem and Ahmed, Toufique, Calibration and correctness of language models for code, ICSE 2025, *to appear*

3.10 Automated Scientific Debugging with LLMs

Sungmin Kang (KAIST – Daejeon, KR)

License Creative Commons BY 4.0 International license
© Sungmin Kang
Joint work of Sungmin Kang, Bei Chen, Shin Yoo, Jian-Guang Lou
Main reference Sungmin Kang, Bei Chen, Shin Yoo, Jian-Guang Lou: "Explainable Automated Debugging via Large Language Model-driven Scientific Debugging", CoRR, Vol. abs/2304.02195, 2023.
URL https://doi.org/10.48550/ARXIV.2304.02195

Existing work on the application of automated program repair (APR) techniques in industry suggest that providing explanations for automatically generated patches could help developer adoption of such tools. We first define criteria for satisfying explanations of patches, then propose the Automated Scientific Debugging (AutoSD) technique, inspired by human developer debugging frameworks, to uncover facts about the debugging scenario and integrate these facts with hypotheses about what is causing the bug. We present our empirical results demonstrating that AutoSD achieves comparable repair performance with other techniques, and that explanations could help developers assess the correctness of patches generated by AutoSD.

3.11 Proactive Debugging

Dongsun Kim (Kyungpook National University, KR)

License Creative Commons BY 4.0 International license
© Dongsun Kim

Automated program repair (APR) techniques successfully fixed programs bugs, but it still follows the process of classical debugging, which is not quite different from a manual debugging process. Although the techniques are fully or partially automated, we have to wait for a bug report or symptoms, reproduce it, localize it, and fix it. However, it might not be necessary to follow the classical and reactive process. What if we can proactively change the source code first to fix a bug?

This talk addresses a series of our initial work on proactive debugging applied to specific bug types, such as memory leaks in single-page web applications and blocking-call bugs in reactive applications. In particular, the talk demonstrates the effectiveness of pattern-based patch generation as the first step of proactive debugging. To show the effectiveness of proactive debugging, the work conducts a series of experiments. The results show that the proactive process significantly helps the developers address bugs in real software systems. The talk illustrates further challenges to improve the activities for proactive debugging, such as fully automated evidence collection and patch generation techniques.

3.12 Energy Consumption of Automated Program Repair

Matías Martínez (UPC Barcelona Tech, ES)

License Ⓒ Creative Commons BY 4.0 International license
© Matías Martínez
Joint work of Matías Martinez, Silverio Martínez-Fernández, Xavier Franch
Main reference Matias Martinez, Silverio Martínez-Fernández, Xavier Franch: "Energy Consumption of Automated Program Repair", in Proc. of the 2024 IEEE/ACM 46th International Conference on Software Engineering: Companion Proceedings, ICSE Companion 2024, Lisbon, Portugal, April 14-20, 2024, pp. 358–359, ACM, 2024.
URL https://doi.org/10.1145/3639478.3643114

Automated program repair (APR) aims to automatize the process of repairing software bugs in order to reduce the cost of maintaining software programs. Moreover, the success (given by the accuracy metric) of APR approaches has increased in recent years. However, no previous work has considered the energy impact of repairing bugs automatically using APR. The field of green software research aims to measure the energy consumption required to develop, maintain, and use software products. This paper combines, for the first time, the APR and Green software research fields. We have as main goal to define the foundation for measuring the energy consumption of the APR activity. We measure the energy consumption of ten traditional program repair tools for Java and ten fine-tuned Large-Language Models (LLM) on source code trying to repair real bugs from Defects4J, a set of real buggy programs. The initial results from this experiment show the existing trade-off between energy consumption and the ability to correctly repair bugs: Some APR tools are capable of achieving higher accuracy by spending less energy than other tools.

3.13 Fact Selection Problem in LLM Based Program Repair

Nikhil Parasaram (University College London, GB)

License Ⓒ Creative Commons BY 4.0 International license
© Nikhil Parasaram
Joint work of Nikhil Parasaram, Huijie Yan, Boyu Yang, Zineb Flahy, Abriele Qudsi, Damian Ziaber, Earl T. Barr, Sergey Mechtaev
Main reference Nikhil Parasaram, Huijie Yan, Boyu Yang, Zineb Flahy, Abriele Qudsi, Damian Ziaber, Earl T. Barr, Sergey Mechtaev: "The Fact Selection Problem in LLM-Based Program Repair", CoRR, Vol. abs/2404.05520, 2024.
URL https://doi.org/10.48550/ARXIV.2404.05520

Recent research shows that including bug-related facts (categories of the information), such as stack traces and GitHub issues, enhances LLMs' bug-fixing abilities. To determine the optimal facts and quantity, we studied over 19K prompts with various fact combinations to repair 314 BugsInPy benchmark bugs. Each fact, from syntactic details to unexplored semantic information like angelic values, proved beneficial in fixing specific bugs. Notably, effectiveness is non-monotonic: too many facts reduce performance. We defined the fact selection problem to find the optimal facts for each task, developing MANIPLE, a model that selects facts tailored to a given bug, outperforming existing methods and repairing 17% more bugs than the best alternative.

3.14 ChangeGuard: Validating Code Changes via Pairwise Learning-Guided Execution

Michael Pradel (Universität Stuttgart, DE)

License Creative Commons BY 4.0 International license
© Michael Pradel
Joint work of Lars Gröninger, Beatriz Souza, Michael Pradel
Main reference Lars Gröninger, Beatriz Souza, Michael Pradel: "ChangeGuard: Validating Code Changes via Pairwise Learning-Guided Execution", CoRR, Vol. abs/2410.16092, 2024.
URL https://doi.org/10.48550/ARXIV.2410.16092

Code changes are an integral part of the software development process. Many code changes are meant to improve the code without changing its functional behavior, e.g., refactorings and performance improvements. Unfortunately, validating whether a code change preserves the behavior is non-trivial, particularly when the code change is performed deep inside a complex project. This talk presents ChangeGuard, an approach that uses learning-guided execution to compare the runtime behavior of a modified function. The approach is enabled by the novel concept of pairwise learning-guided execution and by a set of techniques that improve the robustness and coverage of the state-of-the-art learning-guided execution technique. Our evaluation applies ChangeGuard to a dataset of 224 manually annotated code changes from popular Python open-source projects and to three datasets of code changes obtained by applying automated code transformations. Our results show that the approach identifies semantics-changing code changes with a precision of 77.1% and a recall of 69.5´%, and that it detects unexpected behavioral changes introduced by automatic code refactoring tools. In contrast, the existing regression tests of the analyzed projects miss the vast majority of semantics-changing code changes, with a recall of only 7.6%. We envision our approach being useful for detecting unintended behavioral changes early in the development process and for improving the quality of automated code transformations.

3.15 Testing, Testing, 1-2-3: Test generation with LLMs

Nikitha Rao (Carnegie Mellon University – Pittsburgh, US)

License Creative Commons BY 4.0 International license
© Nikitha Rao
Main reference Nikitha Rao, Kush Jain, Uri Alon, Claire Le Goues, Vincent J. Hellendoorn: "CAT-LM Training Language Models on Aligned Code And Tests", in Proc. of the 38th IEEE/ACM International Conference on Automated Software Engineering, ASE 2023, Luxembourg, September 11-15, 2023, pp. 409–420, IEEE, 2023.
URL https://doi.org/10.1109/ASE56229.2023.00193
Main reference Nikitha Rao, Elizabeth Gilbert, Tahina Ramananandro, Nikhil Swamy, Claire Le Goues, Sarah Fakhoury: "DiffSpec: Differential Testing with LLMs using Natural Language Specifications and Code Artifacts", CoRR, Vol. abs/2410.04249, 2024.
URL https://doi.org/10.48550/ARXIV.2410.04249

LLMs can generate code that is highly similar to that written by humans. However, current models are trained to generate each file separately, as is standard practice in natural language processing. They thus fail to generate meaningful tests. My work leverages software artifacts like code and natural language specifications to make LLM-based test generation more reliable and useful. This includes (1) CAT-LM, a LLM trained to explicitly consider the mapping between code and test files to improve the quality of tests generated. This not only ensures code correctness, but also optimizes for other software testing metrics such as pass/compile rate and code coverage. (2) DiffSpec, a prompt chaining based framework that leverages artifacts like specification documents, bug reports, and code implementations for generating differential tests using LLMs. We evaluate DiffSpec on eBPF and Wasm, and generated over 500 differentiating tests that found real bugs.

3.16 Assured Automatic Programming via Large Language Models

Abhik Roychoudhury (National University of Singapore, SG), Andreea Costea (National University of Singapore, SG)

License Creative Commons BY 4.0 International license
© Abhik Roychoudhury, Andreea Costea
Joint work of Abhik Roychoudhury, Andreea Costea, Abhishek Kr Singh, Martin Mirchev

With the advent of LLMs in automatic programming, the interest in trusted automatic programming via LLMs increases. Unfortunately, it is difficult to give any guarantees about code generated from LLMs, partly also because a detailed specification of the intended behavior is usually not available. In this talk we show how to alleviate this lack of functionality specifications by aligning automatically generated code via LLMs, automatically generated formal specifications (obtained from natural language using LLMs), as well as tests. The conformance between generated programs, generated specifications, and tests – does not provide absolute guarantees but enhances trust. Establishing such conformance also helps us uncover the likely intended program behavior.

3.17 AutoCodeRover: Autonomous Program Improvement

Abhik Roychoudhury (National University of Singapore, SG)

License Creative Commons BY 4.0 International license
© Abhik Roychoudhury
Joint work of Yuntong Zhang, Haifeng Ruan, Zhiyu Fan, Abhik Roychoudhury
Main reference Yuntong Zhang, Haifeng Ruan, Zhiyu Fan, Abhik Roychoudhury: "AutoCodeRover: Autonomous Program Improvement", in Proc. of the 33rd ACM SIGSOFT International Symposium on Software Testing and Analysis, ISSTA 2024, Vienna, Austria, September 16-20, 2024, pp. 1592–1604, ACM, 2024.
URL https://doi.org/10.1145/3650212.3680384

Researchers have made significant progress in automating the software development process in the past decades. Recent progress in Large Language Models (LLMs) has significantly impacted the development process, where developers can use LLM-based programming assistants to achieve automated coding. Nevertheless, software engineering involves the process of program improvement apart from coding, specifically to enable software maintenance (e.g. bug fixing) and software evolution (e.g. feature additions). We propose an automated approach for solving GitHub issues to autonomously achieve program improvement. In our approach called AutoCodeRover, LLMs are combined with sophisticated code search capabilities, ultimately leading to a program modification or patch. In contrast to recent LLM agent approaches from AI researchers and practitioners, our outlook is more software engineering oriented. We work on a program representation (abstract syntax tree) as opposed to viewing a software project as a mere collection of files. Our code search exploits the program structure in the form of classes/methods to enhance LLM's understanding of the issue's root cause, and effectively retrieve a context via iterative search. The use of spectrum-based fault localization using tests, further sharpens the context, as long as a test-suite is available. Experiments on SWE-bench-lite (300 real-life GitHub issues) show increased efficacy in solving GitHub issues (19% on SWE-bench-lite), which is higher than the efficacy of the recently reported SWE-agent. In addition, AutoCodeRover achieved this efficacy with significantly lower cost (on average, $0.43 USD), compared to other baselines. We posit that our workflow enables autonomous software engineering, where, in future, auto-generated code from LLMs can be autonomously improved.

3.18 RepairBench: Leaderboard of Frontier Models for Program Repair

André Silva (KTH Royal Institute of Technology – Stockholm, SE)

License Creative Commons BY 4.0 International license
© André Silva
Joint work of André Silva, Martin Monperrus
Main reference André Silva, Martin Monperrus: "RepairBench: Leaderboard of Frontier Models for Program Repair", CoRR, Vol. abs/2409.18952, 2024.
URL https://doi.org/10.48550/ARXIV.2409.18952

AI-driven program repair uses AI models to repair buggy software by producing patches. Rapid advancements in AI surely impact state-of-the-art performance of program repair. Yet, grasping this progress requires frequent and standardized evaluations. We propose RepairBench, a novel leaderboard for AI-driven program repair. The key characteristics of RepairBench are: 1) it is execution-based: all patches are compiled and executed against a test suite, 2) it assesses frontier models in a frequent and standardized way. RepairBench leverages two high-quality benchmarks, Defects4J and GitBug-Java, to evaluate frontier models against real-world program repair tasks. We publicly release the evaluation framework of RepairBench. We will update the leaderboard as new frontier models are released.

3.19 Debugging and Fixing Fairness Issues in Machine Learning

Gang (Gary) Tan (Pennsylvania State University – University Park, US)

License Creative Commons BY 4.0 International license
© Gang (Gary) Tan
Joint work of Saeid Tizpaz-Niari, Ashish Kumar, Ashutosh Trivedi, Vishnu Asutosh Dasu
Main reference Saeid Tizpaz-Niari, Ashish Kumar, Gang Tan, Ashutosh Trivedi: "Fairness-aware Configuration of Machine Learning Libraries", in Proc. of the 44th IEEE/ACM 44th International Conference on Software Engineering, ICSE 2022, Pittsburgh, PA, USA, May 25-27, 2022, pp. 909–920, ACM, 2022.
URL https://doi.org/10.1145/3510003.3510202
Main reference Verya Monjezi, Ashutosh Trivedi, Gang Tan, Saeid Tizpaz-Niari: "Information-Theoretic Testing and Debugging of Fairness Defects in Deep Neural Networks", in Proc. of the 45th IEEE/ACM International Conference on Software Engineering, ICSE 2023, Melbourne, Australia, May 14-20, 2023, pp. 1571–1582, IEEE, 2023.
URL https://doi.org/10.1109/ICSE48619.2023.00136
Main reference Vishnu Asutosh Dasu, Ashish Kumar, Saeid Tizpaz-Niari, Gang Tan: "NeuFair: Neural Network Fairness Repair with Dropout", in Proc. of the 33rd ACM SIGSOFT International Symposium on Software Testing and Analysis, ISSTA 2024, Vienna, Austria, September 16-20, 2024, pp. 1541–1553, ACM, 2024.
URL https://doi.org/10.1145/3650212.3680380

In this lightning talk, I will discuss some of our recent efforts of debugging and fixing fairness issues in machine learning and thoughts about possibly reusing some of the techniques for improving trustworthiness of automated code repair.

3.20 More Data or More Domain Knowledge? – For LLM-based Automatic Programming and Repair

Lin Tan (Purdue University – West Lafayette, US)

License Creative Commons BY 4.0 International license
© Lin Tan

Recent techniques use deep learning techniques, including Large Language Models, for automatic programming including automated program repair. An important question is, whether adding more data to train deep learning models or adding domain knowledge to the models is a more promising or effective direction to improve automatic programming. I will discuss existing studies and techniques that answer this question positively or negatively:

References

1 Xie, Danning and Zhang, Zhuo and Jiang, Nan and Xu, Xiangzhe and Tan, Lin and Zhang, Xiangyu, Resym: Harnessing llms to recover variable and data structure symbols from stripped binaries, CCS 2024
2 Nan Jiang and Chengxiao Wang and Kevin Liu and Xiangzhe Xu and Lin Tan and Xiangyu Zhang and Petr Babkin, Nova: Generative Language Models for Assembly Code with Hierarchical Attention and Contrastive Learning, ArXiv
3 Shanchao Liang and Yiran Hu and Nan Jiang and Lin Tan, Can Language Models Replace Programmers? REPOCOD Says 'Not Yet', ArXiv
4 Jiang, Nan and Liu, Kevin and Lutellier, Thibaud and Tan, Lin, Impact of Code Language Models on Automated Program Repair, ICSE 2023
5 Wu, Yi and Jiang, Nan and Pham, Hung Viet and Lutellier, Thibaud and Davis, Jordan and Tan, Lin and Babkin, Petr and Shah, Sameena, How Effective are Neural Networks for Fixing Security Vulnerabilities?, ISSTA 2023

3.21 Neural Code Generation Models with Programming Language Knowledge

Yingfei Xiong (Peking University, CN)

License Creative Commons BY 4.0 International license
© Yingfei Xiong

Programming requires the awareness of programming language knowledge, such as the syntax, the typing system, and the semantics. Such knowledge is not easily learnable from end-to-end training. In this talk I will introduce a series of work that integrates programming language knowledge into neural models.

3.22 Automated Program Repair from Fuzzing Perpsective

Jooyong Yi (Ulsan National Institute of Science and Technology, KR)

License Creative Commons BY 4.0 International license
© Jooyong Yi

APR has been actively researched for the last 15 years, introducing various approaches such as template-based, learning-based, and semantics-based approaches. In particular, the repairability of APR tools has been significantly improved over the years. For example, jGenProg, introduced in 2017, could correctly fix only 2% of the 224 bugs in the Defects4J benchmark, while the latest tool, SRepair, can correctly fix 45% of the 695 bugs in the extended benchmark. This is a remarkable achievement in the field of APR.

While research on repairability should continue, there is another important aspect of APR that has received relatively less attention: APR efficiency. More recent APR tools using template-based or learning-based approaches employ a simplistic method for patch-space exploitation. They simply enumerate the patch candidates in a predefined order based on the suspiciousness scores of the program locations to which the patch candidates are applied. Such an approach is suboptimal because it does not consider the runtime information obtained during the repair process.

In this talk, I present the two recent works we did to improve APR efficiency. We show how the APR scheduling algorithm can be viewed as a multi-armed bandit problem, and how this viewpoint can be used to improve APR efficiency.

The slides for this talk are available at the following link: `https://www.jooyongyi.com/slides/Seminar/Dagstuhl/2024/APR_from_fuzzing_perspective.html`

3.23 Automated Programming in the Era of Large Language Models

Lingming Zhang (University of Illinois – Urbana-Champaign, US)

License Creative Commons BY 4.0 International license
© Lingming Zhang

In recent years, Large Language Models (LLMs), such as GPT-4 and Claude-3.5, have shown impressive performance in various downstream applications. In this talk, I will discuss the potential impact of modern LLMs on automated programming, including both program repair and synthesis (e.g., AlphaRepair, ChatRepair, and Agentless). In addition, I will also talk about our recent work on rigorous testing/benchmarking of LLMs in automated programming (e.g., EvalPlus and SWE-bench Lite-S).

4 Panel discussions

4.1 Benchmarks for LLM Code Generation

Satish Chandra (Google – Mountain View, US), Premkumar T. Devanbu (University of California – Davis, US), Martin Monperrus (KTH Royal Institute of Technology – Stockholm, SE), Gustavo Soares (Microsoft Corporation – Redmond, US), Lin Tan (Purdue University – West Lafayette, US)

License Ⓒ Creative Commons BY 4.0 International license
© Satish Chandra, Premkumar T. Devanbu, Martin Monperrus, Gustavo Soares, and Lin Tan

The panel discussion (organized in the format of fishbowl discussion) was led by an academia researcher (Martin Monperrus) and an industrial researcher (Satish Chandra). Existing evaluation benchmarks (e.g., HumanEval) seem rather simple and does not reflect reality. Recent ones try to address this problem by focusing on more realistic tasks (SWEBench, LiveCodeBench, Long Code Arena). However, there are still a lack of benchmark specifically for code maintenance tasks (e.g., refactoring, repo-level changes, code reviewing, code understanding and reasoning, and performance Improvement). An example of such benchmark is CRQBench: A Benchmark of Code Reasoning Questions. The panel also mentioned possible collaboration on " Critical Review on SWE-Bench". Another possible solution is to have a benchmark track in SE conferences to encourage benchmark curation. The discussion also centered several challenges in benchmarks for LLM Code Generation, including curating good evaluation benchmarks, designing criteria, maintaining and evolving evaluation sets.

4.2 LLM-beyond just coding-assistance

Ahmed E. Hassan (Queen's University – Kingston, CA), Lin Tan (Purdue University – West Lafayette, US), Shin Hwei Tan (Concordia University – Montreal, CA)

License Ⓒ Creative Commons BY 4.0 International license
© Ahmed E. Hassan, Lin Tan, and Shin Hwei Tan

The panel discussion (organized in the format of fishbowl discussion) was led by Ahmed E. Hassan. The discussion focused on discussing the future of automated program Repair (i.e., APR.Next), the challenges, and the opportunities to go from Software Engineering 1.0 to Software Engineering 3.0 where systems based on AI agent have been discussed. Despite the advancement of AI-based techniques, traditional APR techniques still play important roles in: (1) domain specific tasks like Android repair, fixing accessibility issues, (2) improving LLM-generated code by fixing bugs by LLMs using traditional approaches. Other tasks that are worthwhile to explore include: (1) research related to binary repair, and (2) code translation (e.g., C to Rust, translation from COBOL to modern languages).

4.3 Obstacles for deploying program repair techniques

Fernanda Madeiral (VU Amsterdam, NL), Premkumar T. Devanbu (University of California – Davis, US), Gustavo Soares (Microsoft Corporation – Redmond, US)

License Creative Commons BY 4.0 International license
© Fernanda Madeiral, Premkumar T. Devanbu, and Gustavo Soares

The panel discussion (organized in according to the format of a fishbowl discussion) was lead by academic researcher (Fernanda Madeiral) and industrial researcher (Gustavo Soares) centered around the obstacles when deploying automated program repair techniques. Challenges mentioned in the discussion include: (1) answering the questions: When (Inner loop, Outer loop)? How (Autonomous agent, Human-in-the-loop)? What (Realistic benchmark)? (2) the need to train developer how to use the technique (3) should write a test that can be reproduce (e.g., the challenge in vulnerability repair lies in writing test for vulnerability. To solve these obstacles, recent work focused on combining the confidence level of a LLM model (e.g., intrinsic probabilities, i.e., probability of a token given previous tokens, per-token log-probs from the model).

Participants

- Earl T. Barr
University College London, GB
- Islem Bouzenia
Universität Stuttgart, DE
- Yuriy Brun
University of Massachusetts Amherst, US
- Cristian Cadar
Imperial College London, GB
- Celso G. Camilo-Junior
Federal University of Goiás, BR
- Satish Chandra
Google – Mountain View, US
- Chunyang Chen
TU München – Heilbronn, DE
- Tse-Hsun Chen
Concordia University – Montreal, CA
- Zimin Chen
Deutsche Telekom – Bonn, DE
- Andreea Costea
National University of Singapore, SG
- Premkumar T. Devanbu
University of California – Davis, US
- Alexander Frömmgen
Google – München, DE
- Ahmed E. Hassan
Queen's University – Kingston, CA
- Sungmin Kang
KAIST – Daejeon, KR
- Dongsun Kim
Kyungpook National University, KR
- Claire Le Goues
Carnegie Mellon University – Pittsburgh, US
- Yiling Lou
Fudan University – Shanghai, CN
- Fernanda Madeiral
VU Amsterdam, NL
- Matías Martínez
UPC Barcelona Tech, ES
- Martin Monperrus
KTH Royal Institute of Technology – Stockholm, SE
- Nikhil Parasaram
University College London, GB
- Michael Pradel
Universität Stuttgart, DE
- Nikitha Rao
Carnegie Mellon University – Pittsburgh, US
- Abhik Roychoudhury
National University of Singapore, SG
- André Silva
KTH Royal Institute of Technology – Stockholm, SE
- Gustavo Soares
Microsoft Corporation – Redmond, US
- Gang (Gary) Tan
Pennsylvania State University – University Park, US
- Lin Tan
Purdue University – West Lafayette, US
- Shin Hwei Tan
Concordia University – Montreal, CA
- Yingfei Xiong
Peking University, CN
- Jinqiu Yang
Concordia University – Montreal, CA
- He Ye
Carnegie Mellon University – Pittsburgh, US
- Jooyong Yi
Ulsan National Institute of Science and Technology, KR
- Jie Zhang
King's College London, GB
- Lingming Zhang
University of Illinois – Urbana-Champaign, US

Report of Dagstuhl Seminar 24432

Behavioural Metrics and Quantitative Logics

Barbara König*[1], Radu Mardare*[2], Prakash Panangaden*[3], Jurriaan Rot*[4], and Florence Clerc†[5]

1 **Universität Duisburg-Essen, DE.** barbara_koenig@uni-due.de
2 **Heriot-Watt University – Edinburgh, GB.** R.Mardare@hw.ac.uk
3 **McGill University – Montréal, CA.** prakash.panangaden@mcgill.ca
4 **Radboud University – Nijmegen, NL.** jrot@cs.ru.nl
5 **University of Strathclyde – Glasgow, GB.** F.Clerc@hw.ac.uk

Abstract

This report documents the program and the outcome of Dagstuhl Seminar "Behavioural Metrics and Quantitative Logics" (24432).

Behavioural metrics and quantitative logics specify quantitative aspects of systems. A metric measures how far apart two systems are in their behaviour while a quantititative logic evaluates the degree to which a state satisfies a formula. They are often intimately connected via a Hennessy-Milner theorem stating that the distance induced by a quantitative logic coincides with behavioural distance. There are various applications in model-checking, differential privacy, hybrid systems and learning. Several challenges in this area have been identified: studying suitable metrics and their corresponding logics, generalizing to the setting of coalgebras by parameterizing the branching type of the system under consideration, developing methods of quantitative algebraic reasoning, finding efficient methods for computing behavioural methods. This Dagstuhl Seminar provided a forum to researchers working in this area, to discuss the state-of-the-art and further developments, and in particular address applications in various domains.

Seminar October 20–25, 2024 – https://www.dagstuhl.de/24432
2012 ACM Subject Classification Theory of computation → Logic and verification; Theory of computation → Equational logic and rewriting
Keywords and phrases Behavioural metrics, quantitative equational reasoning, quantitative logics
Digital Object Identifier 10.4230/DagRep.14.10.58

1 Executive summary

Barbara König
Radu Mardare
Prakash Panangaden
Jurriaan Rot

License Creative Commons BY 4.0 International license
© Barbara König, Radu Mardare, Prakash Panangaden, and Jurriaan Rot

Behavioural equivalences are central in the analysis and verification of concurrent systems. They come with a very well-developed meta-theory including logical characterisations, efficient algorithms, game-theoretic perspectives, algebraic characterisations, and generalisations to a wide variety of state-based systems. Two processes or system states are behaviourally equivalent if they are indistinguishable from the point of view of an external observer. Behavioural equivalences have for instance been used for comparing a system with its

* Editor / Organizer
† Editorial Assistant / Collector

Except where otherwise noted, content of this report is licensed under a Creative Commons BY 4.0 International license

specification, for the analysis of cryptographic protocols, for the verification of model transformations, for efficient comparison of non-deterministic automata and in programming language semantics.

In recent years, behavioural metrics and quantitative logics specifying quantitative aspects of systems have received considerable attention. Previous approaches are qualitative and address the question "do they match?" whereas metrics address the question "to what degree? how much?". Indeed, a metric measures how far apart two systems are in their behaviour, while a quantitative logic evaluates the degree to which a state satisfies a formula. For quantitative systems such as Markov chains, Markov decision processes and probabilistic automata, a behavioural equivalence is often too strict: small deviations in quantitative information, such as probabilities, can cause two states that intuitively behave very much alike to be inequivalent in a formal sense. Indeed, behavioural equivalence misses *robustness* under small perturbations in such systems. Metrics and quantitative logics provide a more nuanced approach than qualitative behavioural equivalence.

As behavioural equivalences, behavioural metrics can be characterized via fixpoint equations. In addition there are alternative characterizations of metrics, either via modal logics or via spoiler-duplicator games. The focus of this Dagstuhl Seminar is in particular on quantitative logics, where evaluating a formula on a state does not yield a boolean value, but a (real) number, intuitively indicating to which degree a state satisfies a formula. This is connected to a quantitative Hennessy-Milner property which is supposed to hold for the behavioural distance.

There are various applications in model-checking, differential privacy, hybrid systems and learning. Several challenges in this area have been identified: studying suitable metrics and their corresponding logics, generalizing to the setting of coalgebras by parameterizing the branching type of the system under consideration, developing methods for quantitative algebraic reasoning, and finding efficient methods for computing behavioural metrics.

This Dagstuhl Seminar provided a forum for researchers working in all areas of behavioural metrics and quantitative logics, to discuss the state-of-the-art and further developments, and in particular to address applications in various domains, including machine learning.

The topics discussed at the seminar included all aspects of behavioural metrics and quantitative logics. In particular:

- Various approaches to define behavioural metrics, including characterizations via fixpoint equations, logics and games and their relations.
- Quantitative logics and their expressiveness.
- Methods and theories for quantitative equational reasoning.
- Algorithms for (compositionally) computing behavioural metrics or distinguishing formulas and their efficiency.
- Applications of behavioural metrics and quantitative logics, including – but not limited to – model-checking, differential privacy, hybrid systems and learning.

The seminar programme left room to address the above individual topics, but the main aim was to connect these topics, which previously have been studied by different research groups and published in different venues. Establishing such connections proved to be the main way forward, both to solidify the field and to identify new research problems and opportunities for applications. In particular, a key challenge was to connect recent approaches to quantitative logics, game characterisations, and quantitative equational theories, and to generalise these connections and algorithmic perspectives to enable their application to a wide range of models. We also focused on practical applications and promoted the usefulness of behavioural metrics. The seminar stimulated interactions between more theoretical scientists with researchers working on application-oriented aspects.

We invited four representatives of the different communities to give tutorial talks in order to introduce fundamental concepts and techniques. Specifically, the following four tutorial talks took place on the first day of the seminar:

- Franck van Breugel: Behavioural metrics
- Clemens Kupke: Coalgebras
- Giorgio Bacci: Quantitative equational reasoning
- Pablo Castro: Behavioural metrics for Reinforcement Learning

On Tuesday and Thursday we organized the following working groups in order to discuss more specific topics which were of interest to a substantial part of the participants:

- Generalised Kantorovich-Rubinstein Duality
- Quantitative Algebras/Coalgebras/Modal Logic
- Applications of Behavioural Metrics
- Fixpoints
- Locksteps

The seminar made it possible for many participants to connect and discuss joint collaborations and research projects. The seminar at Dagstuhl was also the opportunity for Matteo Mio, Lutz Schröder, Henning Urbat and Paul Wild to coordinate between the pre-proposal and the full proposal submission for an ANR-DFG project. The full proposal will be submitted in March 2025. This project also involves other applicants from FAU Erlangen-Nürnberg and ENS-Lyon and it aims at furthering our understanding of the theory of quantitative algebras. This topic was the main subject of many talks of this seminar. Further ideas for follow-up events and outreach activities were developed in the wrapup session (for the report on this session see Section 5.6).

The organizers would like to thank all the participants and speakers for their inspiring talks and many interesting discussions. Furthermore we would like to acknowledge Florence Clerc who helped to write and prepare this report. A special thanks goes to the Dagstuhl staff who were a great help in organizing this seminar.

2 Table of Contents

3 Overview of Tutorials

3.1 Behavioural Metrics

Franck van Breugel (York University – Toronto, CA)

License Creative Commons BY 4.0 International license
© Franck van Breugel

In this tutorial, we motivate the need for behavioural metrics. We show how the notion of probabilistic bisimilarity, due to Kim Larsen and Arne Skou, can be generalized. For this generalization, we consider the characterization of probabilistic bisimilarity in terms of couplings due to Bengt Jonsson and Kim Larsen. The coupling captures how to match transitions such that the target states are probabilistic bisimilar, that is, behave the same. By matching transitions the targets of which behave similarly yet not necessarily the same, we arrive at probabilistic bisimilarity distances. We present the logical characterization due to Josee Desharnais, Vineet Gupta, Radha Jagadeesan, and Prakash Panangaden. Furthermore, we briefly discuss some algorithms to approximate and compute the distances.

3.2 Coalgebras

Clemens Kupke (University of Strathclyde – Glasgow, GB)

License Creative Commons BY 4.0 International license
© Clemens Kupke

In my tutorial I will introduce coalgebra as a general framework to model cyclic structures and various types of transition systems. I will discuss central notions of the coalgebraic theory such as bisimulation and behavioural equivalence. Finally I will survey different logics for reasoning about coalgebras. While the talk will focus on Boolean-valued logics I will also sketch how the framework can be extended to logics for quantitative reasoning.

3.3 Quantitative Equational Reasoning

Giorgio Bacci (Aalborg University, DK)

License Creative Commons BY 4.0 International license
© Giorgio Bacci

In this talk, I review the basic definitions, constructions and key results presented in seminal work by Mardare, Panangaden and Plotkin (LICS'16), Quantitative Algebraic Reasoning, which constitutes the first step of the more ambitious program of understanding the algebraic properties of computational effects on categories enriched over (extended) metric spaces. I provide several motivating examples to offer a general pragmatic picture of how algebraic reasoning works and how it can be used. Finally, I conclude by presenting some influential work in this area of research and discussing some open problems.

3.4 Behavioural Metrics for Reinforcement Learning

Pablo Castro (Google DeepMind – Montreal, CA)

License Creative Commons BY 4.0 International license
© Pablo Castro

In this tutorial we will provide an introduction to reinforcement learning (RL), along with many of the longstanding challenges in the field; one of these is the question of how to deal with very large state spaces. While there are a variety of ways to tackle this problem, we explore the use of behavioural metrics for shaping representations so as to maximize generalizability without sacrificing accuracy. This is of particular interest when combining RL with modern function approximators like deep neural networks. We present a number of recent approaches based on the theory of bisimulation metrics, and provide pointers to open research problems in this area.

4 Overview of Talks

4.1 Behavioural Metrics via Functor Lifting – A Coalgebraic Approach

Barbara König (Universität Duisburg-Essen, DE)

License Creative Commons BY 4.0 International license
© Barbara König

Behavioural distances of transition systems modelled via coalgebras for endofunctors generalize traditional notions of behavioural equivalence to a quantitative setting, in which states are equipped with a measure of how (dis)similar they are. Endowing transition systems with such distances essentially relies on the ability to lift functors describing the one-step behavior of the transition systems to the category of pseudometric spaces. We consider the category theoretic generalization of the Kantorovich/Wasserstein lifting from transportation theory to the case of lifting functors to pseudo-metric spaces.

We also consider compositionality results which are essential ingredients for adapting up-to-techniques to the case of behavioural distances. Up-to techniques are a well-known coinductive technique for efficiently showing lower bounds for behavioural metrics. We illustrate the method with a case study on probabilistic automata.

References

1 Paolo Baldan, Filippo Bonchi, Henning Kerstan, and Barbara König. Coalgebraic behavioral metrics. *Logical Methods in Computer Science*, 14(3), 2018. Selected Papers of the 6th Conference on Algebra and Coalgebra in Computer Science (CALCO 2015). `doi:10.23638/LMCS-14(3:20)2018`.

2 Keri D'Angelo, Sebastian Gurke, Johanna Maria Kirss, Barbara König, Matina Najafi, Wojciech Różowski, and Paul Wild. Behavioural metrics: Compositionality of the Kantorovich lifting and an application to up-to techniques. In *Proc. of CONCUR '24*, volume 311 of *LIPIcs*, pages 20:1–20:19. Schloss Dagstuhl – Leibniz Center for Informatics, 2024.

4.2 Drawing from an Urn is Isometric

Bart Jacobs (Radboud University – Nijmegen, NL)

License Creative Commons BY 4.0 International license
© Bart Jacobs

Drawing (a multiset of) coloured balls from an urn is one of the most basic models in discrete probability theory. Three modes of drawing are commonly distinguished: multinomial (draw-replace), hypergeometric (draw-delete), and Polya (draw-add). These drawing operations are represented as maps from urns to distributions over multisets of draws. The set of urns is a metric space via the Wasserstein distance. The set of distributions over draws is also a metric space, using Wasserstein-over-Wasserstein. It is shown that these three draw operations are all isometries, that is, they exactly preserve the Wasserstein distances. Further, drawing is studied in the limit, both for large urns and for large draws. First it is shown that, as the urn size increases, the Wasserstein distances go to zero between hypergeometric and multinomial draws, and also between Pólya and multinomial draws. Second, it is shown that, as the draw size increases, the Wasserstein distance goes to zero (in probability) between an urn and (normalised) multinomial draws from the urn. These results are known, but here, they are formulated in a novel manner as limits of Wasserstein distances. We call these two limit results the law of large urns and the law of large draws.

4.3 On the Challenges of Quantitative Reasoning

Radu Mardare (Heriot-Watt University – Edinburgh, GB)

License Creative Commons BY 4.0 International license
© Radu Mardare

We present the main principles of Lawvere Logic, a logic interpreted on top of the semiring of extended positive reals taken with reverse order. This realizes and extends some of the ideas of Lawvere for developing the theory of metric spaces. Lawvere logics are proposed both as a way to extend the quantitative equational logics, and as a way of approaching approximation theories.

4.4 Beyond Metrics and Non Expansive Operations in Quantitative Algebra

Matteo Mio (ENS – Lyon, FR)

License Creative Commons BY 4.0 International license
© Matteo Mio

I will present some recent generalisations of the theory of quantitative algebras, originally introduced by Mardare Panangaden Plotkin in a LICS 2016 paper, which can accomodate quantitative algebras whose distance and operations are not restricted to be metrics and nonexpansive maps, respectively.

4.5 Explainability of Probabilistic Bisimilarity Distances

Franck van Breugel (York University – Toronto, CA)

License Creative Commons BY 4.0 International license
© Franck van Breugel

Probabilistic bisimilarity distances measure the similarity of behaviour of states of a labelled Markov chain. The smaller the distance between two states, the more alike they behave. Their distance is zero if and only if they are probabilistic bisimilar. Recently, algorithms have been developed that can compute probabilistic bisimilarity distances for labelled Markov chains with thousands of states within seconds. However, say we compute that the distance of two states is 0.125. How does one explain that 0.125 captures the similarity of their behaviour?

In this talk, we discuss two approaches to explainability: a logical one and a game-theoretic one. In the logical approach, which is joint work with Amgad Rady, we address the question of explainability by returning to the definition of probabilistic bisimilarity distances proposed by Josee Desharnais, Vineet Gupta, Radha Jagadeesan, and Prakash Panangaden more than two decades ago. We use a slight variation of their logic to construct for each pair of states a sequence of formulas that explains the probabilistic bisimilarity distance of the states. We have developed and implemented an algorithm that computes those formulas and each formula can be computed in polynomial time.

The game-theoretic approach is joint work with Anto Nanah Ji, Emily Vlasman, and James Worrell. We demonstrate that the probabilistic bisimilarity distances can be explained by a policy of a game. We start from a characterization of probabilistic bisimilarity distances provided in joint work with Di Chen and James Worrell. We identify properties of policies that make them suitable for explaining the probabilistic bisimilarity distances and present algorithms to generate policies with those properties.

4.6 Approximating Fixpoints of Approximated Functions

Paolo Baldan (University of Padova, IT)

License Creative Commons BY 4.0 International license
© Paolo Baldan

Fixpoints are ubiquitous in computer science and when dealing with quantitative semantics and verification one is commonly led to consider least fixpoints of (higher-dimensional) functions over the non-negative reals. We show how to approximate the least fixpoint of such functions, focusing on the case in which they are not known precisely, but represented by a sequence of approximating functions that converge to them. We concentrate on monotone and non-expansive functions, for which uniqueness of fixpoints is not guaranteed and standard fixpoint iteration schemes might get stuck at a fixpoint that is not the least. Our main contribution is the identification of an iteration scheme, a variation of Mann iteration with a dampening factor, which, under suitable conditions, is shown to guarantee convergence to the least fixpoint of the function of interest. We then argue that these results are relevant in the context of model-based reinforcement learning for Markov decision processes (MDPs), showing that the proposed iteration scheme instantiates to MDPs and allows us to derive convergence to the optimal expected return. More generally, we show that our results can be used to iterate to the least fixpoint almost surely for systems where the function of interest can be approximated with given probabilistic error bounds, as it happens for probabilistic systems, which can be explored via sampling.

4.7 Quantitative Graded Semantics, Modal Logics (and Games)

Lutz Schröder (Universität Erlangen-Nürnberg, DE)

License Creative Commons BY 4.0 International license
© Lutz Schröder
Joint work of Lutz Schröder, Jonas Forster, Paul Wild, Harsh Beohar, Sebastian Gurke, Barbara König, Karla Messing

Behavioural metrics provide a robust measure of the behaviour of systems with quantitative data, such as metric or probabilistic transition systems. In analogy to the linear-time/ branching-time spectrum of two-valued behavioural equivalences on transition systems, behavioural metrics vary in granularity, and are often characterized by fragments of suitable modal logics. In the latter respect, the quantitative case is, however, more involved than the two-valued one; as a salient negative examples, we show that probabilistic metric trace distance cannot be characterized by any compositionally defined modal logic with unary modalities, in a well-delineated but broadly understood sense. We go on to provide a unifying treatment of spectra of behavioural metrics in the emerging framework of graded monads (Milius, Schröder, Pattinson, CALCO 2015), working in coalgebraic generality, that is, parametrically in the system type. In the ensuing development of *quantitative graded semantics*, we introduce algebraic presentations of graded monads on the category **Met** of metric spaces, generalizing previous systems aimed at presenting plain monads on **Met** (Mardare, Panangaden, Plotkin, LICS 2016). Moreover, we provide a general criterion for a given real-valued modal logic to characterize a given behavioural distance, phrased in terms of a separation property that depends both on the modalities and on the propositional operators of the logic. We apply this criterion to a number of case studies, notably including a new characteristic modal logic for trace distance in fuzzy metric transition systems. In a concluding discussion, we give an outlook on generic game characterizations of behavioural metrics and indeed of more general behavioural conformances, such as behavioural preorders or bisimulation topologies.

4.8 A complete Quantitative Axiomatisation of Behavioural Distance of Regular Expressions

Wojciech Różowski (University College London, GB)

License Creative Commons BY 4.0 International license
© Wojciech Różowski

Deterministic automata have been traditionally studied from the point of view of language equivalence, but another perspective is given by the canonical notion of shortest-distinguishing word distance quantifying the difference of states. Intuitively, the longer the word needed to observe a difference between two states, then the closer behaviour is. In this talk, I discuss a sound and complete axiomatisation of shortest-distinguishing-word distance between regular languages. The axiomatisation relies on a recently developed analogue of equational logic for approximate, quantitative reasoning, allowing to manipulate rational-indexed judgements of the form $e =_\epsilon f$, meaning term e is approximately equal to term f within the error margin of ϵ. The completeness argument draws techniques from order theory and Banach spaces to simplify the calculation of behavioural distance to the point it can be then mimicked by axiomatic reasoning.

4.9 Contextual Distances for Probabilistic Higher-Order Languages

Raphaëlle Crubillé (Aix-Marseille University, FR)

License Creative Commons BY 4.0 International license
© Raphaëlle Crubillé
Joint work of Raphaëlle Crubillé, Houssein Mansour

Contextual equivalence considers two higher-order programs as equivalent when they behave the same in any given context, and contexts are modelled as programs written in the same language, making this definition self-contained. In a probabilistic setting, this definition can be made quantitative in a natural way: the distance between two programs M and N is defined as the supremum at the difference that a context can observe in the behaviour of M and N. However, it has been shown by Crubillé and Dal Lago that when all programs terminate with probability 1, this definition of contextual distance actually leads to a trivial pseudometric: two programs are either equivalent, or they are as far away as possible of each other. The original proof of this fact used techniques from real analysis with Bernstein's theorem as a pivotal tool. In this talk, I presented a new proof that hilights the deep connection of this trivialisation result with the law of large numbers.

4.10 Behavioural Metrics for Higher-Order Coalgebras

Henning Urbat (Universität Erlangen-Nürnberg, DE)

License Creative Commons BY 4.0 International license
© Henning Urbat

Higher-order coalgebras are a convenient abstract setting for modelling higher-order languages and their operational semantics. In this talk, I introduce a fibrational approach to reasoning about congruence properties of higher-order coalgebras. It uniformly applies to various notions and applicative bisimilarity and applicative distances known in the literature, as well as to new settings such as higher-order languages combining nondeterministic and probabilistic effects.

4.11 Minimising the Probabilistic Bisimilarity Distance

Qiyi Tang (University of Liverpool, GB)

License Creative Commons BY 4.0 International license
© Qiyi Tang

A labelled Markov decision process (MDP) is a labelled Markov chain with nondeterminism; i.e., together with a strategy a labelled MDP induces a labelled Markov chain. The model is related to interval Markov chains. Motivated by applications to the verification of probabilistic noninterference in security, we study problems of minimising probabilistic bisimilarity distances of labelled MDPs, in particular, whether there exist strategies such that the probabilistic bisimilarity distance between the induced labelled Markov chains is less than a given rational number, both for memoryless strategies and general strategies. We show that the distance minimisation problem is ExTh(R)-complete for memoryless strategies and undecidable for general strategies. We also study the computational complexity of the qualitative problem about making the distance less than one. This problem is known to be NP-complete for memoryless strategies. We show that it is EXPTIME-complete for general strategies.

4.12 Bisimulation and Metrics for Continuous-Time Systems

Florence Clerc (Heriot-Watt University – Edinburgh, GB)

License Creative Commons BY 4.0 International license
© Florence Clerc
Joint work of Florence Clerc, Linan Chen, Prakash Panangaden

In this talk, I will be looking at extending the notion of bismulation and bisimulation metrics from discrete-time to continuous-time. The whole discrete-time approach relies entirely on the step-based dynamics: the process jumps from state to state. What we mean by continuous-time is that processes evolve continuously through time, such as Brownian motion or involve jumps or both.

I will first recall some previous work on how to extend the notion of bisimulation to continuous-time by considering trajectories and I will discuss the limits of this approach.

I will then generalize the concept of bisimulation metric in order to metrize the behaviour of continuous-time processes. Similarly to what is done for discrete-time systems, we follow two approaches and show that they coincide: as a fixpoint of a functional and through a real-valued logic.

4.13 Polytopal Stochastic Games

Pedro d'Argenio (National University – Córdoba, AR)

License Creative Commons BY 4.0 International license
© Pedro d'Argenio

Polytopal stochastic games are an extension of two-player, zero-sum, turn-based stochastic games. In these games the uncertainty over the probability distributions is captured via linear (in)equalities whose space of solutions forms a polytope. The idea has its root in a previous work (Castro et al. EXPRESS/SOS 2023) that explores a distance to quantify robustness of fault-tolerant algorithms. Such distance is the value of a total accumulative reward objective in a stochastic game where one of the players needs to choose transitions from a polytope of distributions. Polytopal stochastic games extend this idea so that both players can choose distribution from a finite set of polytopes of distributions. In the presentation I discuss reachability and different reward objectives and present results of determinacy, decidability and complexity. This presentation is based on a joint work with Pablo Castro (UNRC, CONICET) that has been recently submitted for publication.

4.14 The Kantorovich Pseudometric Is Not The Only Nice One

Josée Desharnais (Université Laval – Québec, CA)

License Creative Commons BY 4.0 International license
© Josée Desharnais

The most studied and accepted pseudo metric for probabilistic processes is one based on the Kantorovich distance between distributions. It comes with many theoretical and motivating results. With all these seals of approval, it has imposed itself as the one to study.

Other notions of pseudometrics have been proposed, one of them (epsilon-distance) based on epsilon-bisimulation. Its advantages over the Kantorovich distance are that it is intuitively easier to understand, it relates systems that have close probabilites even if these differences can imply very different behavior in the long run. For example an imperfect implementation can be considered close to its specification. Finally, it it is easier to compute.

However, this approximate equivalence has only been defined on Markov chains in a structural way. We show that epsilon-distance is also the greatest fixpoint of a functor. The latter is obtained by replacing the Kantorovich distance in the lifting functor with the Lévy-Prokorov distance.

4.15 Type Systems for Numerical Error Analysis

Justin Hsu (Cornell University – Ithaca, US)

License Creative Commons BY 4.0 International license
© Justin Hsu
Joint work of Justin Hsu, Ariel Kellison

Programs in numerical analysis and scientific computing make heavy use of floating-point numbers to approximate ideal real numbers. Operations on floating-point numbers incur roundoff error, and an important practical problem is to bound the overall magnitude of the error for complex computations. Existing work employs a variety of strategies such as interval arithmetic, SMT encodings, and global optimization; however, current methods are not compositional and are limited in scalability, precision, and expressivity.

Today, I'll talk about some of our recent work on NumFuzz, a higher-order language that can bound forward error using a linear, coeffect type system for sensitivity analysis combined with a novel quantitative error type. Subject to certain restrictions, our type inference procedure derives sound relative error bounds for programs that are several orders of magnitude larger than previously possible, while deriving relative error bounds that are several orders of magnitude more precise than prior work.

4.16 Dissimilarity for Linear Dynamical Systems

Giovanni Bacci (Aalborg University, DK)

License Creative Commons BY 4.0 International license
© Giovanni Bacci

We introduce backward dissimilarity (BD) for discrete-time linear dynamical systems (LDS), which relaxes existing notions of bisimulations by allowing for approximate comparisons. BD is an invariant property stating that the difference along the evolution of the dynamics governing two state variables is bounded by a constant, which we call dissimilarity. We demonstrate the applicability of BD in a simple case study and showcase its use concerning: (i) robust model comparison; (ii) approximate model reduction; and (iii) approximate data recovery. Our main technical contribution is a policy-iteration algorithm to compute BDs. Using a prototype implementation, we apply it to benchmarks from network science and discrete-time Markov chains and compare it against a related notion of bisimulation for linear control systems.

5 Summary of Working Groups

5.1 Session on Generalised Kantorovich-Rubinstein Duality

Barbara König (Universität Duisburg-Essen, DE), Jurriaan Rot (Radboud University – Nijmegen, NL)

License Creative Commons BY 4.0 International license
© Barbara König and Jurriaan Rot

We discussed generalisations of the classical Kantorovich-Rubinstein duality to a categorical level. While the primal and dual formulation coincide in the probabilistic case, this is not true any more when these liftings are generalized. In this case one type of lifting (sometimes called Wasserstein lifting) is based on couplings and takes the infimum while the other (sometimes called Kantorovich or codensity lifting) is based on non-expansive price functions and takes the supremum.

We started by discussing whether we assume the same types of predicate liftings (aka modalities) for both liftings, since allowing to vary the modalities would provide easy possibilities to represent one type of lifting as the other.

We continued with the assumption that the modalities are the same for both liftings. In the sequence, particular emphasis was on the subdistribution functor. The common generalisation of probabilistic couplings here enforces too strong a condition, suggesting that the functor lifting based on it does not yield the right notion; we discussed possible variants and Florence Clerc suggested some relevant literature.

A potential challenge to a general formulation of duality is that it should already include arguments for the well-known case of the distribution functor, which even in the discrete case is not immediate. Barbara König presented a concrete proof of this result and we discussed its possible adaptation to the general case.

5.2 Session on Quantitative Algebras/Coalgebras/Modal Logic

Clemens Kupke (University of Strathclyde - Glasgow, GB)

License Creative Commons BY 4.0 International license
© Clemens Kupke

The first point of the discussion was how to relate process algebras (an algebraic theory) with processes (coalgebras). One such connection can be obtained as follows: generate the free algebra (for the operations and equations of the process algebra) and turn this into coalgebra. In order to get all coalgebras (also circular ones) one would need to consider the Cauchy completion. Uniqueness of fixpoints was identified as very useful to be able to approximate this completion. A key problem for moving from processes to process algebra is to axiomatise equational theory generated by bisimulation.

The second point of the discussion concerned what a quantitative coequation is. One proposed idea was to use Stone duality to obtain the canonical model satisfying the "coequation". Expressiveness of coequations is fundamentally linked to HM theorems (examples are relational vs event bisimulation for Markov processes). An identified direction for further discussions is the following simple idea: coequations correspond to subcoalgebras of the final coalgebra. To obtain a quantitative notion of coequation we need to define an approximate/metric version of the notion of subcoalgebra.

5.3 Session on Applications of Behavioural Metrics

Jurriaan Rot (Radboud University – Nijmegen, NL)

License Creative Commons BY 4.0 International license
© Jurriaan Rot

In this breakout session, we discussed possible application domains of behavioural metrics, and had an extensive broader discussion on relations between the theoretical work that receives a lot of emphasis in our community and broader applications and multi-disciplinary work.

We started with a general discussion of the role of theoretical work in computer science. A natural process is that we stay within a "theoretical bubble", which limits potential applications and more long-term the viability and relevance of the field. We discussed whether everyone should "step out of the bubble" or not, what the challenges are for doing so, and how taking on this challenges can be rewarded in a better way than it is currently. We discussed how to enable and support interaction between theory and practice (both ways) by suitable ways of meeting and communicating as a broader scientific community, and had a brainstorm session about possibilities. Ideas that were mentioned included scientific meetings/conferences that focus on short talks and integrating the community rather than on proceedings; newsletters and social media; colocating workshops with bigger venues; and discussed several related areas where this seems to be going well.

More specifically, we identified application areas of behavioural metrics, including in explainability, differential privacy, machine learning and planning, reinforcement learning, and (model-based) software testing.

5.4 Session on Fixpoints

Barbara König (Universität Duisburg-Essen, DE)

License Creative Commons BY 4.0 International license
© Barbara König

This breakout session was concerned with the topic of fixpoints in various contexts. Already in the introduction round on Monday many seminar participants stated an interest in fixpoint theoy in general and for behavioural metrics in particular. Hence, the aim of the session was to reflect on the state of the art, exchange ideas and identify new research directions.

The discussion started with fixpoint games, in particular generalized fixpoint games in complete lattices. We also discussed the connection to mu-calculus and parity games and later to simple stochastic and coalgebraic games and techniques for computing behavioural metrics via solution procedures for games.

This triggered an exchange about algorithms and heuristics for computing behavioural metrics, including exploiting dependencies between variables, the worklist algorithm and chaotic iteration.

Subsequently Radu Mardare presented an axiomatization of fipxoints based on so-called Banach patterns and Banach theory. He also gave a comparison to the metric coinduction principle. The following discussion was rather broad, but for a while centered on the question of extending this theory to a setting with non-unique fixpoints (unlike the case of the Banach theorem that relies on contractions with unique fixpoints).

Afterwards the focus of the participants was on linear time (or trace) semantics and the question whether there is an increase in difficulty wrt. the qualitative case. We discussed a negative result on the existence of expressive logics from a previous talk by Lutz Schröder. In this context we also reviewed coalgebraic approaches for representing linear time such as coalgebras living in Kleisli and Eilenberg-Moore categories.

5.5 Session on Locksteps

Florence Clerc (University of Strathclyde - Glasgow, GB)

License Creative Commons BY 4.0 International license
© Florence Clerc

We discussed about continuous-time Markov chains, continuous-time Markov decision processes and Markov automata and we explored the poccibility of behavioural "scores" that consider the relative speed of the processes associating to the more standard bisimulation metric. We intend to compare this "score" to previously defined bisimulations on continuous-time Markov chains, continuous-time Markov decision processes and Markov automata.

5.6 Wrapup Session

Barbara König (Universität Duisburg-Essen, DE)

License Creative Commons BY 4.0 International license
© Barbara König

The wrapup session gave all participants the opportunity to draw attention to discussions that took place in separate breakout sessions, reflect on the seminar and discuss followup activities. Only very few participants had already left, so that discussion took place with almost all of the participants present.

The session started with a report from the algebra/coalgebra session on Tuesday, where the participants pointed out that definining a quantitative notion of subcoalgebras (respectively coequations) is still an open question and would merit further investigation.

In addition there was a report from the participants of the application session on Thursday where the discussion had centered about strategies for escaping from the "theoretical bubble" as well as observations that theorems (also from category theory) are now starting to become useful in many application areas, such as in machine learning. It was emphasized that researchers should be encouraged to leave their special field and solve problems in other domains. Furthermore, some conferences (such as TACAS and FMICS) have tracks on applications that allow to publish corresponding papers on practical results.

This lead to a discussion on followup activities. Ideas for concrete activities that were mentioned included:

- staying in contact via social media; establish a discussion forum (via Zulip?); announcements of new papers and developments
- organize a one-day workshop affiliated with a conferences such as CONCUR, ETAPS or LICS; such a workshop need not have proceedings, but should serve as a forum for staying in contact in the community
- organize/initiate more sessions and tutorials on behavioural metrics at conferences
- projects (in particular projects that provide money for exchange activities)
- mailing lists

In general the participants perceived the emergence of a new community triggered – among other activities – by the Dagstuhl Seminar.

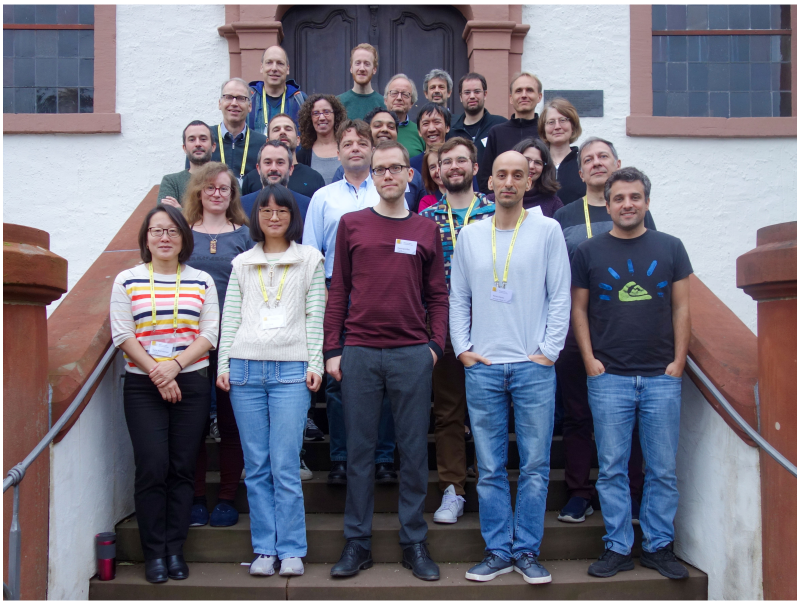
The session and the entire seminar were concluded by a short presentation by Bart Jacobs where he suggested to include more intuition when talking about and teaching probability theory. In particular he reviewed the analogy to water pipes and pumps.

Participants

- Giorgio Bacci
Aalborg University, DK
- Giovanni Bacci
Aalborg University, DK
- Paolo Baldan
University of Padova, IT
- Corina Cirstea
University of Southampton, GB
- Florence Clerc
Heriot-Watt University – Edinburgh, GB
- Raphaëlle Crubillé
Aix-Marseille University, FR
- Pedro R. D'Argenio
National University – Córdoba, AR
- Josée Desharnais
Université Laval – Québec, CA
- Helle Hvid Hansen
University of Groningen, NL
- Justin Hsu
Cornell University – Ithaca, US
- Bart Jacobs
Radboud University – Nijmegen, NL
- Clemens Kupke
University of Strathclyde – Glasgow, GB
- Barbara König
Universität Duisburg-Essen, DE
- Radu Mardare
Heriot-Watt University – Edinburgh, GB
- Matteo Mio
ENS – Lyon, FR
- Renato Neves
University of Minho, PT
- Pedro Nora
Radboud University – Nijmegen, NL
- Jurriaan Rot
Radboud University – Nijmegen, NL
- Wojciech RRóżowski
University College London, GB
- Lutz Schröder
Universität Erlangen-Nürnberg, DE
- Qiyi Tang
University of Liverpool, GB
- Henning Urbat
Universität Erlangen-Nürnberg, DE
- Paul Wild
Universität Erlangen-Nürnberg, DE
- Franck van Breugel
York University – Toronto, CA

Remote Participants

- Pablo Samuel Castro
Google DeepMind – Montreal, CA
- Prakash Panangaden
McGill University – Montréal, CA



Report from Dagstuhl Seminar 24441

Machine Learning Augmented Algorithms for Combinatorial Optimization Problems

Deepak Ajwani*[1], Bistra Dilkina*[2], Tias Guns*[3], and Ulrich Carsten Meyer*[4]

1 **University College Dublin, IE.** deepak.ajwani@ucd.ie
2 **USC – Los Angeles, US.** dilkina@usc.edu
3 **KU Leuven, BE.** tias.guns@kuleuven.be
4 **Goethe University – Frankfurt am Main, DE.** umeyer@ae.cs.uni-frankfurt.de

Abstract

Combinatorial optimization problems are pervasive across critical domains, including business analytics, engineering, supply chain management, transportation, and bioinformatics. Many of these problems are NP-hard, posing significant challenges for even moderately sized instances. Moreover, even when polynomial-time algorithms exist, their practical implementation can be computationally expensive for real-world applications.

This has driven decades of research across diverse fields, encompassing exact and approximation algorithms, parameterized algorithms, algorithm engineering, operations research, optimization solvers (such as mixed-integer linear programming and constraint programming solvers), and nature-inspired metaheuristics.

Recently, there has been a surge in research exploring the synergistic integration of machine learning techniques with algorithmic insights and optimization solvers to enhance the scalability of solving combinatorial optimization problems. However, research efforts in this area are currently fragmented across several distinct communities, including those focused on "Learning to scale optimization solvers," "Algorithm Engineering," "Algorithms with predictions," and "Decision-focused learning."

This seminar brought together researchers from these diverse communities, fostering a dialogue on effectively combining algorithm engineering techniques, optimization solvers, and machine learning to address these challenging problems. The seminar facilitated the development of a shared vocabulary, clarifying similarities and distinctions between concepts across different research areas. Furthermore, significant progress was made in identifying key research directions for the future advancement of this field. We anticipate that these outcomes will serve as a valuable roadmap for advancing this exciting research area.

Seminar October 27–31, 2024 – https://www.dagstuhl.de/24441

2012 ACM Subject Classification Computing methodologies → Discrete space search; Theory of computation → Design and analysis of algorithms; Theory of computation → Discrete optimization

Keywords and phrases Algorithm Engineering, Combinatorial Optimization, Constraint Solvers, Machine Learning

Digital Object Identifier 10.4230/DagRep.14.10.76

* Editor / Organizer

Except where otherwise noted, content of this report is licensed under a Creative Commons BY 4.0 International license

Machine Learning Augmented Algorithms for Combinatorial Optimization Problems, *Dagstuhl Reports*, Vol. 14, Issue 10, pp. 76–100
Editors: Deepak Ajwani, Bistra Dilkina, Tias Guns, and Ulrich Carsten Meyer
Dagstuhl Reports
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

1 Executive Summary

Deepak Ajwani (University College Dublin, IE)
Bistra Dilkina (USC – Los Angeles, US)
Tias Guns (KU Leuven, BE)
Ulrich Carsten Meyer (Goethe University – Frankfurt am Main, DE)

License Creative Commons BY 4.0 International license
© Deepak Ajwani, Bistra Dilkina, Tias Guns, and Ulrich Carsten Meyer

Combinatorial optimization problems are pervasive across critical domains, driving decades of research across diverse fields, including exact and approximation algorithms, algorithm engineering, operations research, and optimization solvers (such as mixed-integer linear programming and constraint programming solvers). In recent years, there has been a surge in research at the intersection between machine learning, optimization solvers, and algorithm engineering. While significant progress has been made, research efforts in this area remain fragmented across several distinct communities:

- "Learning to scale optimization solvers": This community focuses on leveraging learning techniques to improve the performance of solvers, such as mixed-integer linear programming, constraint programming, and satisfiability solvers, often using generic, problem-agnostic features. This includes optimizing solver components and exploring "end-to-end" approaches that replace traditional solvers entirely.
- "Algorithm Engineering": This community utilizes learning techniques to select the most effective components within algorithms, fill in the underspecified parts of the algorithms, and develop novel heuristics, often leveraging problem-specific features and algorithmic insights.
- "Algorithms with predictions": This community develops algorithms that leverage predictions to improve performance, often treating the prediction system as a black box. These algorithms aim for good performance with accurate predictions while maintaining bounded worst-case performance with inaccurate predictions.
- "Decision-focused learning": This research area focuses on systems where optimization decisions are made based on machine learning models, necessitating an integrated approach to avoid suboptimal solutions.

This seminar brought together researchers from these diverse communities, fostering a dialogue on effectively combining algorithm engineering techniques, optimization solvers, and machine learning. The seminar facilitated the development of a shared vocabulary, clarifying similarities and distinctions between concepts across different research areas. Notably, many concepts with different names (e.g., "learning to prune," "predict then search", "finding kernel", "finding a backbone", "learning to branch", "learning constraints") were found to represent similar ideas. Key differences often stem from the source of features, with "learning to prune" typically relying on problem-specific, algorithm-based features, while "learning to branch" often utilizes problem-agnostic features derived from the solver run. This shared understanding fostered significant synergy and cross-fertilization across research areas.

The seminar featured discussions on recent advancements, at the intersection of machine learning, algorithm engineering, and combinatorial optimization. For instance, the tutorial on "ML-Guided Solvers for MIP" focused on recent advances in using contrastive learning for MILP solvers, the talk on "Neural Combinatorial Optimization – One Model to Solve Them All?" described recent advances in foundation models for sequencing combinatorial optimization and the talk on "Graph Learning for Planning" described very impressive results on the usage of learning techniques for planning problems. In addition, many participant talks such as "Progress and Open Questions on DFL and Constrained NNs" and "Machine Learning for Edge Contractions in Multilevel Graph Partitinioning" highlighted interesting open problems in the area.

Furthermore, the seminar identified key research directions, including:

- Theoretical frameworks for machine learning augmented combinatorial optimization algorithms
- Learning to fill the underspecified parts of the algorithms
- Reinforcement learning for combinatorial optimization
- Contextual Stochastic Optimization and DFL
- Fast and/or interpretable machine learning techniques for combinatorial problems
- Scalability and Datasets for Decision Focused Learning
- Finding worst-case instances for algorithms using machine learning techniques

These directions are expected to serve as a valuable roadmap for advancing this exciting research area.

Seminar Overview

The seminar commenced with a vibrant round of introductions, allowing participants to share their research interests and backgrounds. The slides for this round of introductions were collected in advance. To foster interdisciplinary connections, participant order was randomized, encouraging interaction across the diverse research communities represented.

Following this, four insightful tutorials were presented, each from a different research community. These tutorials served to familiarize all participants with the unique vocabulary, key techniques, and recent advancements within each area, providing a common ground for subsequent discussions.

A series of engaging participant talks followed, showcasing cutting-edge research from across the field. The talk schedule was carefully randomized to maximize interaction and foster cross-pollination of ideas among the diverse research communities. This dynamic approach encouraged participants to step outside their immediate research areas and engage with novel perspectives.

Recognizing the importance of collective vision, participants were invited to submit key research directions and open challenges within the field. These valuable contributions were then clustered, enabling the identification of key research themes and areas of significant opportunity.

To delve deeper into these themes, three parallel discussion sessions were held. Each session featured three distinct discussion groups, allowing participants to select the group most aligned with their interests. This dynamic group composition, which changed daily, encouraged participants to engage with a variety of perspectives and fostered a truly inter-community exchange of ideas. It was particularly encouraging to observe that many discussion groups comprised members from all four research communities, demonstrating the successful integration of diverse perspectives throughout the seminar.

2 Table of Contents

3 Tutorials

3.1 Decision-focused learning: Foundations, State of the Art, Benchmark and Future Opportunities

Tias Guns (KU Leuven, BE)

License Creative Commons BY 4.0 International license
© Tias Guns

Increasingly, combinatorial optimisation problems follow a predict + optimize paradigm, where part of the parameters (costs, volumes, capacities) are predicted from data, and those predictions are fed into a downstream combinatorial optimisation problem. How to best train these predictive models? Decision-focused learning (DFL) is an emerging paradigm in machine learning which trains a model to optimize decisions, integrating prediction and optimization in an end-to-end system. This paradigm holds the promise to revolutionize decision-making in many real-world applications which operate under uncertainty, where the estimation of unknown parameters within these decision models often becomes a substantial roadblock to high-quality solutions. This talk presents a comprehensive review of DFL. It provides an in-depth analysis of the various techniques devised to integrate machine learning and optimization models, introduces a taxonomy of DFL methods distinguished by their unique characteristics, and conducts an extensive empirical evaluation of these methods proposing suitable benchmark dataset and tasks for DFL. Finally, we'll provide valuable insights into current and potential future avenues in DFL research.

3.2 Approximation Algorithms with Predictions

Adam Polak (Bocconi University – Milan, IT)

License Creative Commons BY 4.0 International license
© Adam Polak
Joint work of Adam Polak, Antonios Antoniadis, Marek Eliáš, Moritz Venzin

I will talk about utilizing predictions to improve over approximation guarantees of classic algorithms, without increasing the running time. I will cover prior results going in that direction, and then I will talk about our recent work in which we propose a generic method for a wide class of optimization problems that ask to select a feasible subset of input items of minimal (or maximal) total weight. This gives simple (near-)linear-time algorithms for, e.g., Vertex Cover, Steiner Tree, Min-Weight Perfect Matching, Knapsack, and Clique. Our algorithms produce an optimal solution when provided with perfect predictions and their approximation ratio smoothly degrades with increasing prediction error. With small enough prediction error we achieve approximation guarantees that are beyond the reach without predictions in given time bounds, as exemplified by the NP-hardness and APX-hardness of many of the above problems. Although we show our approach to be optimal for this class of problems as a whole, there is a potential for exploiting specific structural properties of individual problems to obtain improved bounds; we demonstrate this on the Steiner Tree problem. I will conclude with an empirical evaluation of our approach. This is based on joint work with Antonios Antoniadis, Marek Eliáš, and Moritz Venzin.

3.3 ML-Guided Solvers for MIP

Bistra Dilkina (USC – Los Angeles, US)

License Creative Commons BY 4.0 International license
© Bistra Dilkina

In this tutorial talk, I present recent advances on (i) augmenting discrete optimization algorithms with learning components and (ii) learning methods that incorporate the combinatorial decisions they inform. While the first set of techniques focus on infusing discrete optimization with machine learning, the second set of techniques focus on infusing machine learning with constrained decision making. The talk is based on my recent publications [1, 2, 3].

References

1 Taoan Huang, Aaron M. Ferber, Yuandong Tian, Bistra Dilkina and Benoit Steiner. Searching Large Neighborhoods for Integer Linear Programs with Contrastive Learning abs/2302.01578 2023.
2 Taoan Huang, Aaron M. Ferber, Arman Zharmagambetov, Yuandong Tian and Bistra Dilkina. Contrastive Predict-and-Search for Mixed Integer Linear Programs. Forty-first International Conference on Machine Learning, ICML 2024,
3 Junyang Cai, Taoan Huang and Bistra Dilkina. Learning Backdoors for Mixed Integer Linear Programs with Contrastive Learning 27th European Conference on Artificial Intelligence ECAI, 2024,

3.4 Learning problems in Algorithm design and engineering

Deepak Ajwani (University College Dublin, IE)

License Creative Commons BY 4.0 International license
© Deepak Ajwani
Joint work of Dena Tayebi, Saurabh Ray, Deepak Ajwani
Main reference Dena Tayebi, Saurabh Ray, Deepak Ajwani: "Learning to Prune Instances of k-median and Related Problems", in Proc. of the Symposium on Algorithm Engineering and Experiments, ALENEX 2022, Alexandria, VA, USA, January 9-10, 2022, pp. 184–194, SIAM, 2022.
URL http://dx.doi.org/10.1137/1.9781611977042.15

In this talk, we explored the learning problems arising in algorithm design, engineering and analysis. We will discuss (i) ways in which the insights from the algorithm engineering literature can be used in problem-specific learning solutions, (ii) how learning techniques can be applied to fill the underspecified parts in algorithms (e.g., ordering in Bellman-Ford algorithm), (iii) how machine learning can help in identifying and estimating the various parameters used in the design of parameterised algorithms, (iv) how learning techniques can be used for the design of interpretable heuristics and (v) if learning techniques can be used for finding combinatorial structures and worst-case inputs for algorithm analysis.

References

1 Dena Tayebi, Saurabh Ray and Deepak Ajwani. Learning to Prune Instances of k-median and Related Problems Proceedings of the Symposium on Algorithm Engineering and Experiments, ALENEX 2022,

4 Overview of Talks

4.1 On LLM prompt optimization and amortization

Brandon Amos (Meta – New York, US)

License Creative Commons BY 4.0 International license
© Brandon Amos

Prompting LLMs is a challenging art where different ways of expressing the same idea can lead to drastically different responses. Not prompting in the right way gives suboptimal performance and has led to the budding space of prompt engineering and optimization techniques. A standard example here is that appending the string "let's think step by step" to the prompt may significantly improve the quality on few-shot classification tasks. In this session, we'll first cover how prompt optimization can be expressed as a combinatorial optimization problem (over the token sequence space) and overview the standard methods here. Beyond this, prompt optimization problems are often not solved a single time in isolation, but are repeatedly solved for every new task or piece of information we want to extract from the language model. So, we'll conclude with an overview of learned optimization, or amortization, to share the information between the repeatedly-solved optimization problems.

4.2 Enhancing constraint programming with machine learning

Quentin Cappart (Polytechnique Montréal, CA)

License Creative Commons BY 4.0 International license
© Quentin Cappart
Joint work of Quentin Cappart, Louis-Martin Rousseau, Swann Bessa, Darius Dabert, Max Bourgeat
Main reference Swann Bessa, Darius Dabert, Max Bourgeat, Louis-Martin Rousseau, Quentin Cappart: "Learning Valid Dual Bounds in Constraint Programming: Boosted Lagrangian Decomposition with Self-Supervised Learning", CoRR, Vol. abs/2408.12695, 2024.
URL http://dx.doi.org/10.48550/ARXIV.2408.12695

Constraint programming (CP) is a well-established method for tackling combinatorial problems. Traditionally, CP has focused on solving isolated problem instances, often overlooking the fact that these instances frequently originate from related data distributions. In recent years, there has been a growing interest in leveraging machine learning, particularly neural networks, to enhance CP solvers by utilizing historical data. Despite this interest, it remains unclear how to effectively integrate learning into CP engines to boost overall performance. In this presentation, I will share my experience in tackling this challenge, from my initial attempts to my current research directions.

4.3 Learning to solve real-world puzzles: from Sudoku to protein design

Marianne Defresne (KU Leuven, BE)

License Creative Commons BY 4.0 International license
© Marianne Defresne

Real-life decision making often involves reasoning on ill-defined problems, where exact constraints or parameters (such as costs) are unknown. The goal of Decision-Focused Learning (DFL) is to automatize the definition of problem parameters by using Deep

Learning to extract knowledge out of the environment. The main challenge here is to combine discrete optimization for reasoning with continuous optimization for learning. I will present one method to learn discrete graphical models, the reasoning framework we chose. We first assessed it on learning the rules of Sudoku, a popular benchmark for DFL methods. We then apply it to Computational Protein Design, a real-world problem that can be described and tackled with graphical models. We deep learned the interactions within existing proteins to better guide the design towards new proteins of interest.

4.4 The differentiable feasibility pump

Alexandre Forel (Polytechnique Montréal, CA)

License Creative Commons BY 4.0 International license
© Alexandre Forel
Joint work of Alexandre Forel, Matteo Cacciola, Andrea Lodi, Antonio Frangioni

The feasibility pump algorithm is a widely used heuristic to find feasible primal solutions to mixed-integer linear problems. Many extensions of the algorithm have been proposed. Yet, its core algorithm remains centered around two key steps: solving the linear relaxation of the original problem to obtain a solution that respects the constraints, and rounding it to obtain an integer solution. This paper shows that the feasibility pump and many of its follow-ups can be seen as gradient-descent algorithms with specific parameters. A central aspect of this reinterpretation is observing that the traditional algorithm differentiates the solution of the linear relaxation with respect to its cost. This reinterpretation opens many opportunities for improving the performance of the original algorithm. We study how to modify the gradient-update step as well as extending its loss function. We perform extensive experiments on instances from the MIPLIB library and show that these modifications can substantially reduce the number of iterations needed to find a primal solution.

4.5 Screening for Data Reductions in Combinatorial Optimization: A Graph Neural Network Approach

Ernestine Großmann (Universität Heidelberg, DE)

License Creative Commons BY 4.0 International license
© Ernestine Großmann
Joint work of Ernestine Großmann, Kenneth Langedal, Christian Schulz

Combinatorial optimization problems play a pivotal role in various domains, challenging researchers to find optimal solutions within large solution spaces. One key strategy to tackle the complexity of these problems is data reduction, where instances are simplified to facilitate more efficient algorithms. Even though there are numerous reduction rules available, they are often not used in applications since the time needed to test them for the whole graph exhaustively becomes infeasible. With this, a new problem in the form of early reduction screening emerges. In this talk, we tackle this problem in the context of maximum weight independent sets using graph neural networks. Before checking if an expensive data reduction rule is applicable, we consult a GNN oracle to decide if the probability of successfully reducing the graph is sufficiently large. With this approach, we can use even expensive reduction rules successfully in feasible time. We introduce a new supervised learning dataset and provide first results using established graph neural network architectures.

4.6 Efficiently learning instance-optimal linear system solvers

Mikhail Khodak (Princeton University, US)

License Creative Commons BY 4.0 International license
© Mikhail Khodak

Augmenting classical algorithms with learned predictions or configurations has found many successful applications in energy management, database systems, scientific computing, and beyond. At the same time it has been theoretically challenging to go beyond the computationally inefficient learning of static predictors and configurations. We study the problem of sequentially solving linear systems, a fundamental problem in numerical computing, and introduce an algorithm that efficiently learns to do instance-optimal linear solver configuration while using only the number of iterations as feedback. Our approach points towards new directions for analyzing iterative methods, designing surrogate objectives for optimizing algorithmic costs, and incorporating tools from online learning into algorithm design.

4.7 The Expressive Power of Graph Neural Networks

Sandra Kiefer (University of Oxford, GB)

License Creative Commons BY 4.0 International license
© Sandra Kiefer

Graph Neural Networks (GNNs) are a machine learning architecture to learn functions on graphs. For example, since problem instances for combinatorial optimisation tasks are often modelled as graphs, GNNs have recently received attention as a natural framework for finding good heuristics in neural optimisation approaches. The question which functions can actually be learnt by message-passing GNNs and which ones exceed their power has been studied extensively. In this talk, I will consider it from a graph-theoretical perspective. I will survey the Weisfeiler-Leman algorithm as a combinatorial procedure to analyse and compare graph structure, and I will discuss some results concerning the power of the algorithm on natural graph classes. The findings directly translate into insights about the power of GNNs and of their extensions to higher-dimensional neural networks.

4.8 Non-Clairvoyant Scheduling with Prediction

Alex Lindermayr (Universität Bremen, DE)

License Creative Commons BY 4.0 International license
© Alex Lindermayr

In this talk, we explore some recent and ongoing advancements in non-clairvoyant scheduling in the learning-augmented framework, which integrates error-prone predictions into online algorithm design. We examine various prediction models, showcasing algorithms for non-clairvoyant scheduling with strong error-dependent performance guarantees. In particular, we ask which type of information is required and how an algorithm should receive it to achieve reasonable performance guarantees. Moreover, we consider recently popular prediction models for other problems, and discuss how they may be integrated into scheduling problems.

4.9 Progress and Open Questions on DFL and Constrained NNs

Michele Lombardi (University of Bologna, IT)

License Creative Commons BY 4.0 International license
© Michele Lombardi

This talk will provide perspectives, progress, and open questions on two distinct forms of integration between optimization and learning. First, we will present an analysis of Decision Focused Learning approaches, highlighting structural problem properties that can cause their advantage w.r.t. classical techniques to be significantly diminished. We will suggest a possible way out, with interesting practical applications, together with one solution technique. Second, we consider the problem of enforcing hard constraints in Neural Networks, discussing a training-time approach with satisfaction guarantees. The method combines Projected Gradient Descent with a neural architecture that embeds a trainable over-approximation module. For both the considered setting, we highlight open problems and promising research directions.

4.10 Open Problem: Machine Learning for Edge Contractions in Multilevel Graph Partitinioning

Nikolai Maas (KIT – Karlsruher Institut für Technologie, DE)

License Creative Commons BY 4.0 International license
© Nikolai Maas

In this talk, I will discuss some open problems in applying machine learning for edge contractions in multilevel graph partitioning. This will be in the context of our graph partitioning software Karlsruhe Hypergraph Partioning (KaHyPar).

4.11 Graph Neural Networks as Ordering Heuristics for Parallel Graph Coloring

Fredrik Manne (University of Bergen, NO)

License Creative Commons BY 4.0 International license
© Fredrik Manne
Joint work of Fredrik Manne, Kenneth Langedal

The graph coloring problem asks for an assignment of the minimum number of distinct colors to vertices in an undirected graph with the constraint that no pair of adjacent vertices share the same color. The problem is a thoroughly studied NP-hard combinatorial problem with several real-world applications. As such, a number of greedy heuristics have been suggested that strike a good balance between coloring quality, execution time, and also parallel scalability. In this work, we introduce a graph neural network (GNN) based ordering heuristic and demonstrate that it outperforms existing greedy ordering heuristics both on quality and performance. Previous results have demonstrated that GNNs can produce high-quality colorings but at the expense of excessive running time. The current paper is the first that brings the execution time down to compete with existing greedy heuristics. Our GNN model is trained using both supervised and unsupervised techniques. The experimental results show that a 2-layer GNN model can achieve execution times between the largest degree first (LF) and smallest degree last (SL) ordering heuristics while outperforming both on coloring quality. Increasing the number of layers improves the coloring quality further, and it is only at four layers that SL becomes faster than the GNN. Finally, our GNN-based coloring heuristic achieves superior scaling in the parallel setting compared to both SL and LF.

4.12 Neural Combinatorial Optimization – One Model to Solve Them All?

Sofia Michel (NAVER Labs Europe – Meylan, FR)

License Creative Commons BY 4.0 International license
© Sofia Michel
Joint work of Sofia Michel, Darko Drakulic, Jean-Marc Andreoli

Recent years have seen considerable progress in machine learning–based heuristics for combinatorial optimization, particularly constructive neural heuristics that use neural networks to build solutions step by step. Despite their success across a variety of combinatorial tasks, these methods often require specialized models trained separately for each problem and instance distribution. In this talk, I will discuss recent works on improving the generalization of neural combinatorial optimization approaches. I will introduce a framework that leverages the symmetries inherent in many combinatorial problems to boost the out-of-distribution generalization. Building on this framework, I will present GOAL, a generalist model capable of efficiently solving multiple combinatorial problems and which can be fine-tuned to handle new ones. By tackling a broad range of tasks – including routing, scheduling, packing, location, and graph problems – GOAL represents a promising step toward developing a foundation model for combinatorial optimization. I will finish with some current challenges and open questions.

4.13 Machine Learning for Algorithm Selection

Nysret Musliu (TU Wien, AT)

License Creative Commons BY 4.0 International license
© Nysret Musliu

Algorithm selection addresses the challenge of determining which of the available algorithms is most appropriate for solving a particular problem instance. Hyper-heuristics are a high-level, problem-independent approach that aims to automate the design of heuristic methods by combining and selecting low-level heuristics.

In this talk, we will present our work on algorithm selection for various problem domains, including scheduling and tree decomposition. We will also discuss our recent work on reinforcement learning for selection hyper-heuristics.

4.14 Group Counterfactual Explanations with a Mathematical Optimization Lens

María Dolores Romero Morales (Copenhagen Business School, DK)

License Creative Commons BY 4.0 International license
© María Dolores Romero Morales

Counterfactual Analysis has shown to be a powerful tool in the burgeoning field of Explainable Artificial Intelligence. In Supervised Classification, this means associating with each record a so-called counterfactual explanation: an instance that is close to the record and whose probability of being classified in the opposite class by a given classifier is high. In this talk we take a stakeholder perspective, and we address the setting in which a group of counterfactual explanations is sought for a group of instances. We introduce some mathematical optimization models as illustration of each possible allocation rule between counterfactuals and instances and tasks beyond Supervised Classification.

4.15 Optimal Action Imitation Learning

Louis-Martin Rousseau (Polytechnique Montréal, CA)

License Creative Commons BY 4.0 International license
© Louis-Martin Rousseau

Predictive models are playing an increasingly pivotal role in optimizing decision-making. This talk introduces an approach wherein a series of (possibly) stochastic sequential decision problems are initially solved offline, and subsequently, a predictive model is constructed based on the offline solutions to facilitate real-time decision-making. The applicability of this methodology is demonstrated through two use cases: patient radiation therapy, where managers need to preserve capacity for emergency cases, and a novel dynamic employee call-timing problem for the scheduling of casual personnel for on-call work shifts.

4.16 Two-Oracle Models for Matroid Optimization Problems

Jens Schlöter (CWI – Amsterdam, NL)

License Creative Commons BY 4.0 International license
© Jens Schlöter
Joint work of Jens Schlöter, Franziska Eberle, Felix Hommelsheim, Alexander Lindermayr, Nicole Megow, Zhenwei Liu

In many combinatorial optimization problems, access to input data is modeled in an abstract way via oracles, e.g., independence oracles for matroids, distance functions in a metric space, or comparators for sorting elements. Depending on the underlying application, the actual use of such oracles can be computationally expensive, as it may require access to slow database systems or large machine learning (ML) models. In the two-oracle model, we assume access to a second, weaker oracle, which is computationally cheap but may provide inaccurate answers. The goal is to design algorithms that exploit the information provided by the cheap oracle to minimize the number of calls to the expensive oracle.

Two-oracle models have been studied in several ML applications, such as data labeling with weak and strong data labelers, text clustering with two oracles for text similarity, or more generally in ML pipelines that combine the use of scalable and expensive models. More recently, Bai and Coester [NeurIPS'23] considered two-oracle models from the point of view of learning-augmented algorithm design for sorting with access to two comparators. As usual, the goal is to design robust algorithms with performance guarantees that improve with the quality of the second, cheap oracle. In this talk, we discuss the two-oracle model and its applications, and study it from the perspective of learning-augmented algorithms by applying it to the fundamental problem of finding a maximum-weight basis in a matroid.

The talk is based on a joint work with Franziska Eberle, Felix Hommelsheim, Alexander Lindermayr, Nicole Megow, and Zhenwei Liu.

4.17 Graph Learning for Planning

Sylvie Thiébaux (University of Toulouse, FR & Australian National University, Canberra, AU)

License Creative Commons BY 4.0 International license
© Sylvie Thiébaux
Joint work of Sylvie Thiébaux, Dillon Chen, Mingyu Hao, Joerg Hoffmann, Felipe Trevizan

I will present recent work on graph representation learning to guide the search of AI planners. I will introduce GNN and other graph learning representations that exploit the relational structure of planning domains. They allow our planner GOOSE to learn search guidance (e.g. heuristic cost estimates, state rankings) from solutions to just a few small problems, and solve substantially larger problems than trained on. Perhaps surprisingly, our experimental results show that classical machine learning approaches vastly outperform deep learning ones in this context. Moreover, Greedy Best-First Search guided by our best learnt heuristics outperforms the state of the art model-based planner, Lama, on the problems of the latest International Planning Competition Learning track, leading to the possibility that learnt heuristics may replace existing model-based heuristics in the near future.

This is joint work with Dillon Chen, Mingyu Hao, Joerg Hoffmann, and Felipe Trevizan

4.18 Learning-Based Algorithms for Graph Searching Problems

Ali Vakilian (TTIC – Chicago, US)

License Creative Commons BY 4.0 International license
© Ali Vakilian
Joint work of Adela Frances DePavia, Erasmo Tani, Ali Vakilian
Main reference Adela Frances DePavia, Erasmo Tani, Ali Vakilian: "Learning-Based Algorithms for Graph Searching Problems", CoRR, Vol. abs/2402.17736, 2024.
URL http://dx.doi.org/10.48550/ARXIV.2402.17736

We consider the problem of graph searching with prediction recently introduced by [1]. In this problem, an agent, starting at some vertex r has to traverse a (potentially unknown) graph G to find a hidden goal node g while minimizing the total distance travelled. We study a setting in which at any node v, the agent receives a noisy estimate of the distance from v to g. We design algorithms for this search task on unknown graphs. We establish the first formal guarantees on unknown weighted graphs and provide lower bounds showing that the algorithms we propose have optimal or nearly optimal dependence on the prediction error. Further, we perform numerical experiments demonstrating that in addition to being robust to adversarial error, our algorithms perform well in typical instances in which the error is stochastic. Finally, we provide alternative simpler performance bounds on the algorithms of [1] for the case of searching on a known graph, and establish new lower bounds for this setting.

The presentation is based on the results in [2].

References

1. S. Banerjee, V. Cohen-Addad, A. Gupta, and Z. Li. Graph Searching with Predictions. In *14th Innovations in Theoretical Computer Science Conference (ITCS 2023)*, Schloss Dagstuhl–Leibniz-Zentrum für Informatik, 2023.
2. A. F. DePavia, E. Tani, and A. Vakilian. Learning-Based Algorithms for Graph Searching Problems. In *Proceedings of the International Conference on Artificial Intelligence and Statistics*, pages 928–936, 2024. PMLR.

4.19 Serving MPE Queries on Tensor Networks by Computing Derivatives

Maurice Wenig (Friedrich-Schiller-Universität Jena, DE)

License Creative Commons BY 4.0 International license
© Maurice Wenig
Joint work of Maurice Wenig, Hanno Barschel, Joachim Giesen, Andreas Goral, Mark Blacher
Main reference Maurice Wenig, Hanno Barschel, Joachim Giesen, Andreas Goral, Mark Blacher: "Serving MPE Queries on Tensor Networks by Computing Derivatives", in Proc. of the Proceedings of The 12th International Conference on Probabilistic Graphical Models, Proceedings of Machine Learning Research, Vol. 246, pp. 515–527, PMLR, 2024.
URL https://proceedings.mlr.press/v246/wenig24a.html

Recently, tensor networks have been proposed as a data structure for weighted model counting. Computing a weighted model count is thus reduced to contracting a factorized tensor expression. Inference queries on graphical models, especially PoE (probability of evidence) queries, can be expressed directly as weighted model counting problems. Maximization problems can also be addressed on the same data structure, only the standard sum-product semiring has to be replaced by either the tropical (max-sum) or the Viterbi (max-product) semiring in the computations, that is, the tensor contractions. However, tensor contractions only provide maximal values, but MPE (most probable explanation) queries on graphical models do not ask for the maximal value, but for a state, or even the states, at which the maximal value is attained. In the special case of tropical tensor networks for ground states of spin glasses, it has been observed that the ground state can be obtained by computing a derivative of the tensor network over the tropical semiring. Here, we generalize this observation, provide a generic algorithm for computing the derivatives, and prove its correctness.

5 Research Directions

5.1 Establishing connections between the related paradigms of machine learning for combinatorial optimization, decision-focused learning, algorithms with predictions and neuro-symbolic computing

We discussed key connections, shared concepts and research directions across the related paradigms of machine learning for combinatorial optimization, decision-focused learning (DFL), algorithms with predictions, and neuro-symbolic computing.

Theme 1: Finding a "Backbone" for Combinatorial Problems. We observed a recurring theme across these paradigms: the identification of a "backbone," a crucial subset of variables that significantly influences the solution of a combinatorial problem. This concept manifests in various forms, such as "learning to prune," "predict then search," and "finding kernel," across different research communities. Herein, we outline key research questions related to formally defining the concept of "backbone" and its effective computation.

- Formalizing the "Backbone" Concept and its Computation: A key challenge lies in formally defining the "backbone" in a general context and finding a generic computational technique to detect a "backbone". While specific problem-tailored techniques for computing "backbone" exist, a more general technique, applicable to problems such as Mixed-Integer Programming (MIP), Constraint Satisfaction Problems (CSP), and general Combinatorial Optimization Problems (COP), is highly desirable.
- Leveraging Symmetries: Investigating how problem symmetries can be exploited to identify and refine the "backbone" represents a promising research direction.
- Constraint Programming Considerations: Exploring the concept of the "backbone" within the context of constraint programming solvers can potentially result in more effective constraint programming solvers.
- Guaranteeing Solution Quality: A critical aspect is to ensure that identifying and potentially reducing the problem based on the "backbone" does not compromise the feasibility or optimality of the solution. Rigorous analysis of the potential risks and error bounds associated with "backbone" predictions is essential.

Theme 2: Learning Effective Proxy Solvers. This theme focused on defining and developing effective "proxy solvers". The research in this direction needs to be cognizant of whether the proxy is being used during training or also during the solving process.

Qualities of a good proxy.

- Accuracy: The proxy solver must accurately reflect the behavior of the original solver, aligning with performance metrics such as regret in decision-focused learning or closely approximating the initial solver's output.
- Computational Efficiency: The proxy solver must be computationally inexpensive to evaluate, ideally allowing for efficient gradient-based optimization and backpropagation.
- Theoretical Guarantees: Establishing theoretical guarantees for the proxy solver is crucial. For example, providing bounds on the error or ensuring that the proxy satisfies specific properties (e.g., ε-approximation) is highly desirable.

Use Cases for Proxy Solvers.

- DFL Objective Functions: Proxy solvers can serve as effective objective functions within the decision focused learning frameworks.
- Improving Combinatorial Solvers: Proxy solvers can be used to enhance the performance of existing combinatorial solvers, such as the use of linear relaxation for obtaining an upper bound.
- Model Pretraining: Proxy solvers can be used to pretrain machine learning models for combinatorial optimization tasks.
- Regret Estimation in DFL: Proxy solvers can be employed to estimate the regret in decision focused learning settings.

Building Effective Proxies. This line of research needs to take into account whether or not we want the proxy to preserve feasibility of solutions. Herein, we outline some potential approaches that can be explored to build effective proxies.

- Contrastive loss function
- Exploring the use of Lagrangian relaxation and decomposition techniques to construct proxies in a differentiable way
- Investigating the applicability of scheduling methods, such as the *envelope method*
- Differentiable Feasibility Pumps
- Investigating the relationship between decision focused learning and inverse optimization and exploring the potential for leveraging the decision focused learning techniques to solve inverse optimization problems

Theme 3: Defining and Exploring Neuro-Symbolic Solvers. This theme focused on defining and exploring the concept of "neuro-symbolic solvers." We can explore the space of "neuro-symbolic solvers" by considering the elements in the "Neuro", "Symbolic" and "Solver" parts.

- **Elements inside the *Neuro* part**: A neural network
- **Elements inside the *Symbolic* part**: A logic (e.g., first-order, fuzzy, temporal, etc.), a search procedure (a CP, MIP, SAT solver, etc.), Markov Random field or a graphical model
- **Elements inside the Solver part**: A way to solve optimization/satisfaction problem

Examples of neuro-symbolic solvers include **DeepProbLog**, **DeepStochLog**, **Toulbar** and methods for weighted model counting. We noted that significant similarities exist between neuro-symbolic approaches and decision focused learning. Historically, neuro-symbolic solvers have often focused on logic, while decision focused learning has primarily focused on constrained optimization. A thorough comparative analysis is needed to identify potential synergies and avoid redundant research efforts in these areas.

5.2 Theoretical frameworks for machine learning augmented combinatorial optimization algorithms

This discussion focused on identifying commonalities in theoretical frameworks and results between contextual optimization, decision-focused learning (DFL), ML augmented CO solvers and algorithms with predictions. For the purpose of this discussion, we established that DFL can be considered a sub-class of contextual optimization problems. We observed that many theoretical results in CO/DFL rely on learning theory, focusing on out-of-sample generalization based on in-sample training data. While ML-augmented CO solvers also exhibit generalization properties, extending these results to prove stronger generalization bounds for techniques like ML-augmented branch-and-bound would be highly valuable. On the other hand, algorithms with predictions often focus on consistency-robustness trade-offs, considering the level of prediction performance as an input. Understanding how to merge these results with the statistical generalization bounds prevalent in CO/DFL would significantly advance our understanding of these interconnected fields.

Specific research questions that emerged in this theme include:

- Robustness in ML-Augmented CO: Can we better understand how ML augmented CO solvers have robustness guarantees that are stronger/more fine-grained than just preserving optimality in the worst-case, and instead depend on the learning guarantees and may use distribution shift concepts (for example) to understand how robust ML augmented solvers are when they are applied to out of distribution data?

- Defining Robustness for CO/DFL: What constitutes the “correct” notion of robustness for CO/DFL? If we consider learning with a perfect future information (PFL) model as a baseline, how can we theoretically compare the performance of DFL and PFL models in a meaningful way?
- Point Predictions in Nonlinear Settings: Under what conditions are point predictions sufficient for 2-stage problems and more generally, for nonlinear optimization problems? How significant can the bias of DFL-trained models be?
- Can we prove some consistency-robustness properties when algorithms with predictions framework is used on branch-and-bound problems? Can we build a bridge between generalization guarantees and algorithms-with-predictions guarantees?
- Fairness and Out-of-Distribution Considerations: How can we incorporate fairness properties and out-of-distribution robustness into DFL formulations? Can counterfactual explanations be leveraged to build fairer DFL models? Can we establish generalization guarantees for fairness metrics in DFL?
- Computational Complexity in DFL, algorithms with prediction and ML augmented combinatorial optimization: How can we define and analyze the computational complexity in these areas, considering factors like the number of gradient or oracle calls? How can we optimally trade off the computational costs of training, inference, and optimization within the DFL framework? What are the optimal sample sizes for training ML-augmented CO algorithms, and are there scaling laws that govern this relationship? Can we get useful lower bounds on the sub-optimality gap while training?
- Dynamic Optimization and Regret: How can we effectively apply ML techniques in dynamic optimization settings, especially when traditional regret bounds may not be well-defined? How can we dynamically update ML models to adapt to changing environments?

5.3 Learning to fill the underspecified parts of algorithms

Many combinatorial optimization algorithms incorporate design choices that do not impact their worst-case theoretical bounds. However, these choices can significantly influence their practical performance, e.g., in terms of running time. By leveraging machine learning techniques to learn optimal or near-optimal choices for typical input instances, we can significantly enhance the efficiency of these algorithms.

We considered two concrete examples:

- The classic Bellman-Ford algorithm for finding shortest paths with negative edge weights relaxes all edges multiple times. While the worst-case running time remains unaffected by the order of edge relaxations, the practical performance can vary dramatically. By learning a near-optimal relaxation order for a given input distribution, we can potentially reduce the number of iterations required for convergence, significantly improving the algorithm’s efficiency.
- In graph partitioning, particularly ϵ-balanced partitioning, multilevel approaches are state-of-the-art. These methods involve coarsening the graph by repeatedly contracting edges, generating an initial partitioning, and then uncoarsening to obtain a partition for the original graph. Current coarsening strategies often rely on greedy heuristics. However, by learning which edges should not be contracted using machine learning techniques, we can potentially improve the quality of the initial partitioning and ultimately the final solution.

Scalability Challenges. A crucial challenge in learning to fill these underspecified parts lies in scalability. Many algorithms, especially those operating on large graphs, require processing millions of edges per second. This stringent requirement necessitates the use of local features, as computing global features can be computationally prohibitive. For instance, in the graph partitioning context, determining which edges should not be contracted often requires global information. However, due to scalability constraints, we are typically limited to using local edge features. To overcome this limitation, we can explore techniques for overestimating global information using local features.

Input Instance Dependence and Training Data. Learning models must be tailored to specific input distributions. Different input distributions may require distinct learning models to achieve optimal performance.

- Data Scarcity and Bias: Obtaining sufficient and representative training data can be challenging. Existing datasets for some graph algorithms may exhibit biases, potentially hindering the generalization ability of learned models.
- Artificial Instance Generation: Generating artificial instances that challenge the algorithm can be crucial for improving the robustness and generalizability of learned models.
- Imitation Learning: Imitation learning techniques can be valuable for collecting high-quality training data by observing the behavior of expert algorithms or human experts.

Dynamic Ordering and Feature Propagation.

- Dynamic Ordering: For algorithms like Bellman-Ford, it might be more effective to learn dynamic orderings, where the order of relaxations changes in each iteration based on information gathered in previous iterations.
- Feature Propagation: In graph-based algorithms, propagating features within the search space (e.g., within a breadth-first search (BFS) tree) can provide valuable information to guide the search process more effectively. Techniques such as graph propagation can be explored to enable such feature propagation.
- Addressing Error Propagation: In iterative algorithms, errors made in early iterations can propagate and significantly impact subsequent iterations. This can pose a challenge for learning-based approaches. Careful consideration must be given to mitigating the impact of early errors on the overall learning process.

5.4 Reinforcement Learning for Combinatorial Optimization

In the context of leveraging reinforcement learning for combinatorial optimization, there has been some progress on end-to-end reinforcement learning (RL) algorithms for combinatorial optimization problems with no global constraints, and for which all constraints can be checked incrementally at each step. The key challenge in this area lies for problems with at least one global constraint. In this case, determining whether a candidate solution satisfies the global constraints often requires evaluating the solution as a whole. This presents a significant challenge, as it is not possible to incrementally check constraint satisfaction during the solution construction process. Potential approaches to address this challenge include:

- Penalty-based Methods: Penalizing "nogoods" (violations of global constraints) during RL training.
- Repair Strategies: Checking constraints at the end of the solving process and then repairing the infeasible solution.

- Predicting Feasibility: Predicting the likelihood that a partial solution can be extended to a complete, consistent solution.
- Perturbation Methods: Working with perturbations of existing feasible solutions rather than directly constructing solutions from scratch.

In contrast to integrating RL within or alongside a combinatorial optimization solver, end-to-end RL approaches may require backtracking mechanisms to handle constraint infeasibility.

A crucial challenge lies in the generalizability of end-to-end and learning-augmented solving. This includes generalization across problem instance sizes, generalization between related problem classes, and the inherent generality of the underlying combinatorial optimization solvers.

Key research questions to address in this area also include (i) developing a comprehensive theoretical framework for reinforcement learning in the context of combinatorial optimization problems and (ii) exploring the potential benefits of active learning techniques for improving the efficiency and effectiveness of learning-based optimization solvers.

5.5 Contextual Stochastic Optimisation and DFL

A key question in the area of machine learning for combinatorial optimization is "Can we develop foundation models that capture the underlying structural properties of different combinatorial optimization tasks?" These foundation models could then be fine-tuned for specific pruning or branching tasks.

Herein, we list some of the other research questions identified in this theme:

- Model Depth: A fundamental question is: How much expressive power, specifically in terms of model depth (e.g., the number of layers in a graph neural network), is necessary for an ML model to effectively address various combinatorial optimization tasks? For instance, what level of expressivity is required for a GNN to act as an effective proxy solver or for learning-to-prune?
- What are the advantages of decision focused learning (DFL) over the standard mean squared error loss functions?
- How can neural networks be effectively trained within the DFL framework?
- What biases get introduced when using point prediction for stochastic settings? There are potential connections between these biases and the field of inverse optimization, which aims to infer the underlying parameters of an optimization problem given observed decisions.
- Can we enable more compact and efficient representations of complex optimization problems in higher-dimensional spaces?
- Drawing inspiration from NLP, where foundation models like large language models (LLMs) are pre-trained on large text corpora, can we explore analogous pre-training tasks for combinatorial optimization? What simple, yet informative, tasks can be used for pre-training in the context of combinatorial optimization? In NLP, next-token prediction serves as a powerful pre-training objective. Can we identify an analogous task for combinatorial optimization problems, perhaps involving predicting partial solutions or exploring the local neighborhood of a given solution?
- What constitutes suitable training data for pre-training such foundation models? Should it include solutions, explored search trees, or other relevant information?

5.6 Fast and/or Interpretable ML Techniques for Combinatorial Problems

When integrating machine learning (ML) models into combinatorial optimization algorithms, the trade-off between efficiency, interpretability and accuracy is crucial. We first outline the motivation for efficiency and interpretability of the learning techniques in the context of optimization problems.

- Efficiency: If an ML model is used as a subroutine within a combinatorial optimization algorithm, it might be invoked very frequently. A fast inference of the learnt model can help ensure scalability of the optimization algorithm to large data.
- Interpretable models can facilitate the development of hand-crafted heuristics, enabling manual optimization and providing a deeper understanding of the algorithm's behavior.
- Interpretability enhances our understanding of how ML-augmented algorithms function, improving our ability to analyze and debug them.

Next, we consider the two popular models in terms of this trade-off between efficiency, interpretability and accuracy.

- Decision Trees: Offer high efficiency and interpretability but may have limited expressive power for complex tasks.
- Neural Networks (NNs): More expressive but generally less efficient and more difficult to interpret.

Specifically, the discussion focused on Graph Neural Networks (GNNs) and we noted that the expressive power of GNNs (without additional features) is equivalent to the Weisfeiler-Leman graph isomorphism test. This connection can be leveraged to reinterpret GNN decisions in terms of the colors assigned by the WL algorithm. However, developing automated techniques for interpreting GNN decisions beyond manual analysis of WL colorings remains an open challenge.

In terms of interpretability of standard neural networks, increasing the number of features or hidden layers in a NN significantly increases its complexity and hinders interpretability. However, the following techniques are used in this context:

- Distillation: Train a simpler model (e.g., using symbolic regression) to imitate the behavior of the original NN.
- LIME: Ranks the feature importance for specific decisions and given input instances.
- Visualization Techniques: Visualize the activation patterns within the NN to understand how it processes information.
- Encode the learning model as a combinatorial optimization problem, e.g., a MIP.
- Model Simplification: Pruning of low-weight connections and quantization can help to simplify a NN.

Unfortunately, the applicability of these techniques is limited and they are often not sufficient to understand complex NNs.

In terms of improving efficiency, the following techniques were discussed:

- Modular Subproblems: Decompose complex problems into smaller, more manageable subproblems, which can be solved more efficiently. An example application for this is planning.
- Select the appropriate ML model (classification, regression, ranking) based on the specific task to optimize efficiency and interpretability.
- GPU Acceleration: Utilize GPUs for efficient inference, although data transfer overhead must be considered. GPU acceleration can be particularly beneficial when the algorithm itself already runs on the GPU.

5.7 Scalability and Datasets for Decision Focused Learning

Decision-focused learning (DFL) has witnessed significant progress in recent years. However, to achieve real-world success and broader applicability, DFL methods must become more scalable. This session focused on open challenges and potential contributions to DFL in terms of:

- Open-Source Code and Datasets: Expanding the availability of open-source code and datasets specifically designed for DFL applications.
- Scalability for Real-World Problems: Enhancing the ability of DFL methods to handle the complexities of real-world data and problem sizes.

Currently, the most widely used open source software for DFL is PyEPO by Khalil et al.[1]. Although it has some limitations, such as being able to be used only for the objective coefficients of a single linear programming instance, it is still a good starting point for implementing DFL methods. The repository of the DFL survey paper [1] has implementations for several problems, namely, optimization, knapsack, matching, portfolio optimization, (Warcraft) shortest path independent of PyEPO.

A key hurdle in research on DFL methodologies is finding real-world data that aligns well with DFL's requirements. For example, in many routing applications, the travel time information is only observed for the selected path. A potential solution to this hurdle is to generate optimization problems synthetically whenever appropriate data is available. An example is the bike-sharing applications where there is a lot of data made available, but a synthetic problem such as a shortest path or traveling salesman problem can be defined on this real data. In addition, there is a need to develop a standardized format for storing data and optimization problems specifically for DFL applications. This would facilitate sharing and collaboration within the DFL research community.

Scalability in DFL can encompass various aspects, including the size of the optimization problem and the available data volume. Notably, problem size doesn't always directly correlate with optimization instance difficulty. To address the scalability issues, a potential approach is to train the learning model with small instances of the optimization problem, and then perform inference on larger instances. This kind of scaling can be done, for example, when the mapping from the features to the problem parameters is deterministic, but the feature is random. This mapping can be learned for small instances in a DFL fashion, and the learned mapping can be used in larger instances.

By addressing these challenges and fostering collaboration through open-source code and datasets, DFL can unlock its full potential for solving real-world decision-making problems at scale.

References

1 Mandi, Jayanta, et al. Decision-focused learning: Foundations, state of the art, benchmark and future opportunities. Journal of Artificial Intelligence Research 80 (2024): 1623-1701.

[1] https://github.com/khalil-research/PyEPO

5.8 Finding worst-case instances for algorithms using machine learning techniques

An important research direction at the intersection between algorithm design and machine learning is to design a learning framework that effectively generates instances that expose the worst-case performance of a given algorithm. Such a framework can greatly accelerate the pace of research in algorithm design and engineering. This is particularly relevant for approximation algorithms where the gap between the upper and lower bounds is very high and the worst-case instances are those that elicit the worst approximation ratio. Such a framework would enable researchers to:

- Identify Algorithmic Weaknesses: Pinpoint specific instances that highlight limitations and potential areas for improvement in existing algorithms.
- Refine Approximation Ratios: Tighten lower bounds on the approximation ratio, providing a more accurate understanding of the algorithm's performance guarantees.
- Guide Algorithm Design: Inform the development of new algorithms that are more robust against the identified worst-case scenarios.

In recent years, some progress was made in this direction by Wagner [1] who used a reinforcement learning framework based on cross-entropy to find counterexamples to some open conjectures. Another recent work [2] improved the state-of-the-art lower bounds for a central extremal graph theory problem which aims to find graphs with a given size (number of nodes) that maximize the number of edges without having 3- or 4-cycles. Recent work [3] in algorithm configuration community has also addressed this question, but from a different perspective.

Since geometric problems generally offer more intuitive visualizations, potentially facilitating the learning and analysis process, we can focus on geometric problems as a more tractable starting point for this research. Here are a couple of examples of such worst-case instances:

- Can we find a configuration of rectangles such that the intersection graph of those rectangles corresponds to a given graph G?
- Can we find a configuration of rectangles such that the ILP formulation (that ensures that the sum of rectangle indicator variables covering any given point sums to atmost 1) for its maximum independent set will have the worst-possible integrality gap?

Potential approaches to address this research direction include:

- Black-Box Optimization: Treat the generation process as a black-box optimization problem, exploring techniques like gradient descent and evolutionary algorithms.
- Penalizing Configurations: Penalize configurations that are known to be not useful for eliciting the worst-case behaviour, including instances for which reduction rules apply easily.
- Solver-Guided Learning: Utilize existing solvers and generators to generate a large number of instances, analyze the performance of the target algorithm on these instances, and use this data to guide the learning process.
- Exploiting Known Hard Instances: Leverage existing knowledge about hard instances to guide the generation process, potentially by creating variations or generalizations of these known difficult configurations.
- Recursive Subgraph Construction: Explore the use of recursive techniques to construct complex structures by combining simpler, known hard subgraphs.
- Identify variables that are always 0 or always 1 in all optimal solutions and removing these variables to simplify and reduce the problem.

References

1 Adam Zsolt Wagner. Constructions in combinatorics via neural networks. CoRR abs/2104.14516 (2021)

2 Abbas Mehrabian, Ankit Anand, Hyunjik Kim, Nicolas Sonnerat, Matej Balog, Gheorghe Comanici, Tudor Berariu, Andrew Lee, Anian Ruoss, Anna Bulanova, Daniel Toyama, Sam Blackwell, Bernardino Romera-Paredes, Petar Velickovic, Laurent Orseau, Joonkyung Lee, Anurag Murty Naredla, Doina Precup and Adam Zsolt Wagner. Finding Increasingly Large Extremal Graphs with AlphaZero and Tabu Search Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence IJCAI 2024

3 Kate Smith-Miles and Mario Andrés Muñoz. Instance Space Analysis for Algorithm Testing: Methodology and Software Tools. ACM Computing Survey 2023

Participants

- Deepak Ajwani
University College Dublin, IE
- Brandon Amos
Meta – New York, US
- Senne Berden
KU Leuven, BE
- Quentin Cappart
Polytechnique Montréal, CA
- Eanna Curran
University College Dublin, IE
- Marianne Defresne
KU Leuven, BE
- Michelangelo Diligenti
University of Siena, IT
- Bistra Dilkina
USC – Los Angeles, US
- Adam Elmachtoub
Columbia University – New York, US
- Aaron Ferber
Cornell University – Ithaca, US
- Alexandre Forel
Polytechnique Montréal, CA
- Emma Frejinger
University of Montreal, CA
- Paul Grigas
University of California – Berkeley, US
- Ernestine Großmann
Universität Heidelberg, DE
- Tias Guns
KU Leuven, BE
- Mikhail Khodak
Princeton University, US
- Sandra Kiefer
University of Oxford, GB
- Alex Lindermayr
Universität Bremen, DE
- Michele Lombardi
University of Bologna, IT
- Nikolai Maas
KIT – Karlsruher Institut für Technologie, DE
- Irfan Mahmutogullari
KU Leuven, BE
- Fredrik Manne
University of Bergen, NO
- Johannes Meintrup
THM – Gießen, DE
- Ulrich Carsten Meyer
Goethe University – Frankfurt am Main, DE
- Sofia Michel
NAVER Labs Europe – Meylan, FR
- Nysret Musliu
TU Wien, AT
- Mathias Niepert
Universität Stuttgart, DE
- Siegfried Nijssen
UC Louvain, BE & KU Leuven, BE
- Ryan O Connor
University College Dublin, IE
- Manuel Penschuck
Goethe University – Frankfurt am Main, DE
- Adam Polak
Bocconi University – Milan, IT
- María Dolores Romero Morales
Copenhagen Business School, DK
- Louis-Martin Rousseau
Polytechnique Montréal, CA
- Jens Schlöter
CWI – Amsterdam, NL
- Christian Schulz
Universität Heidelberg, DE
- Darren Strash
Hamilton College – Clinton, US
- Edward Tansley
University of Oxford, GB
- Sylvie Thiébaux
University of Toulouse, FR & Australian National University, Canberra, AU
- Ali Vakilian
TTIC – Chicago, US
- Jacobus G. M. van der Linden
TU Delft, NL
- Maurice Wenig
Friedrich-Schiller-Universität Jena, DE
- Neil Yorke-Smith
TU Delft, NL